

ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ
ΤΜΗΜΑ ΔΙΔΑΚΤΙΚΗΣ ΤΗΣ ΤΕΧΝΟΛΟΓΙΑΣ ΚΑΙ ΨΗΦΙΑΚΩΝ
ΣΥΣΤΗΜΑΤΩΝ

ΜΕΤΑΠΤΥΧΙΑΚΗ ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

ΜΕΛΕΤΗ ΕΠΙΔΟΣΗΣ ΨΗΦΙΑΚΩΝ
ΔΙΑΜΟΡΦΩΣΕΩΝ ΜΕ ΚΩΔΙΚΟΠΟΙΗΣΗ
ΣΕ ΑΣΥΡΜΑΤΟ ΚΑΝΑΛΙ

Ζώρου Ε. Βασιλική

ΜΑΡΤΙΟΣ 2007

ΠΕΡΙΕΧΟΜΕΝΑ

ΕΙΣΑΓΩΓΗ	ΣΕΛ
ΚΕΦΑΛΑΙΟ 1 “ΨΗΦΙΑΚΗ ΔΙΑΜΟΡΦΩΣΗ”	
1.1 Γενικά	1
1.2 Ψηφιακές Τεχνικές διαμόρφωσης Διέλευσης ζώνης	4
1.3 PSK (Phase Shift Keying)	5
1.3.1 Διαμόρφωση BPSK	6
1.3.2 Διαμόρφωση QPSK	8
1.4 Διαμόρφωση QAM	11
1.4.1 Διαμόρφωση 16QAM	14
1.5 Σύγκριση των τριών παραπάνω διαμορφώσεων	16
ΚΕΦΑΛΑΙΟ 2 “CHANNEL CODING”	
2.1 Γενικά	18
2.2 Convolutional Codes	19
2.2.1 Διάγραμμα Trellis	20
2.2.2 Αποκωδικοποίηση Μέγιστης Πιθανότητας	23
2.3 Αλγόριθμος Viterbi	26
2.4 Ιδιότητες Απόστασης	29
2.5 Διάγραμμα Κατάστασης και Συναρτήσεις Μεταφοράς	31
ΚΕΦΑΛΑΙΟ 3 “AWGN & FADING “	
3.1 Γενικά	36
3.2 Additive White Gaussian Noise	36
3.3 Fading	38
3.4 Fading Models	41
3.5 Παραγωγή φακέλων εξασθένησης Rayleigh	44

ΚΕΦΑΛΑΙΟ 4 “ΑΠΟΔΟΣΗ ΤΩΝ ΨΗΦΙΑΚΩΝ

ΔΙΑΜΟΡΦΩΣΕΩΝ ΣΤΑ ΑΣΥΡΜΑΤΑ

ΚΑΝΑΛΙΑ“

4.1 Γενικά	48
4.2 Κανάλια AWGN	48
4.2.1 Signal-to-Noise Power Ratio and Bit/Symbol Energy	49
4.2.2 Error Probability for BPSK and QPSK	50
4.2.3 Error Probability για MPSK	52
4.2.4 Error Probability for MPAM & MQAM	54
4.3 Κανάλια Fading	56
4.3.1 Outage Probability	57
4.3.2 Μέση Πιθανότητα Λάθους	58
4.4. Moment Generating Function Approach to Average Probability	61

ΚΕΦΑΛΑΙΟ 5 “ ΑΠΟΔΟΣΗ ΤΩΝ ΨΗΦΙΑΚΩΝ ΔΙΑΜΟΡΦΩΣΕΩΝ ΜΕ CODING “

5.1 Γενικά	65
5.2 Ανάλυση Λάθους για convolutional codes	65
5.3 Απαρίθμηση των Διαδρομών μέσα στο Trellis	69
5.4 Node Error Probability P_e και Bit Error Rate P_b	72
5.5 Όριο για την P_b για Discrete Channels	74
5.6 Όριο για την P_d για BPSK σε κανάλι AWGN	75
5.7 Asymptotic Coding Gain	76
5.8 Παράγοντες στην εφαρμογή Convolutional Codes	78

ΚΕΦΑΛΑΙΟ 6 “ ΣΥΣΤΗΜΑ & ΛΟΓΙΣΜΙΚΟ ΠΡΟΣΟΜΟΙΩΣΗΣ”

6.1 Γενικά	81
6.2 Σχεδιασμός Συστήματος	81
6.3 Λογισμικό Προσομοίωσης	84
6.3.1 Παράμετροι Εισαγωγής	85
6.3.2 Χρησιμοποιούμενες Συναρτήσεις	87

6.4 Παράμετροι εισαγωγής & Γραφικές Παραστάσεις	95
6.5 Θεωρητικές εξισώσεις	95
ΚΕΦΑΛΑΙΟ 7 “ΑΠΟΤΕΛΕΣΜΑΤΑ ΠΡΟΣΟΜΟΙΩΣΗΣ”	97
ΚΕΦΑΛΑΙΟ 8 “ΣΥΜΠΕΡΑΣΜΑΤΑ”	111

ΠΑΡΑΡΤΗΜΑΤΑ

ΠΑΡΑΡΤΗΜΑ Ι “ΛΟΓΙΣΜΙΚΟ ΠΡΟΣΟΜΟΙΩΣΗΣ”

ΓΑΛΛΕΣΤΕΛΜΟ ΓΕΡΑΝ

ΠΕΡΙΛΗΨΗ

Στη παρούσα διπλωματική εργασία μελετάται η απόδοση ενός ψηφιακού μικροκυματικού ασύρματου συστήματος, δηλαδή μετρώνται μεγέθη όπως το BER – το SER και το FER κάτω από συγκεκριμένες προϋποθέσεις και εφαρμόζεται η τεχνική του channel estimation για να μελετηθεί η βελτίωση της απόδοσης του καναλιού κατά την μετάδοση με συνθήκες κανονικού fading με και χωρίς την εν λόγω τεχνική.

Ο έλεγχος της απόδοσης ενός τέτοιου συστήματος γίνεται με προσομοίωση του, με κατάλληλο λογισμικό το οποίο εξομοιώνει την λειτουργία του συστήματος υπό διαφορετικές προϋποθέσεις και συνθήκες μετάδοσης και δομής εκπεμπομένων δεδομένων. Το εν λόγω πρόγραμμα δημιουργήθηκε στο MATLAB, και μαζί με έναν αριθμό περιφερειακών συναρτήσεων, οι οποίες υλοποιούν διάφορες λειτουργίες κατά την διαδικασία της εκπομπής-λήψης του σήματος.

Το πρόγραμμα υλοποιεί τέσσερις διαφορετικές περιπτώσεις συγκρότησης συστήματος και μετράει τα παραπάνω αναφερόμενα μεγέθη, ώστε να ελεγχθεί η απόδοση του συστήματος. Οι διαφορετικές περιπτώσεις είναι οι ακόλουθες:

- (1) Μετάδοση μη κωδικοποιημένων (uncoded data) σε
 - i. Rayleigh Flat fading AWGN κανάλι
 - ii. Rayleigh Normal fading AWGN κανάλι
- (2) Μετάδοση κωδικοποιημένων (coded data) σε
 - i. Rayleigh Flat fading AWGN κανάλι
 - ii. Rayleigh Normal fading AWGN κανάλι

Για όλες τις περιπτώσεις εξάγονται οι σχετικές γραφικές παραστάσεις.

Τα συνολικά κεφάλαια που παρουσιάζονται στην παρούσα διπλωματική εργασία είναι οκτώ (8) και το καθένα διαπραγματεύεται την σχετική θεωρία που χρησιμοποιήθηκε κατά την ανάπτυξη του προγράμματος.

Αρχικά παρουσιάζεται η Εισαγωγή, όπου αναφέρεται οι γενικές κατευθύνσεις και τάσεις στην έρευνα, καθώς επίσης και το πεδίο που ερευνήθηκε.

Στο **Κεφάλαιο 1** παρουσιάζεται η θεωρία των διαμορφώσεων για τις οποίες το υλοποιημένο πρόγραμμα 'έτρεξε'. Αυτές είναι η BPSK, QPSK, και η 16QAM.

Στο **Κεφάλαιο 2** περιγράφονται αναλυτικά η δομή και κατασκευή των συνελκτικών κωδίκων (convolutional codes), οι διάφοροι τρόποι περιγραφής τους,

και πιο αναλυτικά το Trellis διάγραμμα. Παρουσιάζεται επίσης η ανάλυση για την αποκωδικοποίηση κατά Viterbi , καθώς επίσης και οι δύο τρόποι αποκωδικοποίησης , soft & hard decision .

Στο **Κεφάλαιο 3** παρατίθεται τα σχετικά στοιχεία για την θεωρία του AWGN καναλιού , το φαινόμενο fading και η επίδραση του στην συμπεριφορά του σήματος. Επίσης υπάρχει η σχετική αναφορά για τα διάφορα Fading Models , με ιδιαίτερη έμφαση σ' αυτό της Rayleigh κατανομής και του τρόπου παραγωγής τέτοιων φακέλων στο matlab.

Στα **Κεφάλαια 4 & 5** παρουσιάζεται η ανάλυση για την απόδοση των συστημάτων για τις περιπτώσεις όπου τα υπό μετάδοση data είναι μη κωδικοποιημένα (uncoded – Κεφάλαιο 4) , καθώς επίσης και για την περίπτωση όπου τα δεδομένα προς μετάδοση κωδικοποιούνται και αποκωδικοποιούνται κατά Viterbi . Παρατίθεται επίσης σχετικές γραφικές παραστάσεις που δείχνουν τη σχέση μεταξύ SNR και BER και SER και για τις δύο περιπτώσεις.

Στο **Κεφάλαιο 6** , παρουσιάζεται αναλυτικά το λογισμικό που αναπτύχθηκε , οι μεταβλητές του, τα δεδομένα εισαγωγής, οι συναρτήσεις που υλοποίησαν τμήματα της εκπομπής- λήψης , όπως την διαμόρφωση/αποδιαμόρφωση , την κωδικοποίηση/αποκωδικοποίηση, καθώς επίσης και την τεχνική που χρησιμοποιήθηκε .

Στο **Κεφάλαιο 7** , δεικνύονται οι σχετικές γραφικές παραστάσεις , οι οποίες εξήχθησαν από την προσομοίωση, για κάθε μία από τις τέσσερις διαφορετικές περιπτώσεις και συγκρίνονται με τα θεωρητικά αποτελέσματα , που παράγονται από τις θεωρητικές εξισώσεις .

Τέλος στο **Κεφάλαιο 8** καταγράφονται τα συμπεράσματα τα οποία εξήχθησαν από την μελέτη των γραφικών παραστάσεων.

ΕΙΣΑΓΩΓΗ

Η τεχνολογική ανάπτυξη στο τομέα των τηλεπικοινωνιών ήταν ραγδαία την τελευταία δεκαετία και συνεχίζει να εξελίσσεται με γρήγορους ρυθμούς. Ιδιαίτερη ανάπτυξη γνωρίζουν οι κάθε μορφής ασύρματες επικοινωνίες .

Οι ασύρματες επικοινωνίες στις μέρες μας πλέον, περιλαμβάνουν όλες σχεδόν τις μορφές , όπως είναι η φωνή , η πρόσβαση στο INTERNET , το WEB BROWSING , paging, short messages , file transfer κλπ. Συστήματα όπως η κινητή τηλεφωνία , τα wireless LAN, τα satellite system και άλλα συμπεριλαμβάνονται στις , εξελιγμένες μορφές ασύρματης επικοινωνίας.

Η γεωγραφική κάλυψη που παρέχετε απ' αυτά τα συστήματα είναι ευρεία και ξεκινάει από μερικά μέτρα και επεκτείνεται σε εκατοντάδες όπως μέσα σε μεγάλα κτίρια, σε Πανεπιστημιούπολεις, σε πόλεις , κράτη κλπ. Το γεγονός ότι υπάρχει μεγάλη ποικιλία σε στάνταρ και υλικό οφείλεται στις διαφορετικές εφαρμογές και απαιτήσεις που αυτές έχουν. Για παράδειγμα η μετάδοση φωνής δεν απαιτεί υψηλές ταχύτητες (~20Kb/s) και έχει μεγάλη αντοχή σε λάθη (BER ~ 10^{-3}) , αλλά η συνολική καθυστέρηση δεν μπορεί να είναι >30 msec ώστε να μην γίνεται αντιληπτή από το ανθρώπινο αυτί. Απ' την άλλη μεριά συστήματα μετάδοσης δεδομένων απαιτούν υψηλότερες ταχύτητες (1-100 Mbps) και πολύ μικρό ρυθμό σφαλμάτων ((BER ~ 10^{-8}) , ενώ ταυτόχρονα τα λανθασμένα bit πρέπει να επαναμεταδίδονται. Εδώ η καθυστέρηση (delay) δεν είναι αποφασιστικής σημασίας.

Οι υψηλές απαιτήσεις για μεγάλες ταχύτητες με ταυτόχρονη μείωση του BER απαιτούν την εξεύρεση νέων τεχνικών οι οποίες θα υποστηρίζουν αυτές τις απαιτήσεις. Η εκθετική αύξηση στη χρήση των κινητών επικοινωνιών και του ασύρματου INTERNET ώθησαν στην αύξηση των τεχνικών απαιτήσεων οι οποίες όλο και περισσότερο βελτιώνονται ενώ ταυτόχρονα αναζητούνται νέες τεχνικές για την επίλυση των θεμάτων που θα ικανοποιούν τις υψηλές απαιτήσεις των χρηστών.

Οι προκλήσεις οδήγησαν στην ανάγκη εξεύρεσης νέων τεχνικών οι οποίες θα οδηγούσαν στη βελτίωση των παραμέτρων του σήματος . Μία απ' αυτές είναι η τεχνική της Adaptive Modulation & Coding. Με την τεχνική αυτή επιτυγχάνεται η αποδοτική και αποτελεσματική λειτουργία και μετάδοση μέσα από διαύλους χρονικά μεταβαλλόμενους. Βασική προϋπόθεση είναι ο υπολογισμός των παραμέτρων του καναλιού μετάδοσης στον δέκτη, και η επανατροφοδότηση του πομπού με τα

στοιχεία αυτά ώστε να προσαρμόσει το σχέδιο εκπομπής (χρησιμοποιούμενη διαμόρφωση , κωδικοποίηση κλπ) στα χαρακτηριστικά του καναλιού.

Οι τεχνικές των adaptive transmission ξεκίνησαν να ερευνώνται προς το τέλος της δεκαετίας του '60 και αρχές του '70. Η εξέλιξη τότε δεν ήταν η αναμενόμενη πιθανώς λόγω περιορισμών στο hardware , και έλλειψης επαρκών τεχνικών για τον υπολογισμό των παραμέτρων του καναλιού. Το ενδιαφέρον ξεκίνησε πάλι όταν άρχισε να αναπτύσσεται η 3^η γενιά ασύρματων συστημάτων με αποτέλεσμα πολλά συστήματα, συμπεριλαμβανομένων και των κυψελωτών (GSM , CDMA) , τα wireless LAN , να υιοθετήσουν την χρήση των τεχνικών των adaptive transmission .

Υπάρχουν πρακτικοί περιορισμοί οι οποίοι καθορίζουν πότε η τεχνική της adaptive modulation μπορεί να χρησιμοποιηθεί . Η τεχνική αυτή απαιτεί ένα δίαυλο επιστροφής από τον δέκτη προς τον πομπό που πολλές φορές αυτό δεν γίνεται εφικτό. Επίσης , εάν το κανάλι μεταβάλλεται γρηγορότερα απ' ότι μπορεί να υπολογιστεί π.χ σε multipath fading τότε η adaptive modulation μπορεί να μην μπορεί να εφαρμοστεί.

Οι τεχνικές διαμόρφωσης και κωδικοποίησης οι οποίες δεν υιοθετούν τις συνθήκες εξασθένισης απαιτούν ένα σταθερό περιθώριο ζεύξης ώστε να μπορούν να συντηρήσουν μία αποδεκτή απόδοση όταν οι συνθήκες μετάδοσης είναι φτωχές.

Από τη στιγμή που η εξασθένιση κατά Rayleigh μπορεί να προκαλέσει απώλεια σήματος ακόμα και 30 db , ο σχεδιασμός για τη λειτουργία στη χειρότερη περίπτωση οδηγεί σε ανεπαρκή χρήση του φάσματος. Ενώ όταν υιοθετούνται τεχνικές που οδηγούν στην αποδοχή των συνθηκών εξασθένισης του καναλιού , αυτό οδηγεί στην αύξηση του μέσου ολικού ρυθμού μετάδοσης (throughput) , μειώνει την απαιτούμενη μέση ισχύ εκπομπής ή μειώνει την μέση πιθανότητα του bit error εκμεταλλευόμενο τις συνθήκες καναλιού για να μεταδώσει σε υψηλότερες ταχύτητες ή να μειώσει την ισχύ και μειώνοντας το data rate ή αυξάνοντας την ισχύ.

Υπάρχουν πολλές παράμετροι οι οποίες μπορεί να μεταβάλλονται κατά τη μετάδοση και οι οποίες σχετίζονται με το channel gain γ , όπως είναι το data rate , το power coding , η error probability ή και συνδυασμός αυτών. Τα προσαρμόσιμα συστήματα μετάδοσης συνδέονται με την προσαρμοσμένη μετάδοση. Το μοντέλο είναι αυτό που χρησιμοποιείται για να καθοριστεί η χωρητικότητα στα κανάλια με flat fading. Το κανάλι θεωρούμαι ότι έχει σταθερό , κατά τη διάρκεια ενός συμβόλου, και ανεξάρτητο χρόνου , $gain = \sqrt{g[i]}$ το οποίο ακολουθεί μία συγκεκριμένη κατανομή $p(g)$, ενώ ο θόρυβος AWGN $n[i]$ έχει πυκνότητα ισχύος ίση με $N_0/2$.

Στη παρούσα διπλωματική γίνεται προσπάθεια να προσεγγιστεί το παραπάνω μοντέλο και να υπολογιστεί το Gain του καναλιού κάνοντας χρήση pilot symbols.

Η προσομοίωση του παραπάνω μοντέλου, η εύρεση του gain του καναλιού, και η επίδραση του στην απόδοση του συστήματος για τα τρία διαφορετικά σχήματα διαμόρφωσης που ελέγχονται (BPSK, QPSK, 16QAM) δεικνύεται σε συνδυασμό με τα θεωρητικά μοντέλα.

Στο Κεφάλαιο 6 αναλύεται το λογισμικό και οι παράμετροι που χρησιμοποιήθηκαν , ενώ στο Κεφάλαιο 7 δίνονται όλες οι περιπτώσεις για τις οποίες 'έτρεξε' το λογισμικό υπό μορφή γραφικών παραστάσεων . Τέλος στο Κεφάλαιο 8 παρουσιάζονται τα συμπεράσματα που εξάγονται από αυτές τις γραφικές παραστάσεις.

ΚΕΦΑΛΑΙΟ 1

ΨΗΦΙΑΚΗ ΔΙΑΜΟΡΦΩΣΗ

1.1 Γενικά

Η ψηφιακή διαμόρφωση είναι μία διαδικασία κατά την οποία ψηφιακά σύμβολα μετασχηματίζονται σε κυματομορφές οι οποίες είναι συμβατές με τα χαρακτηριστικά του καναλιού μετάδοσης. Στη περίπτωση της διαμόρφωσης βασικής ζώνης (baseband modulation), αυτές οι κυματομορφές συνήθως παίρνουν τη μορφή των σχηματισμένων παλμών.

Στη περίπτωση της διαμόρφωσης διέλευσης ζώνης (bandpass modulation) οι παλμοί διαμορφώνουν ένα ημίτονο, το οποίο καλείται φορέας (carrier wave), και για την ασύρματη μετάδοση ο φορέας αυτός μετατρέπεται σε ηλεκτρομαγνητικό κύμα το οποίο μεταδίδεται προς την επιθυμούμενη κατεύθυνση.

1.2 Ψηφιακές τεχνικές διαμόρφωσης διέλευσης ζώνης

Η διαμόρφωση διέλευσης ζώνης είναι η διαδικασία κατά την οποία το σήμα που περιέχει την πληροφορία μετατρέπεται σε μία ημιτονοειδής κυματομορφή όπου για την περίπτωση της ψηφιακής διαμόρφωσης, η περίοδος T του σήματος είναι η διάρκεια του ψηφιακού συμβόλου. Η ημιτονοειδής κυματομορφή έχει τρία βασικά χαρακτηριστικά: το πλάτος, την συχνότητα και τη φάση. Η διαμόρφωση διέλευσης ζώνης ορίζεται ως η διαδικασία όπου ένα από τα παραπάνω χαρακτηριστικά ή και συνδυασμός αυτών μεταβάλλεται, ανάλογα με την πληροφορία που μεταφέρεται. Η μαθηματική έκφραση του φορέα είναι:

$$s(t) = A(t) \cos\theta(t) \quad (1.1)$$

όπου το $A(t)$ είναι το χρονικά μεταβαλλόμενο πλάτος, και το $\theta(t)$ είναι η χρονικά μεταβαλλόμενη γωνία η οποία μπορεί να γραφεί ως εξής:

$$\theta(t) = \omega_0(t) + \varphi(t) \quad (1.2)$$

Η εξίσωση 1.1 μπορεί να γραφεί ως:

$$s(t) = A(t) \cos[\omega_0(t) + \varphi(t)] \quad (1.3)$$

, όπου ω_0 είναι η συχνότητα του φορέα, και $\varphi(t)$ η φάση.

Βασικοί τύποι διαμορφώσεων ζώνης διέλευσης, είναι η PSK, FSK, ASK, MSK. Όταν ο δέκτης αξιοποιεί την γνώση της φάσης του φορέα για να ανιχνεύσει το λαμβανόμενο σήμα τότε η διαδικασία αυτή ονομάζεται συμφασική ανίχνευση (coherent detection), ενώ όταν ο δέκτης δεν την χρησιμοποιεί μη συμφασική ανίχνευση (noncoherent detection). Στην περίπτωση της ιδανικής συμφασικής ανίχνευσης, στον δέκτη είναι διαθέσιμο ένα πρωτότυπο του κάθε πιθανού σήματος που θα ληφθεί. Έτσι κάθε σήμα που λαμβάνεται συγκρίνεται με το πρωτότυπο και ο δέκτης «κλειδώνει» στη φάση του εισερχόμενου σήματος.

Αντίθετα στη μη συμφασική διαμόρφωση ο δέκτης δεν γνωρίζει τις τιμές της φάσης του εισερχόμενου σήματος, και συνεπώς δεν γίνεται υπολογισμός της φάσης. Το πλεονέκτημα της μη συμφασικής ανίχνευσης έναντι της συμφασικής είναι ότι μειώνεται η πολυπλοκότητα στον σχεδιασμό του δέκτη, αλλά η τιμή γι' αυτό είναι η αυξημένη πιθανότητα εμφάνισης λαθών.

Στο παρόν κεφάλαιο θα αναπτυχθούν αναλυτικά τρεις τύποι ψηφιακής διαμόρφωσης οι οποίοι είναι ειδικές περιπτώσεις της κατηγορίας PSK & ASK, η BPSK, QPSK και η 16QAM.

1.3 PSK (Phase Shift Keying)

Η διαμόρφωση αυτή έχει αναπτυχθεί πριν αρκετά χρόνια και είναι ευρέως χρησιμοποιούμενη. Στη διαμόρφωση αυτή δεδομένα μεταφέρονται διαμορφώνοντας τη φάση του φέροντος. Η γενική αναλυτική έκφραση για την PSK είναι :

$$s_i(t) = \sqrt{\frac{2E}{T}} \cos[\omega_c t + \phi_i(t)] \quad 0 \leq t \leq T$$
$$i = 1, \dots, M \quad (1.4)$$

Όπου ο όρος $\phi_i(t)$ είναι η φάση, και έχει M διακριτές τιμές οι οποίες δίνονται από τον ακόλουθο τύπο :

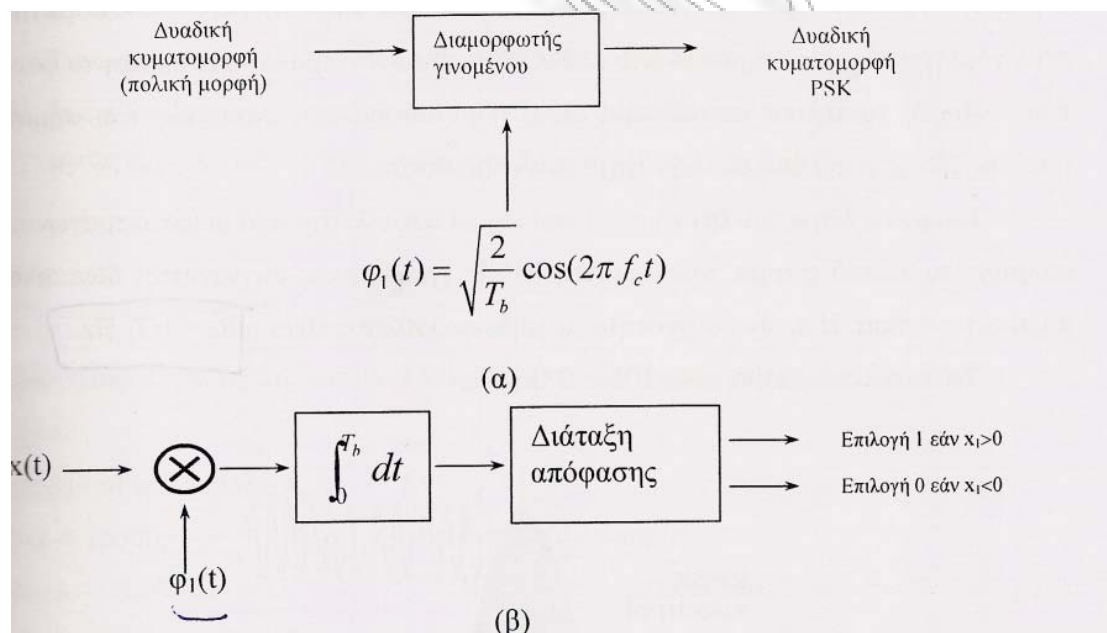
$$\phi_i(t) = (2\pi i)/M \quad i=1, \dots, M.$$

Η παράμετρος E είναι η ενέργεια του συμβόλου που μεταδίδεται, ενώ T είναι η διάρκεια του συμβόλου.

1.3.1 Διαμόρφωση BPSK (Binary Phase Shift Keying)

Η BPSK είναι η απλούστερη μορφή της PSK. Χρησιμοποιεί δύο φάσεις οι οποίες έχουν διαφορά 180° . Έχει μεγάλη ανοχή στο θόρυβο, δεδομένου ότι απαιτείται μεγάλη αλλαγή ώστε να παραπλανηθεί ο δέκτης για να πάρει τη σωστή απόφαση για το σύμβολο. Από την άλλη όμως δεδομένου του γεγονότος ότι μπορεί να διαμορφώνει 1 bit/symbol δεν θεωρείται κατάλληλη για εφαρμογές που απαιτούν υψηλές ταχύτητες.

Το παρακάτω σχήμα δείχνει την υλοποίηση ενός συστήματος εκπομπής-λήψης ενός BPSK σήματος.



Σχήμα 1.1 Διαγράμματα για (α) πομπό BPSK και (β) ομόδουνο δέκτη BPSK

Όπως φαίνεται και από το παραπάνω σχήμα, τα εισαγόμενα ψηφιακά δεδομένα 0 και 1 μετατρέπονται απ' ευθείας στις φάσεις 0 και π αντίστοιχα. Εκφράζοντας μαθηματικά το παραπάνω έχουμε :

$$S(t) = A \cos \{ 2\pi f_c t + \pi d_k \} \quad (1.5)$$

Όπου d_k είναι η ακολουθία των δεδομένων της πληροφορίας.

Σύμφωνα με το διάγραμμα η κυματομορφή ενός BPSK σήματος, δημιουργείται από τον πολλαπλασιασμό του ψηφιακού σήματος των δεδομένων και ενός φορέα .

Δεδομένου ότι υπάρχει ο περιορισμός του εύρους ζώνης θα πρέπει να χρησιμοποιήσουμε ένα Pulse Shaping Filter , πριν προχωρήσουμε στην διαδικασία της διαμόρφωσης. Κατόπιν το σήμα εισάγεται σ' έναν A/D Converter , και πολλαπλασιάζοντας με το φέρον ($A \cos 2\pi f_c t$), του οποίου η συχνότητα είναι η $f_c = n_c / T_b$ για κάποιο ακέραιο n_c), παράγεται το σήμα BPSK , το οποίο και μεταδίδεται.

Αντίστροφα και κατά την λειτουργία της λήψης , η λαμβανόμενη κυματομορφή , αφού πρώτα περάσει από ένα φίλτρο διέλευσης (BandPass Filter), πολλαπλασιάζεται με τον αντίστοιχο φορέα ($A \cos 2\pi f_c t$) . Κατόπιν δειγματοληπτείται , και μετατρέπεται σε αναλογικό σήμα (A/D Converter) . Το τελικό ψηφιακό σήμα (0,1) λαμβάνεται μετά την διαδικασία απόφασης , μέσω ενός κυκλώματος το οποίο <μεταφράζει> τα λαμβανόμενα σύμβολα. Η απόφαση αυτή λαμβάνεται ως εξής :

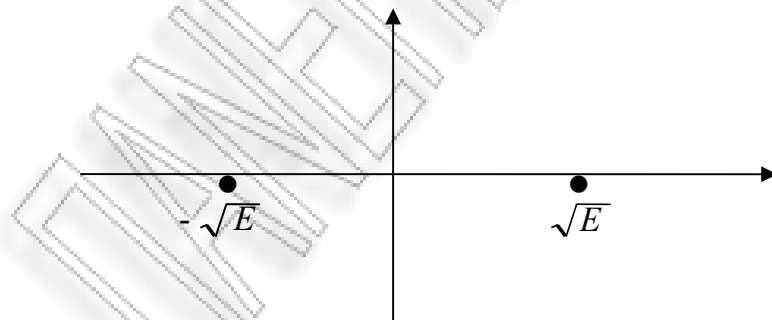
Εάν η έξοδος του συσχετιστή x_1 τη χρονική στιγμή $t=T_b$, είναι μεγαλύτερη από το κατώφλι των $0V (x_1 > 0)$, τότε αποφασίζετε ότι πρόκειται για το σύμβολο 1 , ενώ εάν είναι ($x_1 < 0$) τότε αποφασίζετε ότι πρόκειται για το σύμβολο 0.

Όλα τα σήματα PSK μπορούν να απεικονιστούν γραφικά σ' ένα δύο διαστάσεων σύστημα αναφοράς το οποίο ονομάζεται signal constellation.

Στο σύστημα αυτό ο οριζόντιος και κάθετος άξονας είναι :

$$2/T_b \cos(2\pi f_c t) \ \& \ -2/T_b \sin(2\pi f_c t) \ \text{ με } 0 < t < T_b$$

Το constellation diagram για ένα σήμα BPSK δίνεται από το :



Σχήμα 1.2 BPSK signal constellation

Για κάθε τύπου MPSK διαμόρφωσης , στην οποία k bits μετατρέπονται σε σύμβολα και κάθε σύμβολο απεικονίζεται σ' ένα M -ary αλφαβητάριο ,τότε έχουμε ότι :

$$M=2^k \text{ όπου } k=\log_2 M \text{ (για BPSK } M=2)$$

Εάν T_s ο χρόνος μετάδοσης ενός συμβόλου και $R = \text{data rate}$ τότε :

$$R=k/ T_s = \log_2 M/ T_s \text{ bits/s}$$

Η διάρκεια ενός Bit είναι T_b , τότε :

$$T_b = 1/R = T_s/k = 1/k R_s$$

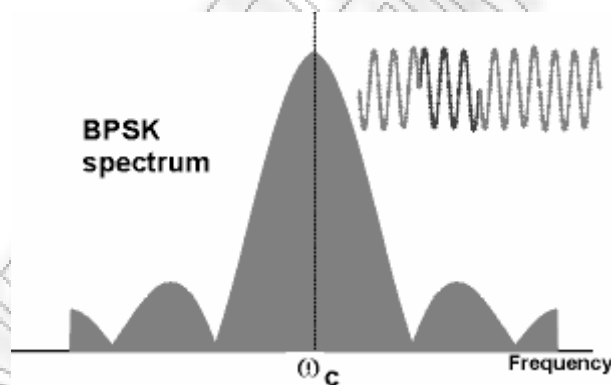
Χρησιμοποιώντας τις εξισώσεις τότε μπορούμε να εκφράσουμε το symbol rate συναρτήσει του bit rate :

$$R_s = R/ \log_2 M$$

Από τις παραπάνω εξισώσεις γίνεται φανερό ότι για οποιαδήποτε ψηφιακή διαμόρφωση κατά την οποία μεταδίδονται $k (= \log_2 M)$ bits , σε διάστημα T_s sec, χρησιμοποιώντας ένα φάσμα W Hz , τότε η φασματική απόδοση δίνεται :

$$R/W = \log_2 M/W T_s = 1/ W T_s \text{ bits/sec/Hz}$$

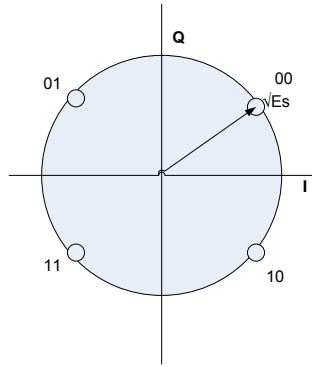
Από τον παραπάνω τύπο , για την BPSK διαμόρφωση έχουμε ότι η φασματική απόδοση είναι 0.5 bit/sec/Hz.



Σχήμα 1.3 Φασματική απεικόνιση BPSK σήματος για θετικές συχνότητες

1.3.2 Διαμόρφωση QPSK

Η διαμόρφωση αυτή είναι μίας από τις μορφές PSK , και πολλές φορές ονομάζεται και 4-PSK. Το constellation diagram, της QPSK έχει ως εξής :



Σχήμα 1.4 Διάγραμμα Αστερισμού QPSK

Σύμφωνα με το ανωτέρω διάγραμμα το οποίο χρησιμοποιεί Gray Coding , δηλαδή κάθε παρακείμενο σύμβολο διαφέρει κατά ένα bit από το διπλανό του.. Η QPSK χρησιμοποιεί τέσσερις φάσεις και μπορεί να κωδικοποιήσει 2 bits/symbol , δηλαδή για κάθε σύμβολο χρησιμοποιούνται 2 bits και ως εκ τούτου έχει τον διπλάσιο ρυθμό μετάδοσης από την BPSK. Έτσι μπορεί να χρησιμοποιηθεί είτε για να διπλασιάσει τον ρυθμό μετάδοσης των δεδομένων (data rate) , ενώ ταυτόχρονα χρησιμοποιεί το ίδιο φάσμα (bandwidth) με την BPSK, ή διατηρώντας το ίδιο data rate με την BPSK , χρησιμοποιεί το μισό φάσμα .

Η QPSK χρησιμοποιεί δύο βασικές συναρτήσεις την sine και την cosine ως φορείς για να δημιουργήσει τα σύμβολα. Μεταβάλλοντας τις φάσεις αυτών των φορέων εκπέμπει 2 bits/Hz .

Όπως δεικνύεται και από το διάγραμμα αστερισμού , υπάρχουν τέσσερα σύμβολα , για κάθε ένα από τα οποία η μαθηματική έκφραση είναι η ακόλουθη :

Symbols	Bits	Έκφραση	Phase	I	Q
S1	00	$\sqrt{\frac{2E_s}{T}} \cos(\omega t + \pi / 4)$	45°	$\frac{1}{\sqrt{2}}$	$\frac{1}{\sqrt{2}}$
S2	01	$\sqrt{\frac{2E_s}{T}} \sin(\omega t + 3\pi / 4)$	135°	$-\frac{1}{\sqrt{2}}$	$\frac{1}{\sqrt{2}}$
S3	11	$\sqrt{\frac{2E_s}{T}} \cos(\omega t + 3\pi / 4)$	225°	$-\frac{1}{\sqrt{2}}$	$-\frac{1}{\sqrt{2}}$
S4	10	$\sqrt{\frac{2E_s}{T}} \sin(\omega t + \pi / 4)$	310°	$\frac{1}{\sqrt{2}}$	$-\frac{1}{\sqrt{2}}$

Πίνακας 1

Ο βασικές συναρτήσεις (I &Q) δίνονται από :

$$\phi_1(t) = \sqrt{\frac{2}{T_s}} \cos(2\pi f_c t) \quad \text{χρησιμοποιείται για το Inphase componet του σήματος.}$$

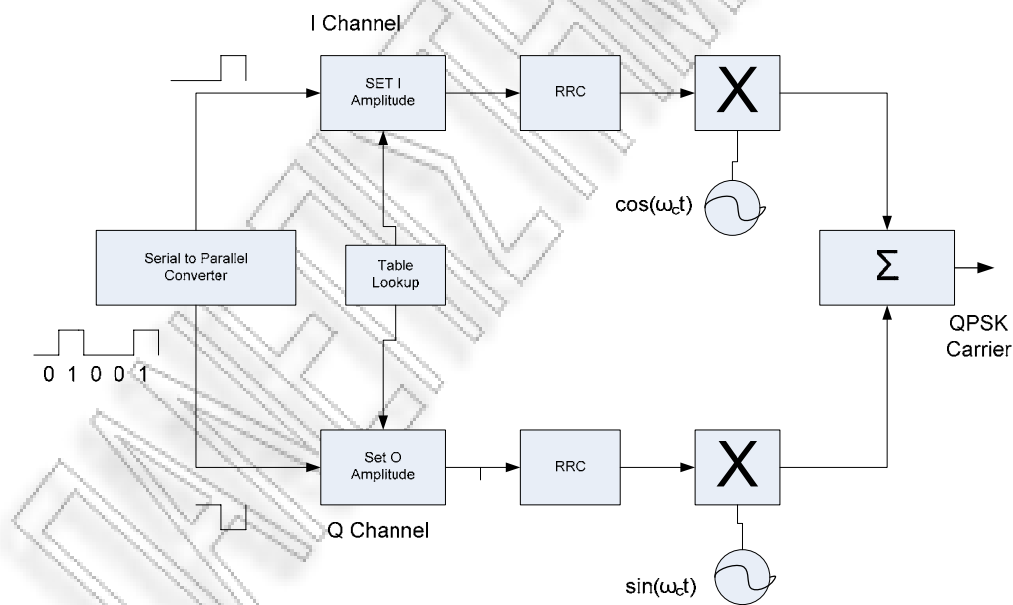
$$\phi_2(t) = \sqrt{\frac{2}{T_s}} \sin(2\pi f_c t) \quad \text{χρησιμοποιείται για το Quadure componet του σήματος}$$

Και αποτελούν ορθοκανονική βάση.

Και η μαθηματική έκφραση σύμφωνα με την οποία δίνονται τα σύμβολα στο διάγραμμα αστερισμού είναι :

$$s_i(t) = \sqrt{\frac{2E_s}{T}} \cos(2\pi f_c t + (2i-1)\frac{\pi}{4}) \quad i = 1,2,3,4$$

Η υλοποίηση ενός QPSK σήματος δεικνύεται στο παρακάτω σχήμα

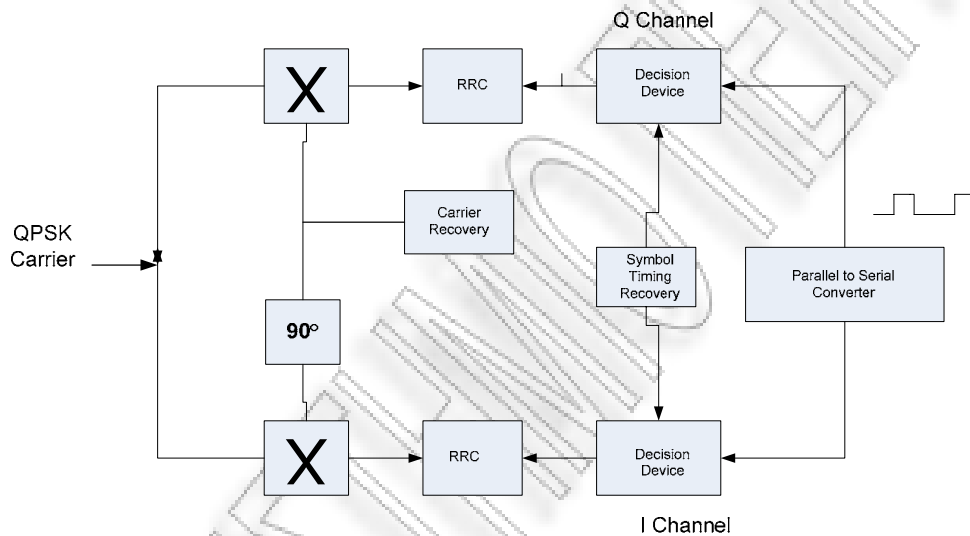


Σχήμα 1.5 Δημιουργία QPSK σήματος

Το εισερχόμενο bit stream αφού μετατραπεί σ' ένα σήμα NRZ (polar) , διαχωρίζεται στο Inphase stream (I Channel) , και στο O stream (Q channel) . Τίθενται τα πλάτη των I & Q , σύμφωνα με το Πίνακα 1. Κατόπιν χρησιμοποιούμε

Root Raised Cosine Filter , και στη συνέχεια πολλαπλασιάζουμε το I channel με το cosine της φέρουσας , συχνότητας f_c , και το Q channel με το sine της φέρουσας . Τέλος προσθέτουμε τα δύο channel και έχουμε το τελικά διαμορφωμένο σήμα κατά QPSK.

Ακολουθώντας αντίστροφη διαδικασία , στη πλευρά του δέκτη έχουμε την ανίχνευση του σήματος QPSK η οποία στηρίζεται στην ανάλυση του σήματος εισόδου στις δύο συνιστώσες , οι οποίες κάθε μία ανιχνεύεται ως μία BPSK.

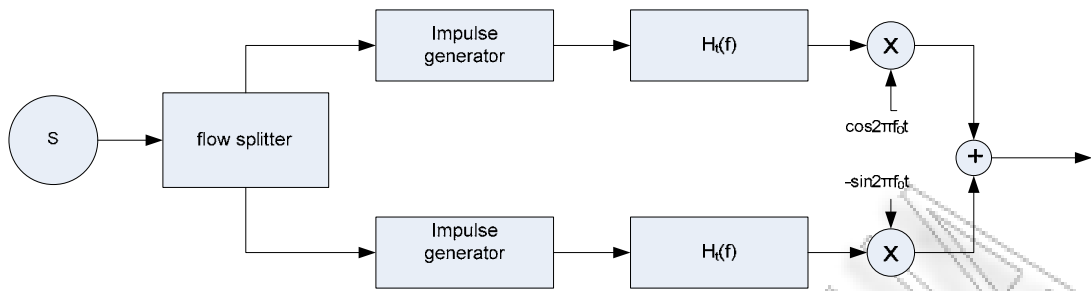


Σχήμα 1.6 Ανίχνευση QPSK

1.4 Διαμόρφωση QAM

Ο τύπος αυτός της διαμόρφωσης χρησιμοποιεί δύο ημιτονικούς φορείς με διαφορά φάσης μεταξύ τους 90° . Επειδή αυτοί οι δύο ορθογώνιοι φορείς χρησιμοποιούν την ίδια ζώνη συχνοτήτων και διαφέρουν μεταξύ τους κατά 90° ο καθένας μπορεί να διαμορφωθεί ανεξάρτητα μεταδίδοντας πάνω στην ίδια ζώνη και ο διαχωρισμός τους πραγματοποιείται στον δέκτη. Για ένα δεδομένο διαθέσιμο εύρος ζώνης , η διαμόρφωση QAM δίνει την δυνατότητα για μετάδοση σε διπλάσια ταχύτητα απ' αυτήν της PAM χωρίς καμία απόκλιση στο BER .

Το ακόλουθο σχήμα δίνει , σε ιδανική μορφή έναν πομπό QAM σήματος με συχνότητα φορέα f_0 και H_f η απόκριση συχνότητας του φίλτρου.



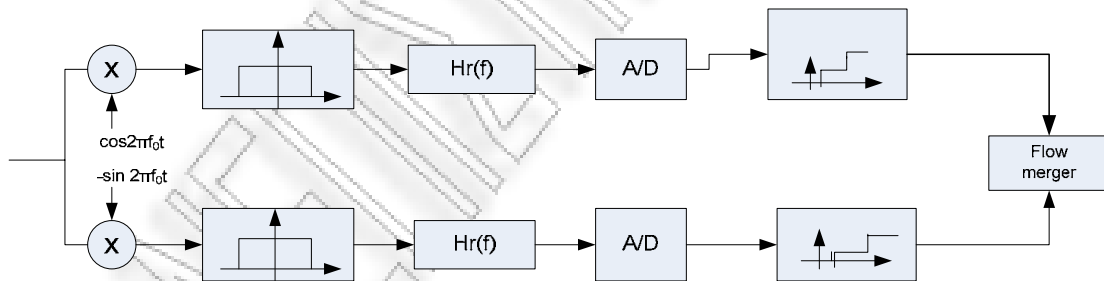
Σχήμα 1.7 Δημιουργία QAM σήματος

Όπου το σήμα εξόδου μπορεί να παρασταθεί από την ακόλουθη μαθηματική έκφραση :

$$s(t) = \sum_{n=-\infty}^{\infty} [u_c[n] * h_t(t - nT_s) \cos(2\pi f_0 t) - u_s[n] * h_t(t - nT_s) \sin(2\pi f_0 t)] \quad (1.6)$$

όπου η $u_c[n]$ και η $u_s[n]$ είναι οι τάσεις που εφαρμόζονται στο n^{th} σύμβολο στο cos και sin αντίστοιχα.

Στο δέκτη εφαρμόζεται η αντίστροφη διαδικασία, και το παρακάτω σχήμα δείχνει ακριβώς τον τρόπο.



Σχήμα 1.8 Ανίχνευση QAM σήματος

Εναλλακτικά ένα σήμα QAM μπορεί να παρασταθεί ως άνυσμα ως εξής :

$$s_m(t) = \text{Re}[V_m e^{j\theta_m} g(t) e^{2\pi f_c t}] = V_m g(t) \cos(2\pi f_c t + \theta_m), \quad m = 1, 2, \dots, M \quad (1.7)$$

όπου

$$V_m = \sqrt{A_{m1}^2 + A_{m2}^2}, \quad \theta_m = \tan^{-1} \left(\frac{A_{m2}}{A_{m1}} \right) \quad (1.8)$$

είναι το πλάτος και η φάση, αντίστοιχα, του m-οστού σήματος από το M-αδικό signal set. Από την έκφραση αυτή είναι προφανές ότι οι κυματομορφές που προκύπτουν από τη διαμόρφωση αυτή μπορούν να θεωρηθούν **ως συνδυασμός διαμόρφωσης πλάτους και φάσης**. Η απόσταση του κάθε σημείου από το κέντρο των αξόνων δηλώνει το πλάτος ενώ η γωνία που σχηματίζει με τον οριζόντιο άξονα καθορίζει τη φάση του εκπεμπόμενου σήματος.

Από τις παραπάνω σχέσεις επίσης προκύπτει ότι:

$$\begin{aligned} A_{m1} &= V_m \cos(\theta_m) = \sqrt{\frac{2E_{\min}}{E_g}} a_m \\ A_{m2} &= V_m \sin(\theta_m) = \sqrt{\frac{2E_{\min}}{E_g}} b_m \end{aligned} \quad (1.9)$$

όπου E_{\min} είναι η ενέργεια του σήματος με το μικρότερο πλάτος, και (a_m, b_m) είναι το ζευγάρι των ανεξάρτητων ακεραίων ανάλογα με τη θέση του συγκεκριμένου signal point. Όπως και στην περίπτωση των σημάτων M-PSK, έτσι και στο QAM, οι κυματομορφές μπορούν να παρασταθούν ως ένας γραμμικός συνδυασμός δύο ορθογώνιων κυματομορφών $\varphi_1(t)$ και $\varphi_2(t)$,

$$s_m(t) = s_{m1} \varphi_1(t) + s_{m2} \varphi_2(t) \quad (1.10)$$

όπου

$$\varphi_1(t) = \sqrt{\frac{2}{E_g}} g(t) \cos(2\pi f_c t) \quad \varphi_2(t) = -\sqrt{\frac{2}{E_g}} g(t) \sin(2\pi f_c t) \quad (1.11)$$

Η ανυσματική αναπαράσταση των διαμορφωμένων σημάτων είναι:

$$s_m = [s_{m1} \ s_{m2}] = \left[\sqrt{\frac{E_g}{2}} A_{m1}, \sqrt{\frac{E_g}{2}} A_{m2} \right] = \left[\sqrt{\frac{E_g}{2}} V_m \cos(\theta_m), \sqrt{\frac{E_g}{2}} V_m \sin(\theta_m) \right] = \left[\sqrt{E_{\min}} a_m, \sqrt{E_{\min}} b_m \right] \quad (1.12)$$

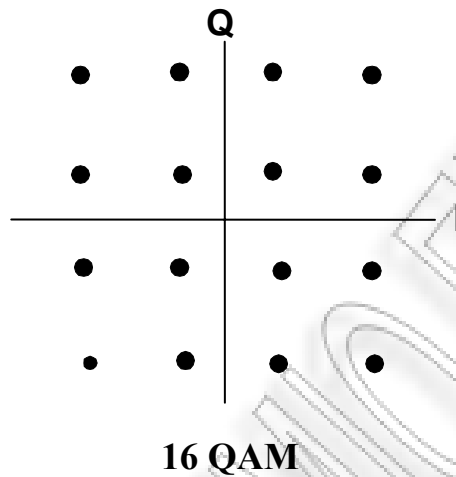
όπου τα (a_m, b_m) είναι στοιχεία ενός $L \times L$ πίνακα που δίνεται ως:

$$\left\{ a_m, b_m \right\} = \begin{bmatrix} (-L+1, L-1) & (-L+3, L-1) & \dots & (L-1, L-1) \\ (-L+1, L-3) & (-L+3, L-3) & \dots & (L-1, L-3) \\ \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot \\ (-L+1, -L+1) & (-L+3, -L+1) & \dots & (L-1, -L+1) \end{bmatrix}$$

όπου $L = \sqrt{M}$.

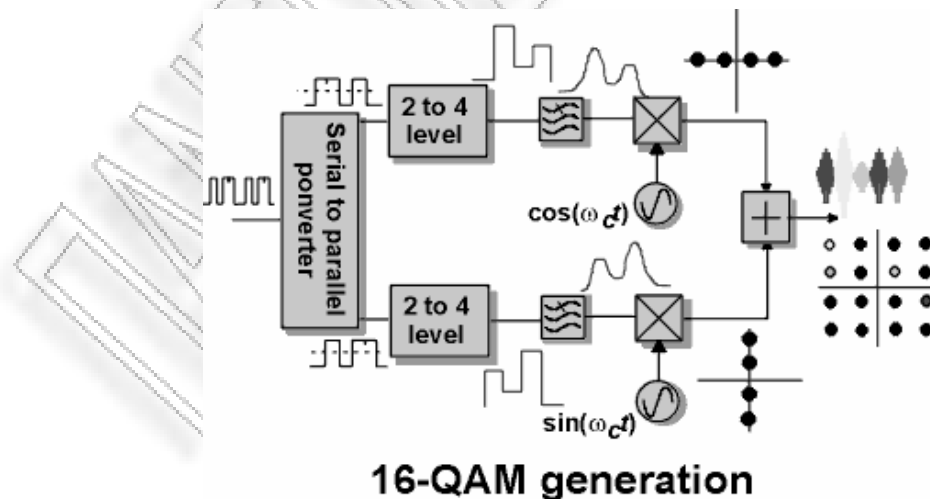
1.4.1 Διαμόρφωση 16 QAM

Είναι φανερό ότι στην ειδική περίπτωση όπου τα πλάτη του σήματος είναι ίσα προκύπτει η διαμόρφωση φάσης M-PSK. Για την περίπτωση όπου έχουμε $M=16$ έχουμε την διαμόρφωση 16 QAM της οποίας το διάγραμμα αστερισμού δεικνύεται στο παρακάτω σχήμα



Σχήμα 1.9 Τετραγωνικό διάγραμμα αστερισμού για 16-QAM

Η δημιουργία ενός σήματος που είναι διαμορφωμένο κατά 16 QAM δεικνύεται στο παρακάτω σχήμα .



Σχήμα 1.10 Δημιουργία σήματος 16-QAM με τετραγωνικό διάγραμμα αστερισμού

Όπως δεικνύεται στο Σχ1.10 το εισερχόμενο bit stream χωρίζεται σε δύο παράλληλα μέρη με ρυθμό $R_b/2$ το καθένα. Στη συνέχεια, για 16-QAM, κάθε 2 bits στο I και 2 bits στο Q μετατρέπονται σε $4 = 2^2$ διαφορετικά πλάτη, έστω $[-3, -1, 1, 3]$ (ανάλογα με το συνδυασμό 2 bit). Αυτή η διαδικασία λαβαίνει μέρος και στο inphase και στο quadrature μέρος. Κατά το άθροισμά τους, παίρνουμε τα εξής σύμβολα μετάδοσης, τα οποία δίνονται παρακάτω από τις προβολές τους (a_m, b_m) στα I και Q:

$$\{a_m, b_m\} = \begin{bmatrix} (-3, 3) & (-1, 3) & (1, 3) & (3, 3) \\ (-3, 1) & (-1, 1) & (1, 1) & (3, 1) \\ (-3, -1) & (-1, -1) & (1, -1) & (3, -1) \\ (-3, -3) & (-1, -3) & (1, -3) & (3, -3) \end{bmatrix}$$

Η ανίχνευση του κάθε σημείου επιτυγχάνεται με τον υπολογισμό τόσο της φάσης του όσο και της ενέργειας του (δηλαδή του πλάτους του). Η πιθανή περιστροφή του ως προς το κέντρο των αξόνων θα είχε επιπτώσεις τόσο στη φάση όσο και στο πλάτος του κάθε σημείου. Αυτό συμβαίνει εάν κατά την ανίχνευση δεν προκύψει σωστή εκτίμηση της φάσης του φέροντος με αποτέλεσμα τον λανθασμένο προσανατολισμό των λαμβανόμενων συμβόλων.

Η Ευκλείδεια απόσταση μεταξύ δύο σημείων στο διάγραμμα αστερισμού M-QAM είναι:

$$d_{mn}^{(e)} = |s_m - s_n| = \sqrt{E_{\min} [(a_m - a_n)^2 + (b_m - b_n)^2]} \quad (1.13)$$

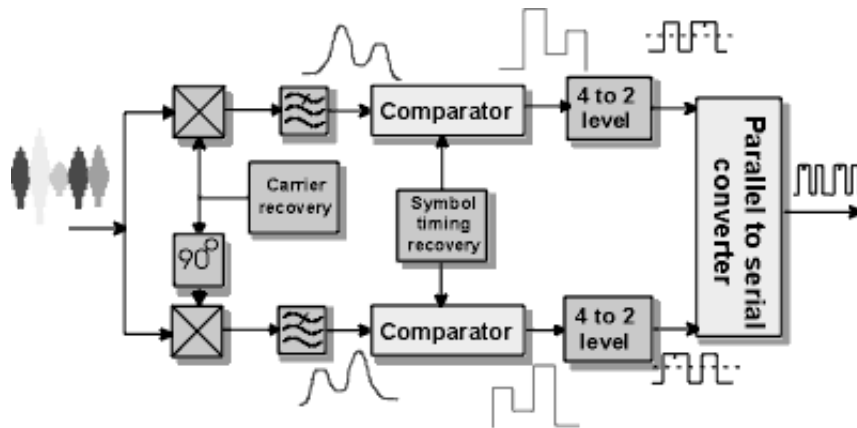
Στην ειδική περίπτωση του τετραγωνικού διαγράμματος αστερισμού όπου τα πλάτη του σήματος έχουν διακριτές τιμές από το σύνολο

$$\{(2m-1-M)E_{\min} \quad m=1, 2, \dots, M\}$$

Η Ευκλείδεια απόσταση μεταξύ δύο γειτονικών σημείων είναι:

$$d_{mn}^{(e)} = \sqrt{2E_{\min}}$$

Κατά τη διαδικασία της λήψης πραγματοποιείται η αντίστροφη διαδικασία όπου το λαμβανόμενο σήμα διαχωρίζεται στους δύο φορείς, οι οποίοι απέχουν μεταξύ τους κατά 90° .



16-QAM detection

Σχήμα 1.11 Ανίχνευση σήματος 16-QAM με τετραγωνικό διάγραμμα αστερισμού

1.5 Σύγκριση των τριών παραπάνω διαμορφώσεων

Και οι τρεις παραπάνω διαμορφώσεις είναι μορφές της βασικής M-PSK. Τα γενικά στοιχεία που χαρακτηρίζουν τη διαμόρφωση M-PSK είναι ότι :

- Στο πεδίο του χρόνου όλα τα σήματα έχουν το ίδιο πλάτος αλλά διαφορετική φάση.
- Όλα τα M σήματα έχουν την ίδια ενέργεια.
- Ο σχηματισμός των M σημάτων απεικονίζεται σε κύκλο με M σημεία που σχηματίζουν μεταξύ τους τόξα ίσα με $2\pi / M$ ακτίνια.
- Διαφέρουν μόνο στη φάση (τα δεδομένα κωδικοποιούνται στη φάση του μεταδιδόμενου σήματος).
- Κάθε φάση διαφέρει κατά $2\pi / M$ ακτίνια (φαίνεται από τον αστερισμό που περιγράφηκε προηγουμένως).

Εάν η πληροφορία που θα μεταδοθεί αποτελείται από μία δυαδική ακολουθία με διάρκεια bit T_b , τότε το εύρος ζώνης που απαιτείται για να μεταδοθεί αυτή η πληροφορία με χρήση BPSK είναι αντιστρόφως ανάλογη του T_b . Αν χρησιμοποιήσουμε M-ary PSK με $M = 2^k$ σύμβολα, τότε $T_s = k \cdot T_b$, και το εύρος ζώνης που απαιτείται είναι αντιστρόφως ανάλογο του $k \cdot T_b$. Έτσι με τη χρήση της M-ary PSK μειώνεται το εύρος ζώνης μετάδοσης κατά ένα συντελεστή k σε σχέση με τη δυαδική σηματοδότηση PSK.

Κατ' επέκταση η φασματική απόδοση, όπως ορίστηκε παραπάνω, είναι:

$$\frac{R_b}{BW} = \frac{R_b}{2R_b} = \frac{k}{2} \text{ bit / sec / Hz}$$

Πρωταρχικός σκοπός μίας φασματικά αποδοτικής διαμόρφωσης είναι η μεγιστοποίηση της απόδοσης εύρους ζώνης (bandwidth efficiency), που ορίζεται ως ο λόγος του ρυθμού δεδομένων εκπομπής προς το εύρος ζώνης του εκπεμπόμενου σήματος (σε μονάδες bits ανά δευτερόλεπτο ανά Hertz), για μια καθορισμένη πιθανότητα σφάλματος συμβόλων. Ένας δευτερεύων σκοπός είναι να επιτευχθεί αυτή η απόδοση εύρους ζώνης με την ελάχιστη κατανάλωση μέσης ισχύος σήματος ή ισοδύναμα, για δίαυλο που διαταράσσεται από προσθετικό θόρυβο Gauss, να επιτευχθεί με την ελάχιστη μέση τιμή λόγου σήματος προς θόρυβο.

Έτσι από την άποψη του εκμετάλλευσης του φάσματος η 16QAM είναι η πιο αποτελεσματική, με λιγότερο την QPSK και τελευταία την πιο απλή μορφή την BPSK. Αντίθετα όσο αφορά την επίδραση του θορύβου και το ISI η BPSK είναι η πιο ανθεκτική, λόγω του ότι απαιτείται μεγάλη παρεμβολή ώστε ο δέκτης να μην έχει την δυνατότητα να ανακτήσει τα σωστά σύμβολα, η QPSK είναι λιγότερο ανθεκτική, ενώ η 16 QAM είναι η πιο «ευάλωτη» στο θόρυβο λόγω του γεγονότος ότι τα σύμβολα απέχουν πολύ μικρή απόσταση με αποτέλεσμα η επίδραση του θορύβου να καθιστά μη ασφαλή την ανάκτηση τους στον δέκτη.

ΚΕΦΑΛΑΙΟ 2

CHANNEL CODING

2.1 Γενικά

Με τον όρο «channel coding» αναφερόμαστε σ' εκείνον τον μετασχηματισμό που γίνεται στο σήμα , ο οποίος εκμεταλλεύεται τα χαρακτηριστικά του σήματος με σκοπό να μειωθεί η επίδραση των στοιχείων που αλλοιώνουν το σήμα , όπως είναι ο θόρυβος , η παρεμβολή και η εξασθένηση και έτσι να έχουμε βελτίωση της απόδοσης της επικοινωνίας.

Η ραγδαία ανάπτυξη της χρήσης VLSI και η επίτευξη υψηλών ταχυτήτων επεξεργασίας , οδήγησαν στην χρήση τέτοιων τεχνικών , οι οποίες μπορούν να παρέχουν έως και 10db βελτιωμένης απόδοσης σήματος με οικονομικότερο και ασφαλέστερο τρόπο , απ' ότι εάν χρησιμοποιούνταν άλλες τεχνικές , όπως υψηλότερες ισχύς εκπομπής ή μεγαλύτερες κεραίες.

Η τεχνική του «channel coding» χωρίζεται σε δύο τομείς έρευνας .Ο ένας αφορά την «κωδικοποίηση της κυματομορφής» (waveform or signal coding) , και η άλλη αυτή των δομημένων ακολουθιών (structured sequences).

Η κωδικοποίηση κυματομορφής , έχει να κάνει με τον μετασχηματισμό του σήματος σε καλύτερη κυματομορφή , ώστε κατά την διαδικασία της ανίχνευσης να προκύψουν όσο το δυνατόν λιγότερα λάθη. Η δεύτερη τεχνική , αυτή της «δομημένης ακολουθίας» έχει να κάνει με τον μετασχηματισμό της ακολουθίας των δεδομένων , προσθέτοντας επιπλέον bits , σ' αυτά της αρχικής ακολουθίας . Αυτά τα επιπλέον bits που τοποθετούνται παρέχουν την δυνατότητα όχι μόνον να ανιχνευθούν τα λάθη που εισέρχονται κατά την μετάδοση , αλλά και να διορθωθούν.

Στο παρόν κεφάλαιο θα αναπτύξουμε αυτήν την τεχνική . Υπάρχουν τρία διαφορετικές τεχνικές οι οποίες έχουν να κάνουν με την μορφή του κώδικα που χρησιμοποιείται. Υπάρχουν τρεις μεγάλες κατηγορίες κωδίκων :

- Block Codes
- Convolution Codes
- Turbo Codes

Στο παρόν κεφάλαιο θα ασχοληθούμε αναλυτικά με τους Convolution Codes

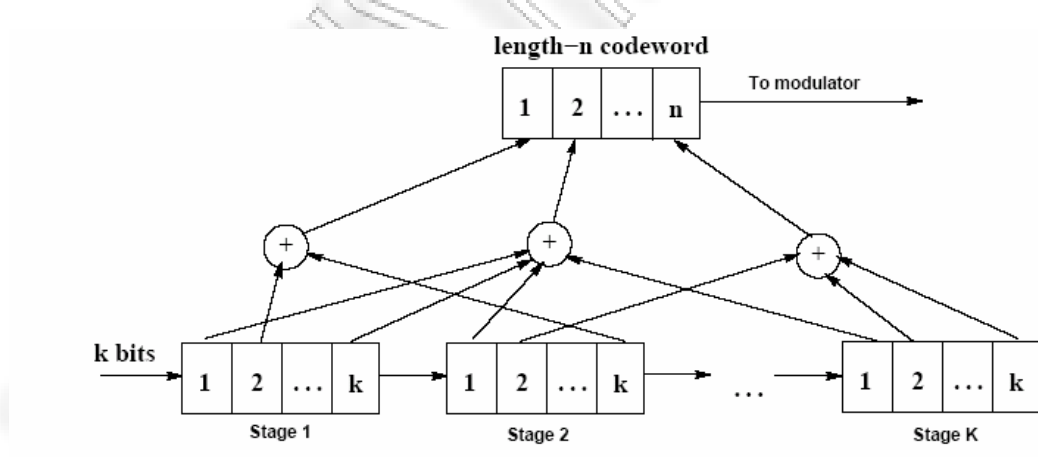
2.2 Convolutional Codes

Ένας convolutional code δημιουργεί κωδικοποιημένα σύμβολα εισάγοντας τα bits πληροφορίας μέσα σ'ένα γραμμικό καταχωρητή πεπερασμένων καταστάσεων (linear finite-state shift register) ως δεικνύεται στο Σχήμα 2.1.

Ο καταχωρητής ολίσθησης (shift register) αποτελείται από K –επίπεδα με k Bits για κάθε επίπεδο. Υπάρχουν n δυαδικοί επιπλέον τελεστές με εισόδους απ' όλα τα K επίπεδα. Αυτοί οι τελεστές παράγουν μία codeword μήκους n για κάθε ακολουθία εισόδου k bits.

Ειδικότερα, η δυαδική ακολουθία εισόδου μεταθέτει σε κάθε επίπεδο του καταχωρητή k bits τη φορά, και κάθε μία από αυτές τις μεταθέσεις παράγει μία κωδικοποιημένη ακολουθία μήκους n . Ο ρυθμός του κώδικα, το code rate, είναι ίσο με $R_c = k/n$. Ο αριθμός των καταστάσεων k του καταχωρητή καλείται **constraint length** του καταχωρητή. Από το σχήμα 2.1 είναι φανερό ότι το μήκος n της codeword εξαρτάται από τα kK εισερχόμενα bits, σε αντίθεση με τους block codes των οποίων οι codewords εξαρτώνται μόνον από τα k εισερχόμενα Bits.

Έτσι ένα σημαντικό χαρακτηριστικό των συνελκτικών κωδίκων, είναι ότι ο encoder έχει μνήμη, από τη στιγμή που στη διαδικασία παραγωγής των codewords παίρνουν μέρος όχι μόνον τα νέα εισερχόμενα bits, αλλά και μερικά από τα προηγούμενα.



Σχήμα 2.1 Convolutional Encoder.

Όταν μία codeword μήκους n δημιουργείται από έναν convolutional encoder, αυτή η codeword εξαρτάται και από τα k εισερχόμενα bits στη πρώτη βαθμίδα (stage) του καταχωρητή, καθώς επίσης και από την κατάσταση του encoder, η οποία καθορίζεται από τα περιεχόμενα των υπολοίπων $K-1$ βαθμίδων του καταχωρητή.

Για να μπορέσουμε να χαρακτηρίσουμε έναν συνελκτικό κώδικα , θα πρέπει να περιγράψουμε πως η δημιουργία της codeword εξαρτάται τόσο από τα k εισερχόμενα bits, όσο και από την κατάσταση του encoder, η οποία έχει 2^k-1 πιθανές τιμές.

Υπάρχουν διάφοροι τρόποι για να περιγραφούν οι συνελκτικοί κώδικες, όπως το διάγραμμα δένδρου (tree diagram), το διάγραμμα κατάστασης (state diagram) , και το trellis διάγραμμα. Το διάγραμμα δένδρου παριστάνει τον encoder σε σχήμα δένδρου , όπου κάθε κλάδος αναπαριστά μία διαφορετική κατάσταση του encoder, και την αντίστοιχη έξοδο της. Το διάγραμμα κατάστασης είναι ένα γράφημα που δείχνει τις διαφορετικές καταστάσεις του encoder ,και τις πιθανές καταστάσεις μετάβασης με τις αντίστοιχες εξόδους. Ένα trellis διάγραμμα χρησιμοποιεί το γεγονός ότι στο διάγραμμα δένδρου υπάρχει επαναληπτικότητα όταν ο αριθμός των βαθμίδων στο δένδρο υπερβαίνει το constraint length του κώδικα. Έτσι το διάγραμμα trellis , απλοποιεί το διάγραμμα δένδρου συγχωνεύοντας κόμβους που αντιστοιχούν στην ίδια κατάσταση.

Στο παρόν κεφάλαιο θα περιγράψουμε το διάγραμμα trellis , δεδομένου ότι είναι από τις πιο διαδεδομένες μορφές αναπαράστασης.

2.2.1 Διάγραμμα trellis

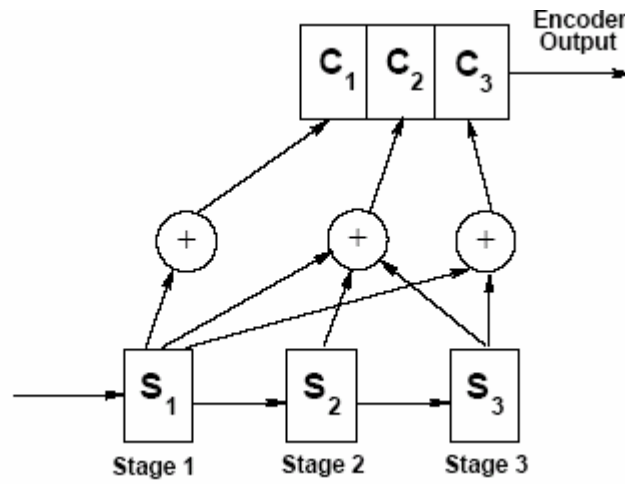
Για τον encoder του σχήματος 2.2 έχουμε $n=3$, $k=1$ και $K=3$. Παρατηρούμε ότι σ' αυτόν τον encoder , ένα bit κάθε φορά μεταφέρεται στη βαθμίδα 1 από τις τρεις του encoder. Τη δεδομένη στιγμή t δηλώνουμε το bit της βαθμίδας 1 ως S_i . Οι τρεις βαθμίδες του καταχωρητή χρησιμοποιούνται για να δημιουργήσουν μία codeword , μήκους 3 , C_1 , C_2 , C_3 , όπου

$$C_1 = S_1 + S_2, \quad C_2 = S_1 + S_2 + S_3, \quad \text{και} \quad C_3 = S_3.$$

Μία ακολουθία U bit μεταφέρεται μέσα στον encoder και δημιουργεί μία ακολουθία κωδικοποιημένων συμβόλων , η οποία δηλώνεται ως C . Παρατηρούμε ότι τα κωδικοποιημένα σύμβολα που συμβολίζονται με C_3 είναι ουσιαστικά τα αρχικά bits πληροφορίας. Όπως συμβαίνει και με τους block codes , σε μία codeword όπου τα κωδικοποιημένα σύμβολα αντιστοιχούν στα αρχικά bits πληροφορίας , τότε έχουμε έναν systematic code.

Ορίζουμε την κατάσταση του encoder $S = S_2S_3$, τα περιεχόμενα των δύο τελευταίων βαθμίδων του encoder , και υπάρχουν $2^2 = 4$ πιθανές τιμές . Για να περιγράψουμε τον encoder , θα πρέπει να δείξουμε για κάθε εισερχόμενο bit και για κάθε πιθανή

κατάσταση του , ποια θα είναι η έξοδος του , και πως κάθε νέα είσοδος ενός νέου bit , αλλάζει την κατάσταση του encoder για το επόμενο bit εισόδου.



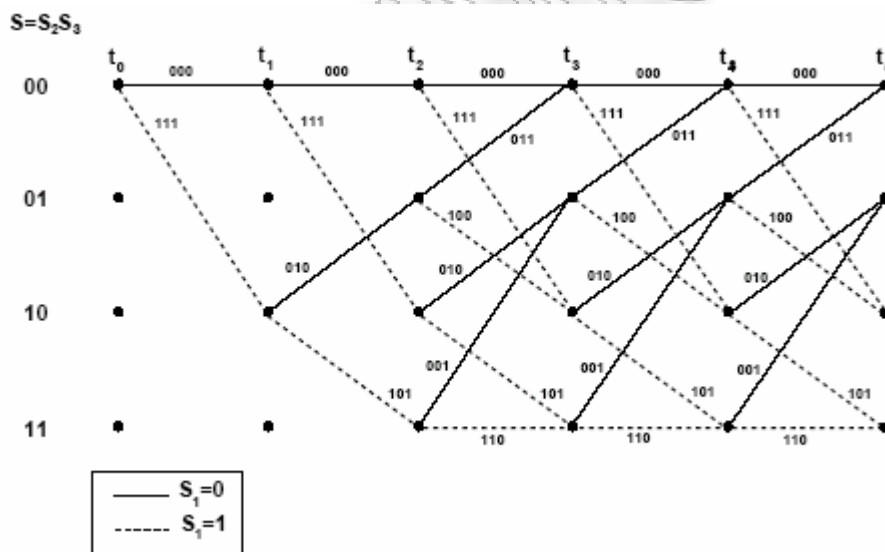
Σχήμα 2.2 Convolutional Encoder ($n = 3, k = 1, K = 3$).

Το διάγραμμα trellis για τον encoder του παραπάνω σχήματος δεικνύεται στο Σχήμα 2.3 . Οι σταθερές γραμμές δεικνύουν την κατάσταση μετάβασης όταν ένα bit “0” εισέρχεται στη βαθμίδα 1 του encoder , ενώ οι διακεκομμένες γραμμές δεικνύουν την αντίστοιχη κατάσταση για την περίπτωση που το bit είναι “1”. Για παράδειγμα , ξεκινώντας από την κατάσταση $S = 00$, εάν bit = 0 είναι το εισερχόμενο στη βαθμίδα 1 , τότε η νέα κατάσταση θα παραμείνει ως είχε δηλαδή $S = 00$. Από την άλλη μεριά , , εάν το εισερχόμενο bit στη βαθμίδα 1 είναι “1” , τότε μετά τις αλλαγές κατάστασης του καταχωρητή, η νέα κατάσταση θα γίνει $S = 10$ (το “1” της βαθμίδας 1 μεταφέρεται στη βαθμίδα 2 , και το “0” της βαθμίδας 2 μεταφέρεται στη βαθμίδα 3 , οπότε η νέα κατάσταση είναι η $S = S_2S_3 = 10$). Η έξοδος του encoder η οποία αντιστοιχεί σε μία δεδομένη κατάσταση του S, και με είσοδο S_1 , σημειώνεται δίπλα στις γραμμές μετάβασης (Σχήμα 2.3) . Αυτή η έξοδος , είναι η έξοδος του encoder , η οποία είναι αποτέλεσμα των προσθετικών λειτουργιών στα bits S_1, S_2 και S_3 σε κάθε βαθμίδα του encoder.

Για παράδειγμα , εάν $S = 00$ και $S_1 = 1$ τότε η έξοδος του encoder είναι $C_1C_2C_3$ με $C_1 = S_1 + S_2 = 1$, $C_2 = S_1 + S_2 + S_3 = 1$, και $C_3 = S_3 = 0$. Αυτή η έξοδος 110 δεικνύετε δίπλα από την διακεκομμένη γραμμή μετάβασης από τη κατάσταση $S = 00$ στη κατάσταση $S = 10$ του Σχήματος 2.3 .

Σημειώνετε ότι η έξοδος του encoder για $S_1 = 0$ και $S = 00$ είναι πάντα οι μηδενικές codeword (όλα τα bit=0), ανεξάρτητα από τις επιπρόσθετες λειτουργίες οι οποίες σχηματίζουν την codeword $C_1C_2C_3$, από τη στιγμή που το άθροισμα οποιοδήποτε μηδενικών είναι 0. Το τμήμα του διαγράμματος trellis μεταξύ των χρονικών στιγμών t_i και t_{i+1} καλείται ο i^{th} κλάδος του trellis. Στο σχήμα 2.3, φαίνεται ότι η αρχική κατάσταση τη στιγμή $t=0$ είναι 0 για όλες τις βαθμίδες. Η μόνιμη κατάσταση (steady state) ορίζεται ως το σημείο όπου όλες οι καταστάσεις μπορούν να έχουν εισοδο από μία από τις δύο προηγούμενες καταστάσεις, κι αυτό συμβαίνει τη χρονική στιγμή t_3 . Από τη στιγμή που αυτή η χρονική στιγμή έχει έρθει, το trellis επαναλαμβάνει τον εαυτό του σε κάθε χρονικό διάστημα. Είναι επίσης φανερό ότι στην μόνιμη κατάσταση, κάθε κατάσταση μεταβαίνει σε μία από τις δύο πιθανές νέες καταστάσεις.

Γενικά, η δομή trellis, ξεκινά τη χρονική στιγμή $t=0$ με όλες τις καταστάσεις 0, και φθάνει την κατάσταση σταθεροποίησης τη χρονική στιγμή $t=k$.



Σχήμα 2.3 Διάγραμμα trellis

Για γενικές τιμές των k και K , το διάγραμμα trellis θα έχει 2^{K-1} , καταστάσεις, όπου η κάθε κατάσταση θα έχει 2^k διαδρομές που θα εισέρχονται στο κάθε κόμβο, και αντίστοιχα 2^k διαδρομές που θα εξέρχονται από τον κόμβο. Έτσι, ο αριθμός των διαδρομών μέσα στο διάγραμμα, αυξάνεται εκθετικά με τα k, K και το μήκος της διαδρομής trellis.

2.2.2 Αποκωδικοποίηση Μέγιστης Πιθανότητας (Maximum Likelihood Decoding)

Ο convolutional code που δημιουργείται από έναν καταχωρητή ολίσθησης πεπερασμένης κατάστασης είναι βασικά μία μηχανή πεπερασμένων καταστάσεων. Αντίθετα από έναν (n, k) block code, όπου η μέγιστη πιθανότητα ανίχνευσης συνεπάγεται την εύρεση της μήκους n της codeword, η μέγιστη πιθανότητα ανίχνευσης ενός convolutional code συνεπάγεται την εύρεση της πιο πιθανής ακολουθίας των κωδικοποιημένων συμβόλων C δοθέντος της λαμβανόμενης ακολουθίας συμβόλων η οποία δηλώνεται με R . Ειδικότερα, για την λαμβανόμενη ακολουθία R , ο αποκωδικοποιητής αποφασίζει ότι η κωδικοποιημένη ακολουθία συμβόλων C^* μεταδόθηκε εάν

$$p(R/C^*) \geq p(R/C) \quad \forall C \quad (2.1)$$

Εφόσον κάθε πιθανή ακολουθία C αντιστοιχεί σε μία διαδρομή του διαγράμματος trellis του κώδικα, η μέγιστη πιθανότητα αποκωδικοποίησης αντιστοιχεί στην εύρεση της διαδρομής της μέγιστης πιθανότητας του διαγράμματος.

Για ένα AWGN κανάλι, ο θόρυβος επηρεάζει κάθε κωδικοποιημένο σύμβολο ανεξάρτητα. Έτσι, για έναν convolution code ρυθμού (code rate) $1/n$, μπορούμε να εκφράσουμε την πιθανότητα της 2.1 ως εξής :

$$p(R/C) = \prod_{i=0}^{\infty} p(R_i / C_i) = \prod_{i=0}^{\infty} \prod_{j=1}^n p(R_{ij} / C_{ij}) \quad (2.2)$$

όπου C_i είναι το τμήμα της ακολουθίας του κώδικα C που αντιστοιχεί στο i th κλάδο του trellis, R_i είναι το τμήμα του λαμβανόμενου κώδικα ακολουθίας R που αντιστοιχεί στο i th κλάδο του trellis, C_{ij} είναι το j th κωδικοποιημένο σύμβολο που αντιστοιχεί στο C_i και R_{ij} είναι το j th λαμβανόμενο κωδικοποιημένο σύμβολο που αντιστοιχεί στο R_i . Η συνάρτηση log της παραπάνω πιθανότητας ορίζεται ως :

$$\log p(R/C) = \sum_{i=0}^{\infty} \log p(R_i / C_i) = \sum_{i=0}^{\infty} \sum_{j=1}^n \log p(R_{ij} / C_{ij}) \quad (2.3)$$

Η έκφραση :

$$B_i = \sum_{j=1}^n \log p(R_{ij} / C_{ij}) \quad (2.4)$$

ονομάζεται **branch metric** και δείχνει το στοιχείο της 2.3 το σχετιζόμενο με το i th κλάδο του trellis. Η ακολουθία ή η διαδρομή η οποία μεγιστοποιεί την συνάρτηση πιθανότητας, μεγιστοποιεί επίσης και την εκθετική συνάρτηση πιθανότητας, εφόσον

η εκθετική συνάρτηση είναι πάντα αυξητική . Εν τούτοις , είναι υπολογιστικά πιο βολικό για τον αποκωδικοποιητή να χρησιμοποιήσει την εκθετική συνάρτηση πιθανότητας λόγω του ότι χρησιμοποιεί άθροισμα και όχι παράγωγο. Η εκθετική συνάρτηση πιθανότητας η συνδεόμενη με μία δεδομένη διαδρομή μέσα στο trellis καλείται επίσης και path metric η οποία από την 2.3 , είναι ίση με το άθροισμα των branch metrics κατά μήκος κάθε κλάδου της διαδρομής. Η διαδρομή με το μέγιστο path metric αντιστοιχεί στη διαδρομή της μέγιστης πιθανότητας.

Ο αποκωδικοποιητής μπορεί να χρησιμοποιήσει είτε hard decision είτε soft decision για την έκφραση $\log p(R_{ij} | C_{ij})$. Για hard decision αποκωδικοποίηση το R_{ij} αποκωδικοποιείται ως 1 ή 0. Η πιθανότητα της hard decision αποκωδικοποίησης εξαρτάται από τη διαμόρφωση και την δηλούμενη πιθανότητα p . Εάν οι ακολουθίες \mathbf{R} και \mathbf{C} είναι μήκους L bits και διαφέρουν σε d σημεία (π.χ ο Hamming τους έχει απόσταση d), τότε

$$p(\mathfrak{R} / C) = p^d (1-p)^{L-d} \quad \text{και}$$

$$\log p(\mathfrak{R} / C) = -d \log \frac{1-p}{p} + L \log(1-p) \quad (2.5)$$

Δεδομένου ότι $p < .5$, 2.5 γίνεται ελάχιστη όταν η απόσταση d ελαχιστοποιείται . Ετσι η κωδικοποιημένη ακολουθία \mathbf{C} με την ελάχιστη Hamming απόσταση από την λαμβανόμενη ακολουθία \mathbf{R} αντιστοιχεί στην μέγιστη πιθανή ακολουθία.

Στη περίπτωση της αποκωδικοποίησης με soft decision οι τιμές των λαμβανομένων κωδικοποιημένων συμβόλων (R_{ij}) χρησιμοποιούνται ως έχουν απ' ευθείας στον αποκωδικοποιητή , και όχι κβαντισμένες σε 1 ή 0. Για παράδειγμα εάν το C_{ij} μεταδίδεται με διαμόρφωση BPSK μέσα από ένα AWGN κανάλι τότε

$$\mathfrak{R}_{ij} = \sqrt{E_C} (2C_{ij} - 1) + n_{ij} \quad (2.6)$$

όπου $E_C = k/n * E_b$ είναι η ενέργεια ανά κωδικοποιημένο σύμβολο και το n_{ij} δηλώνει τον Gaussian θόρυβο με μέση τιμή =0 και διασπορά $\sigma^2 = .5N_0$. Ετσι,

$$p(\mathfrak{R}_{ij} / C_{ij}) = \frac{1}{\sqrt{2\pi\sigma}} \exp \left\{ - \frac{(\mathfrak{R}_{ij} - \sqrt{E_C} (2C_{ij} - 1))^2}{2\sigma^2} \right\} \quad (2.7)$$

Μεγιστοποιώντας την συνάρτηση πιθανότητας είναι ισοδύναμο με την επιλογή του C_{ij} το οποίο είναι πλησιέστερο στην Ευκλείδεια απόσταση (Euclidean distance) R_{ij} . Καθορίζοντας ποια ακολουθία \mathbf{C} μεγιστοποιεί την συνάρτηση εκθετικής πιθανότητας (2.3) , και κάθε όρος που είναι κοινός στις δύο διαφορετικές ακολουθίες

C_1 και C_2 μπορεί να απορριφθεί , από τη στιγμή που συνεισφέρει το ίδιο ποσό στο άθροισμα.

Όμοια , μπορούμε να ζυγοσταθμίσουμε όλους τους όρους της 2.3 χωρίς να αλλάξουμε την μεγιστοποίηση της ακολουθίας .Έτσι απορρίπτοντας παράγοντες και όρους στην 2.7 οι οποίοι είναι κοινοί σε κάθε C_{ij} , μπορούμε να αντικαταστήσουμε

με $\sum_{j=1}^n \log p(R_{ij} / C_{ij})$ στην (2.3) με το **equivalent branch metric**

$$\mu_i = \sum_{j=1}^n \mathfrak{R}_{ij} (2C_{ij} - 1) \quad (2.8)$$

και να έχουμε την ίδια μέγιστη πιθανότητα εξόδου .

Τώρα , μπορούμε να δείξουμε τον υπολογισμό του path metric και με hard και soft decisions για έναν convolution code του Σχήματος 2.2 και με διάγραμμα trellis αυτό του σχήματος 2.3 . Για την απλοποίηση του πράγματος , θα θεωρήσουμε μόνον δύο πιθανές διαδρομές ,και θα υπολογίσουμε τις αντίστοιχες πιθανότητες για μία δεδομένη λαμβανόμενη ακολουθία \mathbf{R} . Υποθέτουμε ότι ξεκινάμε τον χρόνο τη στιγμή t_0 με όλες τις καταστάσεις στο 0. Η πρώτη διαδρομή θεωρούμε ότι είναι αυτή με όλα 0 , και αντιστοιχεί σε μηδενικές εισόδους.

Η δεύτερη διαδρομή θεωρούμε ότι ξεκινά στην κατάσταση $S = 00$ τη στιγμή t_0 και μετάβαση στη κατάσταση $S = 10$ τη στιγμή t_1 , μετά στη κατάσταση $S = 01$ τη στιγμή t_2 , και τέλος στη κατάσταση $S = 00$ τη στιγμή t_3 , σημείο το οποίο αυτή η διαδρομή συγχωνεύεται με τις μηδενικές διαδρομές . Από τη στιγμή που οι διαδρομές και τα branch metrics τους τις στιγμές $t < t_0$ and $t \geq t_3$ είναι τα ίδια , η μέγιστη πιθανή διαδρομή αντιστοιχεί στη διαδρομή της οποίας το άθροισμα των branch metrics απ' όλους τους κλάδους είναι το μικρότερο .

Από το σχήμα 2.3 βλέπουμε ότι όλες οι μηδενικές διαδρομές δημιουργούν την κωδικοποιημένη ακολουθία $C_0 = 00000000$ κατά τη διάρκεια των τριών πρώτων κλάδων στο διάγραμμα trellis. Η δεύτερη διαδρομή δημιουργεί την κωδικοποιημένη ακολουθία $C_1 = 110110011$ κατά τη διάρκεια των τριών πρώτων κλάδων στο trellis. Ας θεωρήσουμε πρώτα την αποκωδικοποίηση με hard decision με πιθανότητα λάθους p . Υποθέτουμε ότι η λαμβανόμενη ακολουθία κατά τη διάρκεια αυτών των τριών κλάδων είναι $\mathbf{R} = 100110111$. Σημειώνετε ότι η απόσταση Hamming μεταξύ \mathbf{R} and C_0 είναι 6 ενώ η απόσταση Hamming μεταξύ \mathbf{R} and C_1 είναι 2. Έτσι η πιο πιθανή διαδρομή αντιστοιχεί στην C_1 αφού έχει την μικρότερη απόσταση Hamming με την ακολουθία \mathbf{R} . Το path metric για όλες τις μηδενικές διαδρομές είναι

$$M_0 = \sum_{i=0}^2 \sum_{j=1}^3 \log P(\mathcal{R}_{ij} / C_{ij}) = 6 \log p + 3 \log(1-p) \quad (2.9)$$

ενώ το path metric για την άλλη διαδρομή είναι

$$M_1 = \sum_{i=0}^2 \sum_{j=1}^3 \log P(\mathcal{R}_{ij} / C_{ij}) = 2 \log p + 7 \log(1-p) \quad (2.10)$$

Θεωρώντας ότι $p \ll 1$, τότε έχουμε ότι $M_0 \approx 6 \log p$ and $M_1 \approx 2 \log p$. Έτσι αφού $\log p < 1$, επιβεβαιώνεται ότι η δεύτερη διαδρομή έχει μεγαλύτερο path metric από την πρώτη.

Ας θεωρήσουμε ότι στη δεύτερη περίπτωση έχουμε αποκωδικοποίηση soft decision κατά τη διάρκεια από t_0 έως t_3 . Υποθέτουμε ότι η λαμβανόμενη ακολουθία (πριν την αποδιαμόρφωση) κατά τη διάρκεια αυτών των τριών κλάδων, για $E_c = 1$, είναι $\mathbf{Z} = (.8, -.35, -.15, 1.35, 1.22, -.62, .87, 1.08, .91)$. Το path metric για όλες τις μηδενικές διαδρομές είναι

$$M_0 = \sum_{i=0}^2 \mu_i = \sum_{i=0}^2 \sum_{j=1}^3 R_{ij} (2C_{ij} - 1) = \sum_{i=0}^2 \sum_{j=1}^3 -R_{ij} = -5.11.$$

Το path metric για τη δεύτερη διαδρομή είναι

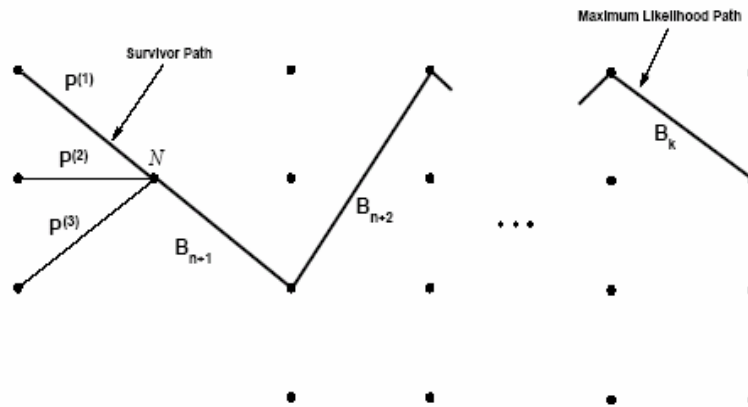
$$M_1 = \sum_{i=0}^2 \sum_{j=1}^3 R_{ij} (2C_{ij} - 1) = 4.91$$

Έτσι, η δεύτερη διαδρομή έχει μεγαλύτερο path metric από την πρώτη. Με σκοπό να καθοριστεί εάν η δεύτερη διαδρομή είναι η μέγιστη πιθανή, θα πρέπει να συγκρίνουμε το path metric της με εκείνο των υπολοίπων διαδρομών του διαγράμματος trellis. Η δυσκολία με την αποκωδικοποίηση της μέγιστης πιθανότητας είναι η πολυπλοκότητα του υπολογισμού της συνάρτησης της εκθετικής πιθανότητας (2.3) η οποία αυξάνεται εκθετικά με την μνήμη του κώδικα, και έτσι ο υπολογισμός αυτός θα πρέπει να γίνει για κάθε πιθανή διαδρομή μέσα στο trellis. Ο αλγόριθμος Viterbi, για τον οποίο θα αναφερθούμε στην επόμενη παράγραφο, μειώνει την πολυπλοκότητα της αποκωδικοποίησης με την μέγιστη πιθανότητα, εκμεταλλευόμενος την δομή του υπολογισμού του path metric.

2.3 Αλγόριθμος Viterbi

Ο αλγόριθμος Viterbi, ανακαλύφθηκε από τον Viterbi το 1967, και μείωσε την πολυπλοκότητα της αποκωδικοποίησης με την μέγιστη πιθανότητα, αφαιρώντας συστηματικά όλες τις διαδρομές με τις οποίες δεν μπορούσε να επιτευχθεί το

μέγιστο path metric. Η βασική πρόταση είναι να εξεταστούν τα μερικά path metrics τα οποία συνδέονται με όλες τις διαδρομές που εισέρχονται σ' ένα δεδομένο κόμβο (Node N) στο trellis. Δεδομένου ότι οι πιθανές διαδρομές μέσα στο trellis που αφήνουν τον κόμβο N είναι οι ίδιες με τις εισερχόμενες, η ολοκληρωμένη διαδρομή με το υψηλότερο path metric η οποία περνάει από τον κόμβο N θα πρέπει να συμπίπτει με τη διαδρομή η οποία έχει το υψηλότερο μερικό path metric μέχρι τον κόμβο N . Αυτό δεικνύεται στο σχήμα 2.4, όπου η διαδρομή 1, διαδρομή 2, και η διαδρομή 3 εισέρχονται στο κόμβο N (depth n) με μερικά path metrics $P^l = \sum_{i=0}^N B_i^l$ με $l=1,2,3$.



Σχήμα 2.4 : Partial Path Metrics on Maximum Likelihood Path

Υποθέτουμε ότι P^l είναι η μεγαλύτερη απ' αυτές τις μερικές path metrics. Η πλήρης διαδρομή με το μεγαλύτερο metric, δεικνύεται με έντονη γραμμή, έχει branch metrics $\{B_k\}$ μετά τον κόμβο N . Η μέγιστη πιθανή διαδρομή ξεκινά από τον κόμβο N , για παράδειγμα η διαδρομή που ξεκινά από τον κόμβο N με το μεγαλύτερο

path metric, έχει μερικό path metric $\sum_{k=n}^{\infty} B_k$. Το συνολικό path metric για τις

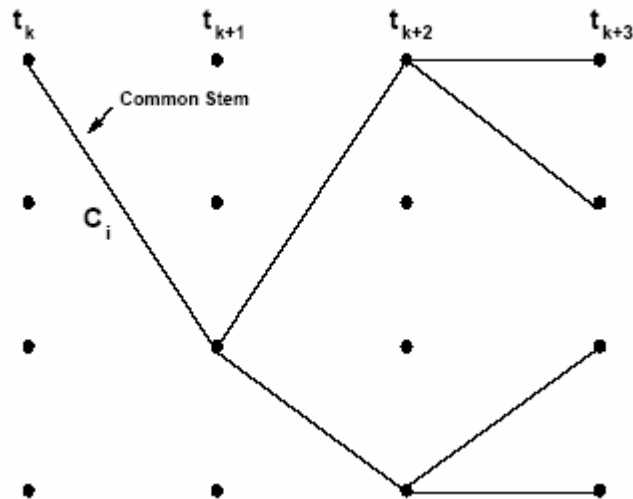
διαδρομές 1,2,3 μέχρι το κόμβο N και η διαδρομή με τη μέγιστη πιθανότητα μετά τον

κόμβο N είναι $P^l + \sum_{k=n}^{\infty} B_k = B_k$, $l=1,2,3$ και έτσι η διαδρομή με το μεγαλύτερο μερικό

path metric P^l μέχρι τον κόμβο N (Path 1 στο παράδειγμα) θα πρέπει να αντιστοιχεί στη διαδρομή με το μεγαλύτερο path metric που περνάει στον κόμβο N .

Ο αλγόριθμος Viterbi εκμεταλλεύεται αυτή τη δομή παραμερίζοντας όλες τις διαδρομές που εισέρχονται σ' ένα δεδομένο κόμβο εκτός από τη διαδρομή με το μεγαλύτερο μερικό path metric μέχρι εκείνο τον κόμβο. Η διαδρομή αυτή που δεν απορρίπτετε ονομάζεται διαδρομή επιβίωσης (**survivor path**).

Για το παράδειγμα του σχήματος 2.4, η διαδρομή 1 είναι η διαδρομή επιβίωσης στο κόμβο N και οι διαδρομές 2 και 3 παραμερίζονται από περαιτέρω επεξεργασία. Για κάθε βαθμίδα στο trellis, υπάρχουν $2^K - 1$ διαδρομές επιβίωσης, μία για κάθε πιθανή κατάσταση του encoder. Ένας κλάδος για μία δεδομένη κατάσταση δεν μπορεί να αποκωδικοποιηθεί μέχρι όλες οι διαδρομές επιβίωσης στις μεταγενέστερες καταστάσεις επικαλυφθούν, μ' αυτό τον κλάδο, ως δεικνύεται στο σχήμα 2.5.



Σχήμα 2.5 Common Stem for All Survivor Paths in the Trellis

Το σχήμα δείχνει τις διαδρομές επιβίωσης τη χρονική στιγμή t_{k+3} . Παρατηρούμε ότι όλες οι διαδρομές επιβίωσης μπορούν να ανιχνευθούν σε μία κοινή ακμή από τη χρονική στιγμή t_k έως την t_{k+1} . Σ' αυτό το σημείο ο αποκωδικοποιητής μπορεί να δώσει στην έξοδο του την codeword C_i τη σχετιζόμενη μ' αυτόν τον κλάδο. Θα πρέπει να σημειωθεί ότι δεν υπάρχει σταθερή καθυστέρηση στην αποκωδικοποίηση που να σχετίζεται με το πόσο πίσω στο trellis υπάρχει η κοινή ακμή (common stem) για ένα δεδομένο σετ από διαδρομές επιβίωσης. Η

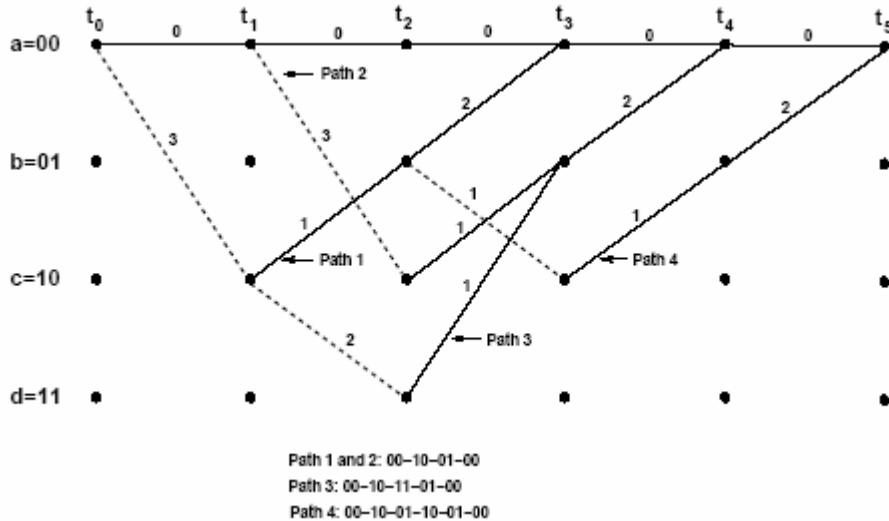
καθυστερήση που υπάρχει εξαρτάται μόνον από τις παραμέτρους k και K , και τις ειδικές ιδιότητες του κώδικα.

Προς αποφυγή της τυχαίας καθυστέρησης αποκωδικοποίησης, ο αλγόριθμος Viterbi τυπικά τροποποιείται με τέτοιο τρόπο ώστε σε μία δεδομένη βαθμίδα στο trellis, ο πιο πιθανός κλάδος, n βαθμίδων πίσω αποφασίζεται βάσει του μερικού path metrics μέχρι εκείνου του σημείου. Με αυτή τη διαφοροποίηση δεν επιτυγχάνεται ακριβώς η αποκωδικοποίηση μέγιστης πιθανότητας, αλλά όταν το n είναι αρκετά μεγάλο ($n \geq 5K$) θεωρείται μία πολύ καλή προσέγγιση.

Ο αλγόριθμος πρέπει να κρατά ίχνη $2^{k(K-1)}$ διαδρομών επιβίωσης και τις αντίστοιχες μετρήσεις τους. Σε κάθε βαθμίδα, 2^k μετρήσεις θα πρέπει να υπολογισθούν για κάθε κόμβο ώστε να καθοριστεί η διαδρομή επιβίωσης, αντιστοίχως των 2^k διαδρομών που εισέρχονται σε κάθε κόμβο. Έτσι, ο αριθμός των υπολογισμών κατά την αποκωδικοποίηση και οι απαιτήσεις της μνήμης για τον αλγόριθμο αυξάνουν εκθετικά με τα k και K . Αυτό σημαίνει ότι στη πράξη για την ανάπτυξη των convolutional codes χρησιμοποιούνται σχετικά μικρές τιμές για k και K .

2.4 Ιδιότητες Απόστασης (Distance Properties)

Όπως με τους block codes, η δυνατότητα για διόρθωση λαθών ενός convolution code εξαρτάται από την απόσταση μεταξύ των ακολουθιών της codeword. Εφόσον οι convolution codes είναι γραμμικοί, η ελάχιστη απόσταση μεταξύ όλων των ακολουθιών της codeword μπορεί να βρεθεί ορίζοντας την ελάχιστη απόσταση από κάθε ακολουθία ή ισοδυνάμως οποιαδήποτε προς σ' όλες τις μηδενικές ακολουθίες ή διαδρομές στο trellis path. Είναι εμφανές ότι η διαδρομή με την μικρότερη απόσταση απ' όλες τις μηδενικές διαδρομές θα αποκλίνει και θα ξαναενώνεται μ' όλες τις μηδενικές διαδρομές, έτσι ώστε οι δύο διαδρομές να συμπίπτουν, εκτός από κάποιου κλάδους. Για να βρούμε αυτήν την ελάχιστη απόσταση διαδρομής, θα πρέπει να λάβουμε υπόψη όλες τις διαδρομές που αποκλίνουν από τις μηδενικές καταστάσεις και μετά ξανασυγχωνεύονται μ' αυτές. Ως παράδειγμα αναφερόμαστε στο σχήμα 2.6, όπου έχουμε σχεδιάσει όλες τις διαδρομές του σχήματος 2.3 μεταξύ των χρονικών στιγμών t_0 και t_5 οι οποίες αποκλίνουν και μετά συγχωνεύονται με όλες τις μηδενικές καταστάσεις.



Σχήμα 2.6 Path Distances to the All-Zero Path

Σημειώνεται ότι η διαδρομή 2 είναι όμοια με τη διαδρομή 1, απλώς μετατοπισμένη στο χρόνο, και ως εκ τούτου δεν λαμβάνεται υπόψη ως ξεχωριστή διαδρομή. Φαίνεται επίσης ότι μπορούμε να εξετάσουμε ένα μεγαλύτερο διάστημα, αλλά οποιοδήποτε διαδρομή η οποία αποκλίνει και ξανασυγχωνεύεται μέσα σ' αυτό το μεγαλύτερο διάστημα θα διέρχεται από τους ίδιους κλάδους (μετατοπισμένους στο χρόνο), και ως εκ τούτου θα έχουν μεγαλύτερα path metrics. Ειδικότερα, βλέπουμε ότι η διαδρομή 4 διέρχεται από τους ίδιους κλάδους όπως και η διαδρομή 1, 00-10-01 και μετά 01-00, πλέον των κλάδων 01-10-01. Έτσι δεν χρειάζεται να λάβουμε υπόψη μεγαλύτερο διάστημα για να βρούμε την ελάχιστη απόσταση διαδρομής. Για κάθε διαδρομή του σχήματος 2.3 σημαίνουμε την απόσταση Hamming της codeword πάνω σε κάθε κλάδο και τις μηδενικές codeword στους αντίστοιχους κλάδους μηδενικής διαδρομής.

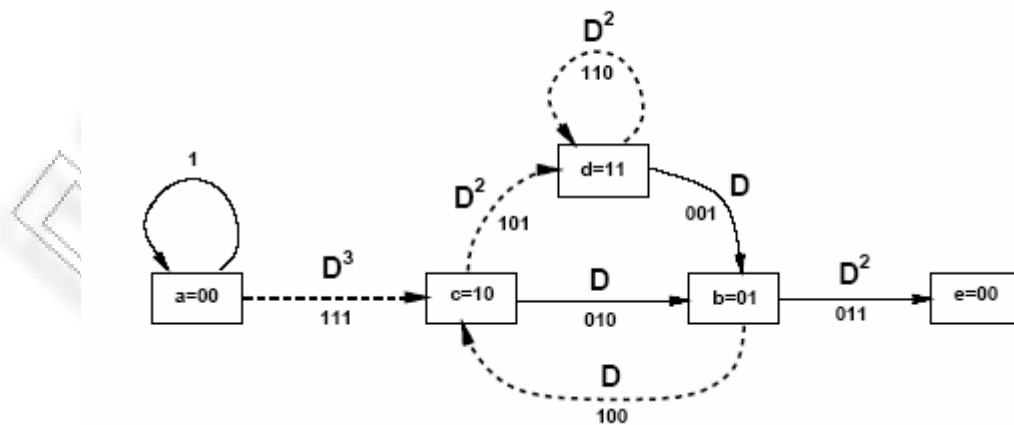
Προσθέτοντας τις αποστάσεις Hamming όλων των κλάδων της κάθε διαδρομής θα δούμε ότι η διαδρομή 1 έχει απόσταση Hamming 6 και οι διαδρομές 3 και 4 έχουν 8. Εφόσον οι διακεκομμένες γραμμές συμβολίζουν είσοδο bit "1", ενώ οι σταθερές bit "0", βλέπουμε ότι η διαδρομή 1 αντιστοιχεί σε εισερχόμενη ακολουθία bit 10000 από τη χρονική στιγμή t_5 , ενώ η διαδρομή 3 αντιστοιχεί σε ακολουθία εισόδου 11000, και η διαδρομή 4 αντιστοιχεί σε είσοδο 10100. Συμπερασματικά, η διαδρομή 1 δίνει ένα bit λάθος, σε σχέση με την μηδενική ακολουθία, και οι διαδρομές 3 και 4 δίνουν 2 bit λάθος.

Ορίζουμε ως **ελάχιστη ελεύθερη απόσταση (minimum free distance)** ενός convolutional code, την οποία και ορίζουμε ως $dfree$, επίσης καλούμενη και ως ελεύθερη απόσταση (free distance), να είναι η ελάχιστη απόσταση Hamming των διαδρομών μέσα στο trellis σε σχέση με όλες τις μηδενικές διαδρομές, και η οποία για το συγκεκριμένο παράδειγμα είναι 6. Η δυνατότητα για διόρθωση λάθους του κώδικα εξάγεται με τον ίδιο τρόπο όπως και για τους block codes, εάν αντικαταστήσουμε το d_{min} από το df , έτσι ο κώδικας μπορεί να διορθώσει t λάθη με :

$$t = \left\lfloor \frac{df - 1}{2} \right\rfloor$$

2.5 Διάγραμμα Κατάστασης και Συναρτήσεις Μεταφοράς

Η συνάρτηση μεταφοράς ενός convolution code χρησιμοποιείται για να χαρακτηρίσει τις διαδρομές οι οποίες αποκλίνουν και ξανασυγχωνεύονται με τις μηδενικές διαδρομές, καθώς επίσης και για να μετρήσει την πιθανότητα του λάθους. Η συνάρτηση αυτή εξάγεται από το διάγραμμα κατάστασης του κώδικα το οποίο αντιπροσωπεύει τις πιθανές μεταβάσεις από όλες τις μηδενικές καταστάσεις σ' όλες τις μηδενικές. Το διάγραμμα κατάστασης για τον κώδικα του σχήματος 2.3 δεικνύεται στο σχήμα 2.7, με όλες τις μηδενικές καταστάσεις $a=00$ να χωρίζονται σ' ένα δεύτερο κόμβο e , ώστε να διευκολυνθεί η αναπαράσταση διαδρομών οι οποίες αρχίζουν και ξεκινούν σ' αυτή τη κατάσταση.



Σχήμα 2.7 State Diagram

Οι μεταβάσεις μεταξύ καταστάσεων οι οποίες σε εισερχόμενο “0” bit φαίνονται με σταθερές γραμμές, ενώ οι μεταβάσεις που οφείλονται σε “1” με διακεκομμένες γραμμές. Οι κλάδοι του διαγράμματος κατάστασης σημαίνονται είτε με $D^0 = 1$, D^1 , ή D^2 , όπου το εκθετικό του D αντιστοιχεί στην απόσταση Hamming μεταξύ της codeword, η οποία δεικνύεται για κάθε μετάβαση κλάδου, ενώ όλες οι μηδενικές codeword αντιστοιχούν σ’ όλες τις μηδενικές διαδρομές. Οι γραμμές που κλείνουν στον ίδιο κόμβο, όπως στον κόμβο a μπορούν να αγνοηθούν από τη στιγμή που δεν συνεισφέρουν στις ιδιότητες απόστασης του κώδικα.

Το διάγραμμα κατάστασης μπορεί να περιγραφεί με τις εξισώσεις κατάστασης για κάθε κατάσταση. Για το παράδειγμα του σχήματος 2.3 έχουμε τις ακόλουθες εξισώσεις καταστάσεις οι οποίες δίνουν τις τέσσερις καταστάσεις :

$$X_c = D^3 X_a + DX_b, \quad X_b = DX_c + DX_d, \quad X_d = D^2 X_c + D^2 X_d, \quad X_e = D^2 X_b. \quad (2.11)$$

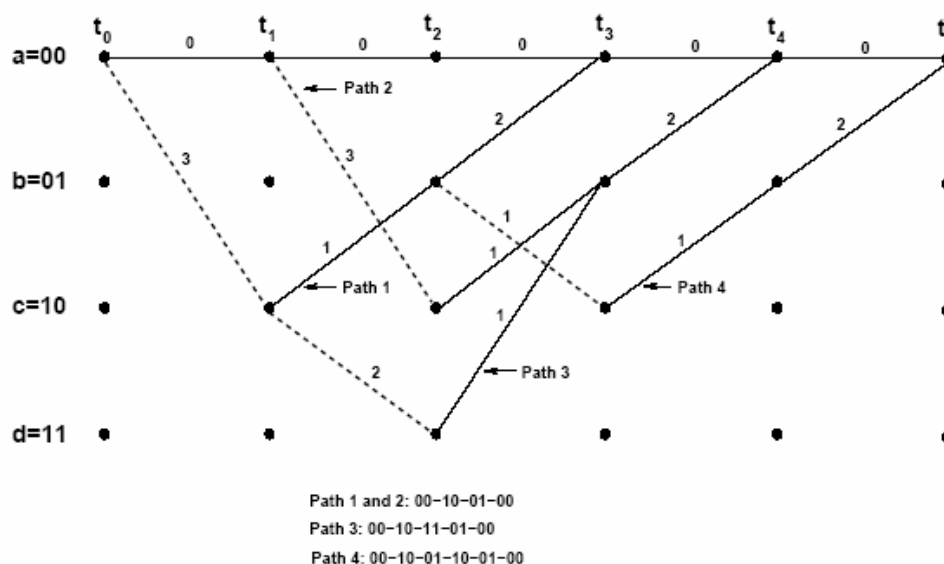
όπου X_a, \dots, X_e είναι «ψεύτικες» μεταβλητές οι οποίες χαρακτηρίζουν τις μερικές διαδρομές. Η συνάρτηση μεταφοράς του κώδικα, η οποία περιγράφει τις διαδρομές από την κατάσταση a στη κατάσταση e, ορίζεται ως $T(D) = X_e/X_a$. Λύνοντας τις εξισώσεις κατάστασης, χρησιμοποιώντας την φόρμουλα Mason, έχουμε την συνάρτηση μεταφοράς ως εξής :

$$T(D) = \sum_{d=d_f}^{\infty} a_d D^d \quad (2.12)$$

όπου a_d είναι ο αριθμός των διαδρομών με απόσταση Hamming d από όλες τις μηδενικές διαδρομές. Όπως ειπώθηκε, η ελάχιστη απόσταση Hamming είναι d_f , και η συνάρτηση μεταφοράς $T(D)$ δείχνει ότι υπάρχουν a_{d_f} διαδρομές με αυτήν την ελάχιστη απόσταση. Λύνοντας τις εξισώσεις κατάστασης (2.11) θα έχουμε την ακόλουθη συνάρτηση μεταφοράς :

$$T(D) = \frac{D^6}{1-2D^2} = D^6 + 2D^8 + 4D^{10} + \dots \quad (2.13)$$

Από την συνάρτηση μεταφοράς βλέπουμε ότι υπάρχει μία διαδρομή με ελάχιστη απόσταση $df = 6$, και δύο διαδρομές με απόσταση Hamming ίσον με 8, γεγονός που συμφωνεί με το σχήμα 2.8.



Σχήμα 2.8 Path Distances to the All-Zero Path

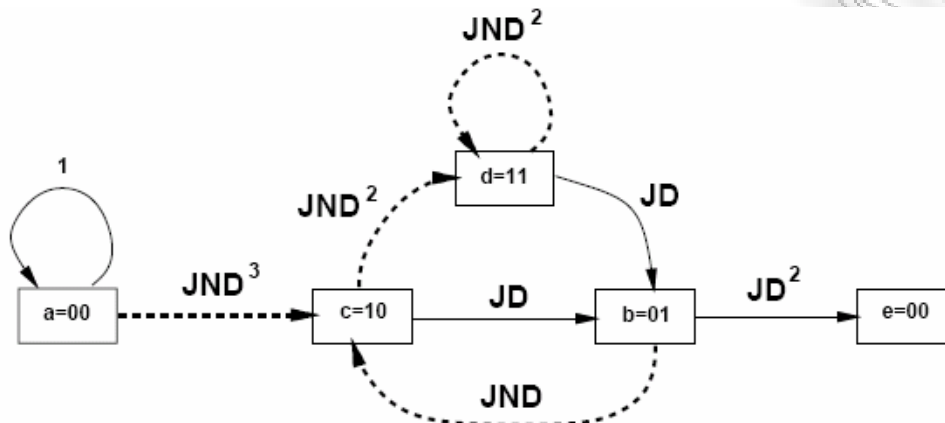
Η συνάρτηση μεταφοράς είναι ένας βολικός τρόπος για καταγραφή όλων των αριθμών και των αντίστοιχων αποστάσεων Hamming όλων των διαδρομών ενός κώδικα, οι οποίες αποκλίνουν και μετά συγχωνεύονται με όλες τις μηδενικές διαδρομές. Ενώ η συνάρτηση μεταφοράς είναι επαρκής για να έχουμε όλες τις αποστάσεις Hamming όλων των διαδρομών που υπάρχουν μέσα σ' ένα διάγραμμα trellis, χρειαζόμαστε ένα πιο λεπτομερειακό χαρακτηριστικό για να υπολογίσουμε την πιθανότητα λάθους bit (error probability) ενός convolution code.

Για το λόγο αυτό εισάγουμε δύο επιπλέον παραμέτρους στην συνάρτηση μεταφοράς, τους N και J . Ο παράγοντας N εισάγεται σ' όλες τις μεταβάσεις των κλάδων που αφορούν το bit "1" ως είσοδο, (διακεκομμένες γραμμές στο σχήμα 2.7). Ο παράγοντας J εισάγεται σε κάθε κλάδο του διαγράμματος κατάστασης, το εκθετικό του οποίου, στη συνάρτηση μεταφοράς ισούται με τον αριθμό των κλάδων για κάθε διαδρομή από τον κόμβο a στο κόμβο e .

Το διάγραμμα επέκτασης που ανταποκρίνεται στο trellis του σχήματος 2.3 δεικνύεται στο σχήμα 2.9

Το διάγραμμα αυτό , μπορεί επίσης να εκφραστεί με τις εξισώσεις κατάστασης. Για το παράδειγμα του σχήματος 2.9 έχουμε

$$X_c = JND^3 X_a + JNDX_b, \quad X_b = JDX_c + JDX_d, \quad X_d = JND^2 X_c + JND^2 X_d, \quad X_e = JD^2 X_b. \quad (2.14)$$



2.9 Διάγραμμα επέκτασης

Όμοια με τον προηγούμενο ορισμό της συνάρτησης μεταφοράς , η συνάρτηση μεταφοράς του διαγράμματος επέκτασης ορίζεται ως $T(D,N, J) = X_e/X_a$, όπου για αυτό το παράδειγμα είναι :

$$T(D, N, J) = \frac{J^3 ND^6}{1 - JND^2(1+J)} = J^3 ND^6 + J^4 N^2 D^8 + J^5 N^2 D^8 + J^5 N^3 D^{10} + \dots \quad (2.15)$$

Ο παράγοντας J παίζει σημαντικό ρόλο όταν πρόκειται να μεταδώσουμε πεπερασμένο μήκος ακολουθίας . Για άπειρο μήκος ακολουθίας τυπική τιμή για το J = 1, και έχουμε

$$T(D, N) = T(D, N, J = 1) \quad (2.16)$$

Η συνάρτηση μεταφοράς του διαγράμματος επέκτασης μας δίνει περισσότερες πληροφορίες για την απόκλιση και επανένωση των διαδρομών. Έτσι , η ελάχιστη απόσταση διαδρομής με απόσταση Hamming ίσον με 6 , είναι μήκους 3 και δίνει ένα bit λάθος, (το εκθετικό του N είναι 1) , μία διαδρομή με απόσταση Hamming ίσον με 8, είναι μήκους 4 και δίνει 2 bit λάθος , και οι άλλες διαδρομές με απόσταση Hamming 8 είναι μήκους 5 και δίνει 2 bit λάθος , γεγονός που συμφωνεί με το σχήμα 2.8. Η συνάρτηση μεταφοράς του διαγράμματος επέκτασης είναι μία βολική

περιγραφή η οποία μας δίνει την απόσταση Hamming το μήκος , και τον αριθμό των λαθών (bit errors) .Αυτή η συνάρτηση είναι πολύ χρήσιμη για να μπορέσουμε να υπολογίσουμε την πιθανότητα λαθών στους convolution codes.

ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΡΡΑΙΑ

ΚΕΦΑΛΑΙΟ 3

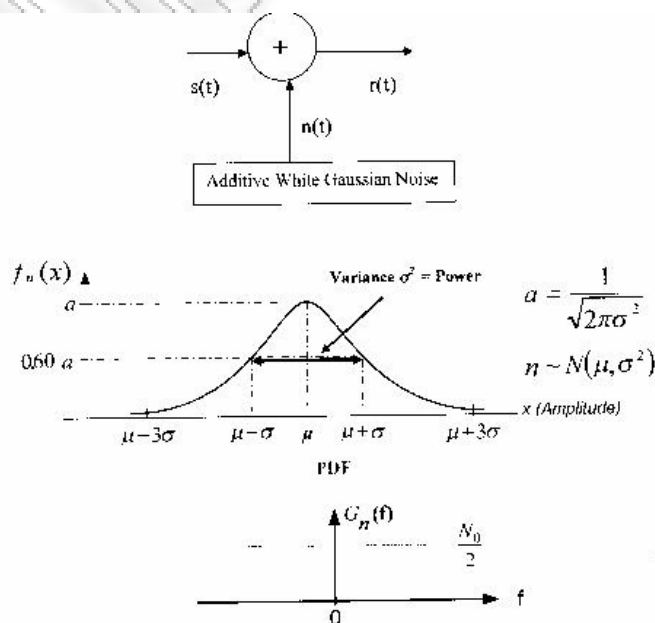
AWGN & FADING

3.1 Γενικά

Ο θόρυβος είναι μία μεταβλητή που επηρεάζει σημαντικά την απόδοση των ψηφιακών συστημάτων μετάδοσης, ειδικά αυτών των κινητών επικοινωνιών. Ο «προσθετικός λευκός θόρυβος», AWGN, ο οποίος είναι σχετικά πιο εύκολο να αντιμετωπιστεί και στις επόμενες παραγράφους αναλύεται σχετικά. Υπάρχει όμως ένας παράγοντας πολύ σημαντικότερος του οποίου η επίδραση μειώνει δραστικά την απόδοση των συστημάτων, και αυτός είναι η εξασθένιση – fading-. Η εξασθένιση ακολουθεί στατιστικό μοντέλο και υπάρχουν τρεις κατανομές που ακολουθούνται για την μοντελοποίηση των διαύλων σε fading, η Rayleigh την οποία και θα αναλύσουμε και θα χρησιμοποιήσουμε για την προσομοίωση, η Rice κατανομή και τέλος Nakagami-m.

3.2 Additive White Gaussian Noise

Το σήμα λήψης στην είσοδο ενός δέκτη είναι παραμορφωμένο από αθροιστικό λευκό Gaussian θόρυβο, με Gaussian συνάρτηση πυκνότητας πιθανότητας (PDF) και φασματική πυκνότητα πιθανότητας (power spectral density, PSD) ίση με $N_0/2$ Watt/Hz (Σχήμα 1).



Σχήμα 3.1 AWGN 2-sided power spectral density of noise

Το σήμα λήψης δίνεται από :

$$r(t) = s(t) + n(t) \quad (3.1)$$

όπου $n(t)$ είναι δείγματα θορύβου AWGN.

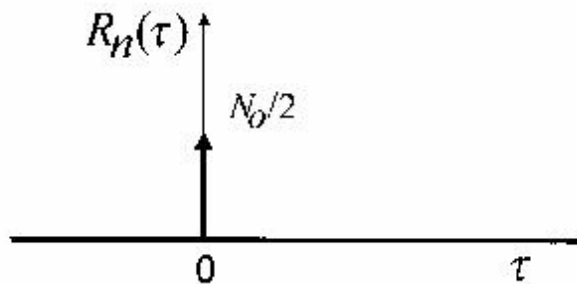
Αυτός ο τύπος θορύβου ονομάζεται Αθροιστικός White Gaussian (AWGN) με PSD που δίνεται ως :

$$G_n(f) = \frac{N_0}{2} \quad [\text{W/Hz}] \quad (3.2)$$

Από την PSD μπορούμε να υπολογίσουμε τη συνάρτηση αυτοσυσχετισμού της διαδικασίας του AWGN ,ως εξής :

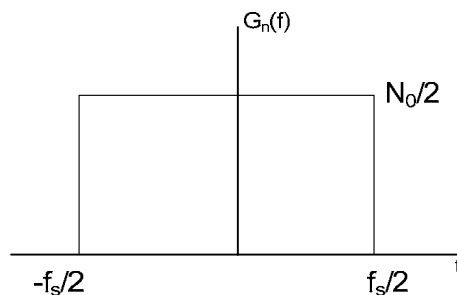
$$R_n(\tau) = E[n(t)n(t+\tau)] = F^{-1}\{G_n(f)\} = \frac{N_0}{2} \delta(\tau) \quad (3.3)$$

Αυτό σημαίνει ότι δύο διαφορετικά δείγματα θορύβου $n(t)$ όσο κοντά και αν είναι στο χρόνο, έχουν μηδενική συσχέτιση .



Σχήμα 3.2 Συνάρτηση συσχέτισμού για δείγματα θορύβου AWGN

Απ' αυτό συμπεραίνουμε ότι uncorrelated δείγματα (με μηδενική συσχέτιση) προέρχονται από PSD η οποία είναι σταθερή για το simulation bandwidth, το οποίο είναι $|f| < f_s/2$. Η διακύμανση (variance) είναι εξ' ορισμού το εμβαδόν της $G_n(f)$ στο φάσμα της προσομοίωσης , όπως φαίνεται στο Σχήμα 3.3



Σχήμα 3.3 PSD για ανεξάρτητα (independent) δείγματα.

Επομένως το noise variance με την PSD , με τη σχέση :

$$\sigma_n^2 = \frac{N_0 f_s}{2} \quad (3.4)$$

ή

$$N_0 = \frac{2\sigma_n^2}{f_s} \quad (3.5)$$

Το πηλίκο σήματος –προς- θόρυβο SNR ορίζεται ως E_s/N_0 όπου f_s , είναι η συχνότητα δειγματοληψίας του διακριτού συστήματος προσομοίωσης. Επομένως :

$$SNR = f_s \frac{E_s}{2\sigma_n^2} \quad (3.6)$$

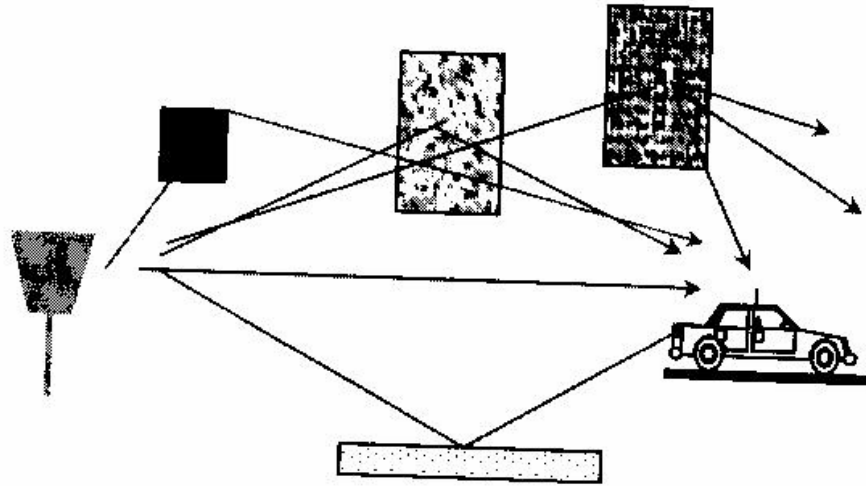
Και το noise standard déviation που πρέπει να εισάγουμε στην προσομοίωση για μία συγκεκριμένη τιμή του SNR είναι :

$$\sigma_n = \sqrt{\frac{f_s E_s}{2SNR}} = \sqrt{f_s \frac{N_0}{2}} \quad (3.7)$$

3.3 Fading

Ο όρος fading χρησιμοποιεί για να περιγράψει τις γρήγορες μεταβολές στο πλάτος του λαμβανόμενου σήματος για μία μικρή χρονική διάρκεια. Το fading απαντάται στα mobile communication channels και προκαλείται από την άθροιση πολλαπλών αντιγράφων του σήματος εκπομπής που φτάνουν σχεδόν ταυτόχρονα στην κεραία του δέκτη από διαφορετικές κατευθύνσεις. Το συνιστάμενο σήμα μεταβάλλεται κατά το πλάτος και τη φάση . Τα χαρακτηριστικά της μεταβολής αυτής εξαρτώνται από διάφορους παράγοντες , όπως το περιβάλλον διάδοσης(επηρεάζει τις σχετικές διαφορές στο propagation time των κυμάτων), το εύρος φάσματος (bandwidth) του εκπεμπόμενου σήματος. Γενικά στη διαδικασία του fading , θεωρούμαι ότι όλα τα συνιστώσα κύματα φτάνουν ταυτόχρονα, οπότε λαμβάνονται ως ένα path του οποίου το πλάτος αυξομειώνεται.

Για παράδειγμα , όπως φαίνεται στο Σχήμα 3.4 , όσο κινείται τι αυτοκίνητο αλλάζουν οι multipath συνιστώσες που φτάνουν στην κεραία του δέκτη λόγω της αλλαγής του περιβάλλοντα χώρου , άρα και των ανακλώμενων κυμάτων που φτάνουν στην κεραία του δέκτη.



Σχήμα 3.4 Multipath περιβάλλον διάδοσης

Έστω ότι το σήμα εκπομπής είναι ένα απλό ημιτονοειδές σήμα στη συχνότητα του φέροντος (μη διαμορφωμένο σήμα στην f_c) :

$$s(t) = A \cos(2\pi f_c t) \quad (3.8)$$

Το λαμβανόμενο σήμα δίνεται ως (αγνοώντας για λίγο τον θόρυβο)

$$r(t) = A \sum_{i=1}^{N_1} a_i \cos(2\pi f_c t + \theta_i) \quad (3.9)$$

όπου

- 1) a_i είναι η εξασθένιση (attenuation) της i -th multipath συνιστώσας
- 2) θ_i είναι η αλλαγή φάσης (phase-shift) της i -th multipath συνιστώσας

Θεωρούμε ότι οι N_1 συνιστώσες φτάνουν σχεδόν ταυτόχρονα , οπότε αυτό που παρατηρείται στο δέκτη είναι μόνο το συνιστάμενο σήμα. Οι όροι a_i και θ_i είναι τυχαίες μεταβλητές (random variables) . Η παραπάνω έκφραση μπορεί να αναλυθεί στις δύο ορθογώνιες συνιστώσες $\{\cos, \sin\}$ του φέροντος :

$$r(t) = A \left\{ \left(\sum_{i=1}^{N_1} a_i \cos(\theta_i) \right) \cos(2\pi f_c t) - \left(\sum_{i=1}^{N_1} a_i \sin(\theta_i) \right) \sin(2\pi f_c t) \right\} \quad (3.10)$$

Επίσης αν εισάγουμε δυο τυχαίες διαδικασίες (random processes) $X_1(t)$ και $X_2(t)$ μπορούμε να γράψουμε:

$$r(t) = A\{X_1(t)\cos(2f_c t) - X_2(t)\sin(2f_c t)\} \quad (3.11)$$

Αν το N_1 είναι αρκετά μεγάλο (μεγάλος αριθμός από scattered waves είναι παρόντα), και χρησιμοποιώντας το Central Limit Theorem, μπορούμε να προσεγγίσουμε τα $X_1(t)$ και $X_2(t)$ με Gaussian random variables με zero mean και variance σ^2 . Η 3.10 γίνεται:

$$r(t) = AR(t)\cos(2f_c t + \theta(t)) \quad (3.12)$$

όπου

- Το πλάτος του σήματος λήψης $R(t)$ δίνεται ως

$$R(t) = \sqrt{X_1(t)^2 + X_2(t)^2} \quad (3.13)$$

όπου αφού οι $X_1(t)$ και $X_2(t)$ είναι Gaussian random variables μπορεί να δειχτεί ότι η $R(t)$ είναι Reileigh distributed random variable με PDF:

$$f_R(a) = \frac{a}{2\sigma^2} e^{-\frac{a^2}{2\sigma^2}}, \quad a \geq 0 \quad (3.14)$$

- Η φάση του σήματος λήψης $\theta(t)$ δίνεται ως:

$$\theta(t) = \tan^{-1}\left(\frac{X_2(t)}{X_1(t)}\right) \quad (3.15)$$

όπου αφού οι $X_1(t)$ και $X_2(t)$ είναι Gaussian random variables μπορεί να δειχτεί ότι η $\theta(t)$ είναι uniform distributed random variable με PDF:

$$f_\theta(\theta) = \frac{1}{2\pi}, \quad -\pi \leq \theta \leq \pi \quad (3.16)$$

Η εξασθένιση πλάτους $R(t)$ επηρεάζει σημαντικά την επίδοση των ψηφιακών τηλεπικοινωνιακών συστημάτων που λειτουργούν με fading channels. Πάντως η διάρκεια για την οποία η εξασθένιση (fading) λαμβάνεται σταθερή είναι τουλάχιστον για τη διάρκεια ενός εκπεμπόμενου συμβόλου.

Το μοντέλο της multipath διάδοσης για το κανάλι το οποίο καταλήγει σε fading του σήματος, είναι πρωταρχικά ένα αποτέλεσμα των χρονικών μεταβολών των φάσεων των διαφόρων κυμάτων στην κεραία του δέκτη. Δηλαδή, οι χρονικά μεταβαλλόμενες τυχαίες φάσεις που σχετίζονται με τα ανύσματα λήψης, συχνά καταλήγουν σε ανύσματα που προστίθενται με τρόπο αναιρετικό (destructive addition). Όταν συμβαίνει κάτι τέτοιο, το λαμβανόμενο σήμα $r(t)$ που προκύπτει έχει μικρό πλάτος ή και πρακτικά μηδέν. Σε άλλες περιπτώσεις τα ανύσματα προστίθενται με τρόπο ενισχυτικό (constructive addition) κι έτσι το λαμβανόμενο σήμα είναι μεγάλο. Έτσι οι μεταβολές του πλάτους στο λαμβανόμενο σήμα αποτελούν το fading

του σήματος και οφείλονται στα χρονικά μεταβαλλόμενα χαρακτηριστικά του multipath του καναλιού.

Όταν η απόκριση του καναλιού μοντελοποιείται ως μια μιγαδική Gaussian διαδικασία με μηδενική μέση τιμή, η περιβάλλουσα σε κάθε χρονική στιγμή ακολουθεί κατανομή Rayleigh. Σε αυτή την περίπτωση έχουμε ένα κανάλι με Rayleigh fading. Στην περίπτωση που υπάρχουν ακίνητοι σκεδαστές ή ανακλαστές του σήματος στο κανάλι, δηλαδή μια Line of Site (LOS) διαδρομή, εκτός από τους τυχαία κινούμενους σκεδαστές, η απόκριση του καναλιού δεν μπορεί πλέον να μοντελοποιηθεί έχοντας μηδενική μέση τιμή. Σε αυτή την περίπτωση η περιβάλλουσα ακολουθεί κατανομή Rice και έχουμε ένα κανάλι Rice fading.

Κατά τη μελέτη ασύρματων επικοινωνιών (wireless communications) χρησιμοποιούμε συνήθως τη στατιστική της περιβάλλουσας που ακολουθεί κατανομή Rayleigh, διότι το μοντέλο προσεγγίζει αρκετά τις πραγματικές συνθήκες μετάδοσης σε κινητά κανάλια επικοινωνίας. Η περιβάλλουσα του fading που ακολουθεί κατανομή Rayleigh εμφανίζεται συχνά στα HF και στα τροποσκεδαστικά κανάλια με αποτέλεσμα αυτό το μοντέλο καναλιού να είναι ευρύτατα αποδεκτό.

3.4 Fading models

α) Rayleigh: Η κατανομή Rayleigh χρησιμοποιείται συχνά σε multipath fading μοντέλα χωρίς LOS, δηλαδή χωρίς απευθείας διαδρομή. Σε αυτή την περίπτωση το πλάτος του καναλιού διάλειτουργίας R είναι random variable με PDF (Σχήμα):

$$f_R(a, \Omega) = \frac{2a}{\Omega} \exp\left(-\frac{a^2}{\Omega}\right), \quad a \geq 0 \quad (3.17)$$

όπου $\Omega = E\{R^2\} = 2\sigma^2$ δηλώνει τη μέση ισχύ της διαδρομής λήψης, που είναι το άθροισμα όλων των ανακλώμενων διαδρομών που φτάνουν με την ίδια καθυστέρηση. Επομένως, στιγμιαίο SNR ανά bit:

$$\gamma_b = R^2 \frac{E_b}{N_0} \quad (3.18)$$

είναι τώρα μία random variable με PDF chi-squared που δίνεται από (Σχήμα)

$$p_{\gamma_b}(\gamma_b, \overline{\gamma_b}) = \frac{1}{\overline{\gamma_b}} \exp\left(-\frac{\gamma_b}{\overline{\gamma_b}}\right), \quad \gamma_b \geq 0 \quad (3.19)$$

όπου $\overline{\gamma_b} = \frac{E_b}{N_0} E\{R^2\} = \frac{E_b}{N_0} \Omega$ δηλώνει το μέσο

(average) SNR ανά bit για το κανάλι.

Η κατανομή Rayleigh τυπικά συμφωνεί πολύ καλά με πειραματικές μετρήσεις σε κινητά κανάλια όπου δεν υπάρχει LOS διαδρομή ανάμεσα στην εκπέμποσα και την λαμβάνουσα κεραία.

β) Rice: Η κατανομή Rice χρησιμοποιείται συχνά σε μοντέλα διαύλων που περιέχουν μια απευθείας LOS διαδρομή και άλλες τυχαίες πιο αδύναμες διαδρομές. Το πλάτος της διάλωσης του καναλιού ακολουθεί την κατανομή:

$$p_R(a; \Omega, K) = \frac{2(1+K)e^{-K}a}{\Omega} \exp\left(-\frac{2(1+K)a^2}{\Omega}\right) I_0\left(2Ka\sqrt{\frac{1+K}{\Omega}}\right), \quad a \geq 0 \quad (3.20)$$

όπου $K = \frac{\text{average direct power}}{\text{average scattered power}}$ είναι η παράμετρος Rice, η οποία παίρνει τιμές

$K \geq 0$. Το SNR ανά bit του καναλιού, γ_b , είναι μία random variable με PDF:

$$p_R(\gamma; \overline{\gamma_b}, K) = \frac{(1+K)e^{-K}}{\overline{\gamma_b}} \exp\left(-\frac{(1+K)\gamma}{\overline{\gamma_b}}\right) I_0\left(2Ka\sqrt{\frac{(1+K)\gamma}{\overline{\gamma_b}}}\right), \quad \gamma \geq 0 \quad (3.21)$$

Η κατανομή Rice μεταβάλλεται από τη χειρότερη κατάσταση της Rayleigh, όταν $K=0$, μέχρι και καθόλου fading όταν $K = \infty$.

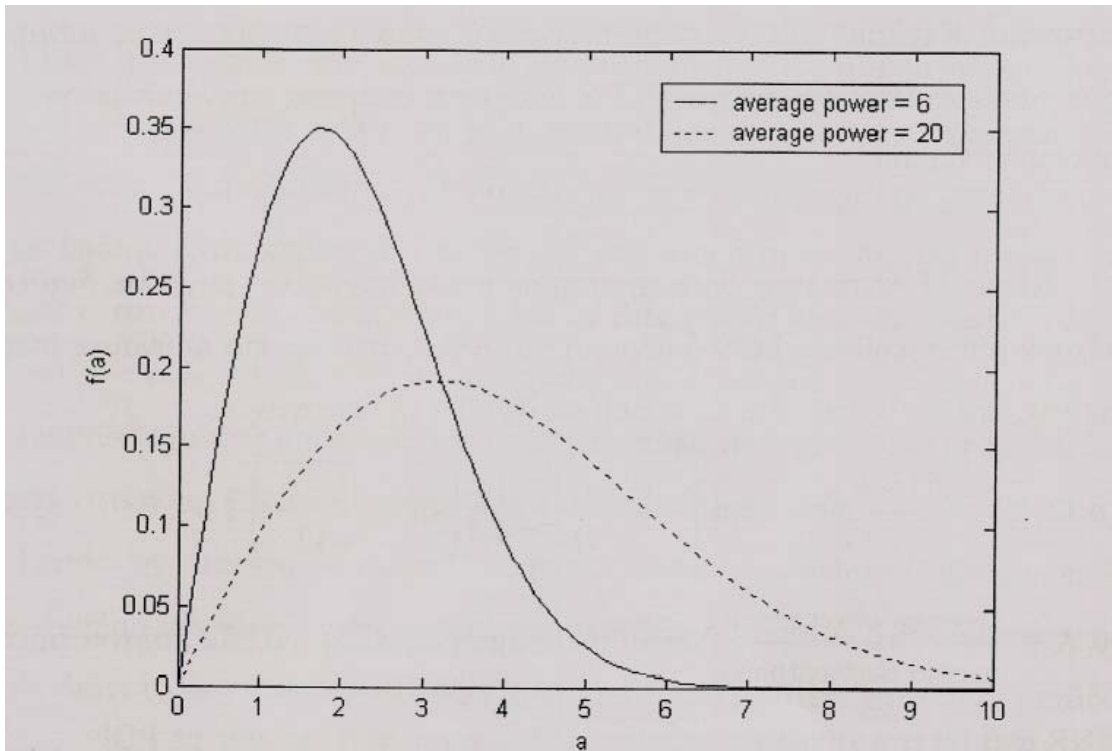
γ) Nakagami-m: Η κατανομή Nakagami-m δίνεται από τη σχέση:

$$p_R(a; \Omega, m) = \frac{2m^m a^{2m-1}}{\Omega^m \Gamma(m)} \exp\left(-\frac{ma^2}{\Omega}\right), \quad a \geq 0 \quad (3.22)$$

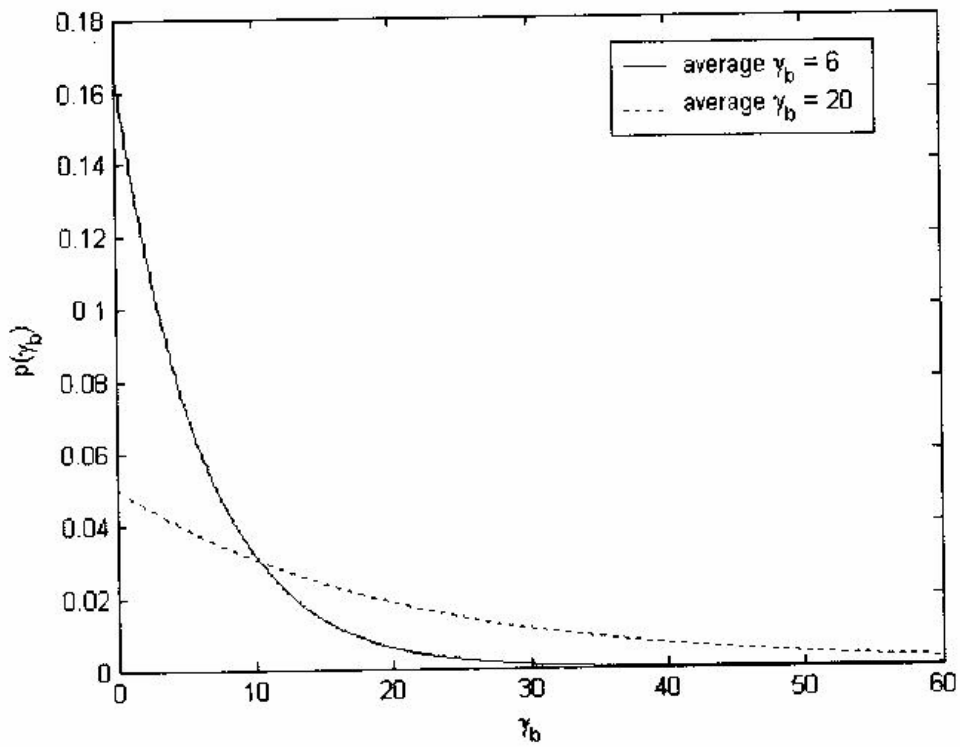
όπου $\Gamma(\cdot)$ είναι η συνάρτηση Gamma και m είναι η παράμετρος της Nakagami-m διάλωσης η οποία παίρνει τιμές από $\frac{1}{2}$ έως ∞ . Σε αυτή την περίπτωση το SNR ανά bit, γ_b , κατανέμεται σύμφωνα με την κατανομή Gamma που δίνεται από τον τύπο:

$$f_{\gamma_b}(\gamma; \overline{\gamma_b}, m) = \frac{m^m \overline{\gamma_b}^{m-1}}{\Gamma(m) \overline{\gamma_b}^{m-1}} \exp\left(-\frac{m\gamma}{\overline{\gamma_b}}\right), \quad \gamma_b \geq 0 \quad (3.23)$$

Η κατανομή Nakagami-m διαθέτει μέσω της παραμέτρου m τη μεγαλύτερη ποικιλία στατιστικών διαλείψεων. Για παράδειγμα περιλαμβάνει τη μονόπλευρη Gaussian κατανομή για $m=12$ και τη κατανομή Rayleigh για $m=1$. Στο όριο όπου $m = +\infty$ το κανάλι της κατανομής Nakagami-m τείνει προς ένα κανάλι χωρίς διαλείψεις. Τέλος όταν $m > 1$ τότε προσεγγίζει ένα κανάλι με Rician κατανομή διάλωσης.



Σχήμα 3.5 α PDF της τυχαίας μεταβλητής με κατανομή Rayleigh



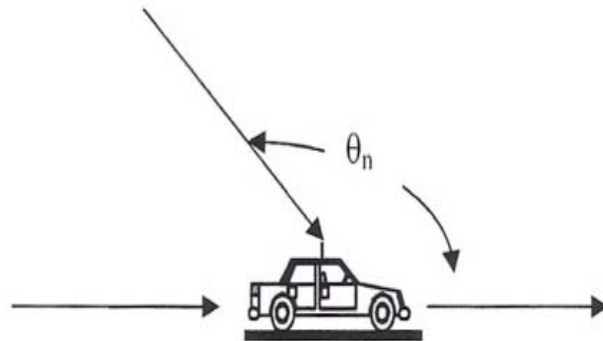
Σχήμα 3.5 β PDF του $\gamma_b = \frac{R^2 E_b}{N_0}$, όπου το R έχει Rayleigh κατανομή

3.5 Παραγωγή φακέλων εξασθένισης Rayleigh

Στην ενότητα αυτή θα δημιουργήσουμε Rayleigh fading για προσομοίωση κινητών επικοινωνιών με βάση την προηγούμενη ανάλυση. Θεωρούμε ότι ο κινητός δέκτης λαμβάνει τα κύματα, όπως φαίνεται στο Σχήμα 3. Ας αρχίσουμε από το μηχανισμό με τον οποίο συμβαίνει το fading. Το καθυστερημένο κύμα με γωνία εισόδου (incident angle) θ_n δίνεται από την εξίσωση 3.23 η οποία αντιστοιχεί στο Σχήμα 3 όταν ένα συνεχές κύμα μίας συχνότητας f_c (Hz) εκπέμπεται από το base station:

$$r_n = \text{Re}[e_n(t) \exp(j2\pi f_c t)] \quad (3.24)$$

όπου $\text{Re}[\]$ δηλώνει το πραγματικό μέρος του μιγαδικού αριθμού και δίνει το μιγαδικό φάκελο (envelope) του εισερχόμενου κύματος από την κατεύθυνση του αριθμού n .



Σχήμα 3.6 Δίαυλος πολλαπλών διαδρομών

Η $e_n(t)$ δίνεται από την 3.24 χρησιμοποιώντας το μήκος διάδοσης (propagation path length) από το base station για το εισερχόμενο κύμα, L_n (m), την ταχύτητα του κινητού δέκτη (mobile station), u (m/s), και το μήκος κύματος, λ (m).

$$e_n(t) = R_n(t) \exp \left[j \left(- \frac{2\pi(L - ut \cos \theta_n)}{\lambda} + \phi_n \right) \right] = x_n(t) + jy_n(t) \quad (3.25)$$

όπου $R_n(t)$ και $\phi_n(t)$ είναι ο φάκελος και η φάση του n -οστού εισερχόμενου κύματος, αντίστοιχα. Τα $x_n(t)$ και $y_n(t)$ είναι οι in-phase και quadrature συνιστώσες του $e_n(t)$,

αντίστοιχα. Το εισερχόμενο n-οστό κύμα μεταβάλλει τη συχνότητα του φέροντος κατά $u \cos(\theta_n)/\lambda$ (Hz) λόγω του φαινομένου Doppler shift στις κινητές ασύρματες επικοινωνίες. Το μέγιστο Doppler shift το οποίο ορίζεται ως f_d , έχει τιμή u/λ , όταν το εισερχόμενο κύμα έρχεται από την κατεύθυνση της κίνησης του δέκτη, οπότε και $\cos(\theta_n)=1$. Επομένως αυτό είναι το μέγιστο Doppler shift. Επίσης το καθυστερημένο κύμα που έρχεται από την πίσω πλευρά του κινητού δέκτη έχει frequency shift $-f_d$ (Hz).

Αφού το συνολικό σήμα λήψης $r(t)$ είναι το άθροισμα των σημάτων, αν έχουμε N_1 κύματα που φτάνουν ταυτόχρονα από διάφορες κατευθύνσεις, έχουμε :

$$\begin{aligned} r(t) &= \sum_{n=1}^{N_1} r_n(t) = \text{Re} \left[\left(\sum_{n=1}^{N_1} e_n(t) \right) \exp(j2f_c t) \right] \\ &= \text{Re} \left[(x(t) + jy(t)) (\cos(2f_c t) + j \sin(2f_c t)) \right] \\ &= x(t) \cos(2f_c t) - jy(t) \sin(2f_c t) \end{aligned} \quad (3.26)$$

όπου $x(t)$ και $y(t)$ δίνονται από

$$\begin{aligned} x(t) &= \sum_{n=1}^{N_1} x_n(t) \\ y(t) &= \sum_{n=1}^{N_1} y_n(t) \end{aligned} \quad (3.27)$$

και $x(t)$ και $y(t)$ είναι κανονικοποιημένες τυχαίες μεταβλητές με μέση τιμή 0 και διασπορά σ , όταν το N_1 είναι αρκετά μεγάλο.

Η γωνία λήψης των λαμβανόμενων σημάτων έχει ομοιόμορφη κατανομή (uniform distribution). Έστω επίσης ότι ο αριθμός των κυμάτων που φτάνουν ταυτόχρονα στο δέκτη από διαφορετικές διαδρομές είναι N_1 . Σε αυτή την περίπτωση, η μιγαδική εξασθένιση (fading fluctuation) σε ένα ισοδύναμο lowpass system αποδεικνύεται ότι είναι:

$$\begin{aligned} r(t) &= x(t) + jy(t) \\ &= \left[\sqrt{\frac{2}{N_1+1}} \sum_{n=1}^{N_1} \cos\left(\frac{\pi}{N_1}\right) \cos\left\{2f_d \cos\left(\frac{2\pi}{N}\right)t\right\} + \frac{1}{\sqrt{N_1+1}} \cos(2f_d t) \right] \\ &\quad + j \sqrt{\frac{2}{N_1}} \sum_{n=1}^{N_1} \sin\left(\frac{\pi}{N_1}\right) \cos\left\{2f_d \cos\left(\frac{2\pi}{N}\right)t\right\} \end{aligned} \quad (3.28)$$

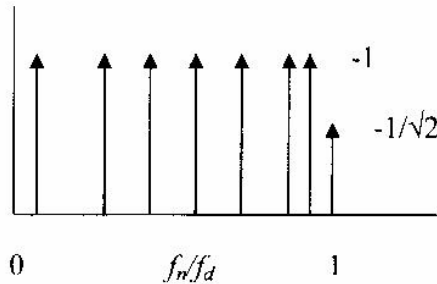
όπου N είναι ένας περττός αριθμός και το N_1 δίνεται από την

$$N_1 = 4N - 1 \quad (3.29)$$

Σε αυτή την περίπτωση οι παρακάτω σχέσεις ικανοποιούνται:

$$\begin{aligned} E[x^2(t)] &= E[y^2(t)] = \frac{1}{2} \\ E[x(t)y(t)] &= 0 \end{aligned} \quad (3.30)$$

Για $N_1=8$ οι συχνότητες οι οποίες δίνουν το φάσμα του κινητού καναλιού (mobile radio spectrum) φαίνονται στο Σχήμα 4.



Σχήμα 3.7 Συχνότητες ($N_1=8$) που δίνουν το mobile radio spectrum

Η σχέση που δίνει τη συνδυασμένη πυκνότητα πιθανότητας (combination probability density) $p(x,y)$, όπου $x=x(t)$, $y=y(t)$, είναι:

$$p(x, y) = \frac{1}{2\pi\sigma^2} \exp\left(-\frac{x^2 + y^2}{2\sigma^2}\right) \quad (3.31)$$

Επίσης το $r(t)$ μπορεί να εκφραστεί χρησιμοποιώντας το πλάτος και τη φάση του λαμβανόμενου σήματος:

$$r(t) = R(t) \cos(2\pi f_c t + \theta(t)) \quad (3.32)$$

όπου $R(t)$ και $\theta(t)$ δίνονται από:

$$\begin{aligned} R(t) &= R = \sqrt{x^2 + y^2} \\ \theta(t) &= \theta = \tan^{-1}\left(\frac{y}{x}\right) \end{aligned} \quad (3.33)$$

Χρησιμοποιώντας ένα μετασχηματισμό των μεταβλητών, η $p(x,y)$ μπορεί να εκφραστεί ως $p(R,\theta)$

$$p(R, \theta) = \frac{R}{2\pi\sigma^2} \exp\left(-\frac{R^2}{2\sigma^2}\right) \quad (3.34)$$

Αν ολοκληρώσουμε την $p(R,\theta)$ για θ από 0 έως 2π (όλες οι δυνατές τιμές της θ), προκύπτει η συνάρτηση πυκνότητας πιθανότητας $p(R)$

$$p(R) = \frac{R}{\sigma^2} \exp\left(-\frac{R^2}{2\sigma^2}\right) \quad (3.35)$$

Επίσης μπορούμε να βρούμε τη συνάρτηση πυκνότητας πιθανότητας $p(\theta)$ αν ολοκληρώσουμε την $p(R, \theta)$ για όλες τις τιμές του R από 0 έως ∞

$$p(\theta) = \frac{1}{2\pi} \quad (3.36)$$

Από αυτές τις σχέσεις προκύπτει ότι ο φάκελος κυμαίνεται με κατανομή Rayleigh ενώ η φάση μεταβάλλεται με ομοιόμορφη κατανομή.

ΚΕΦΑΛΑΙΟ 4

ΑΠΟΔΟΣΗ ΤΩΝ ΨΗΦΙΑΚΩΝ ΔΙΑΜΟΡΦΩΣΕΩΝ ΣΤΑ

ΑΣΥΡΜΑΤΑ ΚΑΝΑΛΙΑ

4.1 Γενικά

Σ' αυτό το κεφάλαιο θα αναπτύξουμε το θέμα της απόδοσης των ψηφιακών διαμορφώσεων όταν χρησιμοποιούνται σε κανάλια με AWGN θόρυβο, και σε κανάλια με flat-fading. Υπάρχουν δύο κριτήρια που συνήθως χρησιμοποιούνται για να περιγράψουμε την απόδοση : το ένα είναι η πιθανότητα του λάθους , η οποία μπορεί να αποδίδει την πιθανότητα λάθους στο bit ή στο σύμβολο, το δεύτερο κριτήριο είναι η outage πιθανότητα , η οποία ορίζεται ως η πιθανότητα όπου η στιγμιαία τιμή του λόγου σήματος προς θόρυβο πέφτει κάτω από ένα δεδομένο κατώφλι. Το Flat-fading μπορεί να προκαλέσει δραματική αύξηση είτε στη μέση τιμή του ρυθμού σφαλμάτων (bit-error-rate) είτε στην πιθανότητα απώλειας (outage probability).

Στα ασύρματα κανάλια , υπάρχουν επίσης και άλλοι δύο παράμετροι , η εξασθένιση επιλεκτικής συχνότητας (frequency selective fading) καθώς επίσης και η ολίσθηση λόγω Doppler. Στη περίπτωση της επιλεκτικής εξασθένισης συχνότητας αυτό μπορεί να προκαλέσει αύξηση της παρεμβολής ενδοσυμβόλου (ISI), η οποία με τη σειρά της προκαλεί ένα ανεπίδεκτο μικρό περιθώριο λάθους στο λαμβανόμενο σήμα. Από την άλλη μεριά το φαινόμενο Doppler προκαλεί διεύρυνση του φάσματος , η οποία οδηγεί σε παρεμβολή παρακείμενου καναλιού (τυπικά μικρή για λογικές ταχύτητες χρήστη), καθώς επίσης δημιουργεί και μία εξαιρετικά μικρή οροφή λάθους σε σήματα με διαφορική κωδικοποίηση φάσης (π.χ DPSK).

Σ' αυτό το κεφάλαιο θα περιγράψουμε αναλυτικά την επίδραση και την απόδοση των συστημάτων διαφόρων ψηφιακών διαμορφώσεων, σε περιβάλλον με θόρυβο, και με εξασθένιση (flat-fading) & Rayleigh κατανομή.

4.2 Κανάλια AWGN

Στη παράγραφο αυτή θα αναφέρουμε τις παραμέτρους που παρακάτω θα χρησιμοποιούμε για την ανάλυση μας . Ως πρώτη αναφέρεται, ο λόγος ισχύων σήματος προς θόρυβο (signal-to-noise power ratio) (SNR) και οι σχετικές μ' αυτόν , E_b , η οποία ορίζεται ως η ενέργεια του κάθε bit και το E_s που δίνει την ενέργεια του κάθε συμβόλου.

4.2.1 Signal-to-Noise Power Ratio and Bit/Symbol Energy

Σ' ένα AWGN κανάλι το διαμορφωμένο σήμα $s(t) = R\{u(t)e^{j2\pi f_c t}\}$ έχει θόρυβο $n(t)$ ο οποίος προστίθεται πριν την λήψη του σήματος στο δέκτη. Ο θόρυβος αυτός $n(t)$ είναι μία λευκή Gaussian τυχαία διαδικασία, με μέση τιμή μηδενική και φασματική πυκνότητα ισχύος $N_0/2$.

Το σήμα που λαμβάνεται στο δέκτη είναι ίσο με $r(t) = s(t) + n(t)$. Ορίζουμε το λόγο σήματος προς θόρυβο (SNR), ως το λόγο της ισχύος του λαμβανομένου σήματος P_r προς την ισχύ του θορύβου, μέσα στο εύρος ζώνης του μεταδιδόμενου σήματος $s(t)$. Η λαμβανόμενη ισχύς P_r καθορίζεται από την εκπεμπόμενη ισχύ, τις απώλειες διάδοσης, τις απώλειες λόγω σκίασης και των φαινομένων διάλειψης. Η ισχύς του θορύβου καθορίζεται από το εύρος ζώνης του εκπεμπόμενου σήματος και των ιδιοτήτων του $n(t)$. Ειδικότερα, εάν το εύρος ζώνης της περιβάλλουσας $u(t)$ του $s(t)$ είναι B τότε το εύρος ζώνης του μεταδιδόμενου σήματος $s(t)$ είναι $2B$.

Από τη στιγμή που ο θόρυβος $n(t)$ έχει σταθερή φασματική πυκνότητα ισχύος ίση με $N_0/2$, η ολική ισχύς θορύβου μέσα σε $2B$ εύρος ζώνης είναι ίση με $N = N_0/2 \times 2B = N_0B$. Έτσι το λαμβανόμενο SNR δίνεται από

$$\text{SNR} = P_r/N_0B \quad (4.1)$$

Σε συστήματα με παρεμβολή, αντί του μεγέθους SNR για τους υπολογισμούς της πιθανότητας λάθους, χρησιμοποιούμε το μέγεθος SINR το οποίο είναι ο λόγος λαμβανόμενου σήματος και παρεμβολής προς το θόρυβο. Εάν η παρεμβολή στατιστικά προσεγγίζει τον Gaussian θόρυβο, τότε μπορούμε να κάνουμε την ακόλουθη προσέγγιση:

Το λαμβανόμενο SINR δίνεται από

$$\text{SINR} = P_r/(N_0B + P_I)$$

Όπου η P_I είναι η μέση ισχύς της παρεμβολής. Το SNR συχνά εκφράζεται με όρους ενέργειας σήματος ανά bit (E_b) ή ενέργειας ανά σύμβολο (E_s) και έχουμε ότι το SNR είναι ίσο με:

$$\text{SNR} = \frac{P_r}{N_0B} = \frac{E_s}{N_0BT_s} = \frac{E_b}{N_0BT_b} \quad (4.2)$$

όπου το T_s είναι η διάρκεια συμβόλου και το T_b είναι η διάρκεια bit (για δυαδική διαμόρφωση το $T_s = T_b$ και $E_s = E_b$). Για δυαδικούς παλμούς με $T_s = 1/B$, για παράδειγμα για raised cosine pulses με $\beta = 1$, έχουμε $\text{SNR} = E_s/N_0$ για multilevel

και $SNR = E_b/N_0$ για δυαδικά σήματα . Για την γενική περίπτωση έχουμε ότι $T_s = k/B$ για μερικά σταθερά k , για τα οποία ισχύει $k \cdot SNR = E_s/N_0$.

Οι παράμετροι $\gamma_s = E_s/N_0$ and $\gamma_b = E_b/N_0$ ονομάζονται ως SNR ανά σύμβολο και SNR ανά bit, αντίστοιχα. Για την περίπτωση της απόδοσης , ενδιαφερόμαστε για την πιθανότητα λάθους bit , αυτό που ονομάζουμε bit error probability, P_b ως συνάρτηση του γ_b . Εν τούτοις , για M-ary σήματα όπως MPAM και MPSK, η bit error probability εξαρτάται και από την symbol error probability και από την αντιστοίχιση(mapping) των bits στα σύμβολα .

Έτσι , τυπικά, υπολογίζουμε την symbol error probability P_s ως συνάρτηση του γ_s και μετά την P_b σαν συνάρτηση του γ_b χρησιμοποιώντας ακριβής ή προσεγγιστική μετατροπή. Κατά την προσεγγιστική μετατροπή υποθέτουμε ότι τυπικά, η ενέργεια του συμβόλου μοιράζεται εξίσου σ' όλα τα bits, και εφόσον χρησιμοποιείται η κωδικοποίηση κατά Gray , για λογικές τιμές του SNR, το λάθος σ' ένα σύμβολο αντιστοιχεί σε λάθος ενός bit. Αυτές οι υποθέσεις για την περίπτωση των M-ary σημάτων οδηγούν στις ακόλουθες προσεγγίσεις

$$\begin{aligned} \gamma_b &\approx \frac{\gamma_s}{\log_2 M} \\ P_b &\approx \frac{P_s}{\log_2 M} \end{aligned} \quad (4.3)$$

4.2.2 Error Probability for BPSK and QPSK

Θα εξετάσουμε πρώτα τη περίπτωση της BPSK διαμόρφωσης με συμφασική ανίχνευση και τέλεια ανάκτηση της συχνότητας και της φάσης του φορέα. Με την δυαδική διαμόρφωση (BPSK) κάθε σύμβολο αντιστοιχεί σ' ένα bit, έτσι το symbol και το bit error rate είναι ουσιαστικά τα ίδια .

Το εκπεμπόμενο σήμα είναι το $s_1(t) = Ag(t) \cos(2\pi fct)$ το οποίο στέλνει το 0 bit και το σήμα $s_2(t) = -Ag(t) \cos(2\pi fct)$ στέλνει το bit 1.

Γνωρίζουμε ότι η probability error δίνεται από τον τύπο

$$P_b = Q\left(\frac{d_{\min}}{\sqrt{2N_o}}\right) \quad (4.4)$$

Γνωρίζουμε ότι $d_{\min} = \|s_1 - s_0\| = \|A - (-A)\| = 2A$. Συσχετίζοντας το A με την ενέργεια /bit έχουμε :

$$E_b = \int_0^{T_b} S_1^2(t)dt = \int_0^{T_b} S_2^2(t)dt = \int_0^{T_b} A^2 g^2(t) \cos^2(2\pi f_c t) dt = A^2 \quad (4.5)$$

Έτσι, για την περίπτωση της BPSK τα δύο σύμβολα στο διάγραμμα αστερισμού, με όρους ενέργειας δίνονται από $s_0 = \sqrt{Eb}$ και $s_1 = -\sqrt{Eb}$. Έτσι η ελάχιστη απόσταση των δύο συμβόλων είναι $d_{min} = 2A = 2\sqrt{Eb}$. Αντικαθιστώντας στην 4.4 έχουμε την πιθανότητα λάθους (bit) :

$$P_b = Q\left(\frac{2\sqrt{Eb}}{\sqrt{2N_0}}\right) = Q\left(\sqrt{\frac{2Eb}{N_0}}\right) = Q(\sqrt{2\gamma_b}) \quad (4.6)$$

Η QPSK διαμόρφωση αποτελείται από BPSK διαμορφώσεις και στο in-phase και στο quadrature components του σήματος. Με τέλεια ανίχνευση φάσης και φορέα, οι δύο αυτές συνιστώσες του λαμβανόμενου σήματος είναι ορθογώνιες μεταξύ τους.

Η bit error probability για κάθε συνιστώσα είναι ίδια με αυτή της BPSK, δηλαδή : $P_b = Q(\sqrt{2\gamma_b})$.

Αντίστοιχα., η πιθανότητα λάθος συμβόλου για την κάθε συνιστώσα είναι ίση με :

$$P_s = 1 - [1 - Q(\sqrt{2\gamma_b})]^2 \quad (4.7)$$

Από τη στιγμή που η ενέργεια του συμβόλου μοιράζεται ίσα μεταξύ της in-phase συνιστώσας και της quadrature, τότε έχουμε $\gamma_s = 2\gamma_b$. Αντικαθιστώντας αυτό στην 4.7 έχουμε για το P_s :

$$P_s = 1 - [1 - Q(\sqrt{\gamma_s})]^2 \quad (4.8)$$

Γνωρίζουμε ότι η P_s για την περίπτωση της QPSK είναι ίση με :

$$P_s \leq 2Q\left(\frac{A}{\sqrt{N_0}}\right) + Q\left(\frac{\sqrt{2A}}{\sqrt{N_0}}\right) \quad (4.9)$$

Δεδομένου ότι $\gamma_s = 2\gamma_b = A^2/N_0$, έχουμε ότι :

$$P_s \leq 2Q(\sqrt{\gamma_s}) + Q(\sqrt{2\gamma_s}) \leq eQ(\sqrt{\gamma_s}) \quad (4.10)$$

$$P_s \leq \frac{3}{\sqrt{\pi}} \exp\left[-\frac{.5A^2}{N_0}\right] = \frac{3}{\sqrt{\pi}} \exp\left[-\frac{\gamma_s}{2}\right] \quad (4.11)$$

Χρησιμοποιώντας το γεγονός ότι η ελάχιστη απόσταση των σημείων του διαγράμματος αστερισμού είναι $d_{min} = \sqrt{2}A^2$, έχουμε την κοντινότερη προσέγγιση που δίνει :

$$P_s \approx 2Q\left(\sqrt{\frac{A^2}{N_0}}\right) = 2Q(\sqrt{\gamma_s/2}) \quad (4.12)$$

Για την περίπτωση που έχουμε κωδικοποίηση κατά Gray , μπορούμε να υπολογίσουμε προσεγγιστικά την πιθανότητα P_b από την πιθανότητα P_s λόγω του ότι $P_b \approx P_s/2$, (έχουμε 2 bits/σύμβολο) .

4.2.3 Error Probability για MPSK

Στο διάγραμμα αστερισμού για την περίπτωση των MPSK σημάτων τα σύμβολα δίνονται από τους ακόλουθους τύπους :

$$\begin{aligned} S_{i1} &= A \cos \left[\frac{2\pi(i-1)}{M} \right] & i = 1, \dots, M \\ S_{i2} &= A \sin \left[\frac{2\pi(i-1)}{M} \right] \end{aligned} \quad (4.13)$$

Η ενέργεια συμβόλου είναι ίση με $E_s = A^2$, και το $\gamma_s = A^2/N_0$. Για το λαμβανόμενο σήμα , το οποίο το αναπαριστούμε σε πολικές συντεταγμένες $\mathbf{x} = r e^{j\theta}$, ένα λάθος συμβαίνει εάν το i th σημείο του διαγράμματος αστερισμού μεταδίδεται και το $\theta \notin (2\pi(i-1-.5)/M, 2\pi(i-1+.5)/M)$.

Η δεσμευμένη κατανομή του r δίνεται από τον ακόλουθο τύπο :

$$p(r, \theta) = \frac{r}{\pi N_0} \exp \left[-\frac{1}{N_0} \left(r^2 - 2\sqrt{2E_s} r \cos \theta + 2E_s \right) \right] \quad (4.14)$$

Δεδομένου ότι η error probability, εξαρτάται μόνον από την κατανομή του θ , ολοκληρώνουμε ως προς το r , και έχουμε :

$$p(\theta) = \int_0^{\infty} p(r, \theta) dr = \frac{1}{\pi} e^{-2\gamma_s \sin^2(\theta)} \int_0^{\infty} z \exp \left[-\left(z - \sqrt{2\gamma_s} \cos(\theta) \right)^2 \right] dz \quad (4.15)$$

Εκ συμμετρίας, η πιθανότητα λάθους είναι ίδια για κάθε σημείο του διαγράμματος αστερισμού. Έτσι για να έχουμε την P_s από την error probability, υποθέτουμε ότι έχουμε το σημείο $s_1 = (A, 0)$ το οποίο μεταδίδεται, και είναι :

$$P_s = 1 - \int_{-\frac{\pi}{M}}^{\frac{\pi}{M}} p(\theta) d\theta = 1 - \int_{-\frac{\pi}{M}}^{\frac{\pi}{M}} \frac{1}{\pi} e^{-2\gamma_s \sin^2(\theta)} \int_0^{\infty} 2 \exp \left[-\left(z - \sqrt{2\gamma_s} \cos(\theta) \right)^2 \right] dz \quad (4.16)$$

Μία κλειστή φόρμα για το παραπάνω ολοκλήρωμα δεν υπάρχει για $M > 4$, και έτσι η ακριβής τιμή του P_s θα πρέπει να υπολογιστεί αριθμητικά .

Κάθε σημείο στο διάγραμμα αστερισμού του MPSK έχει δύο άλλα κοντινά σημεία με απόσταση $d_{min} = 2A \sin(\pi/M)$. Έτσι, η κοντινότερη γειτονική προσέγγιση του P_s δίνεται από την ακόλουθη εξίσωση :

$$P_s \approx 2Q\left(\sqrt{2A/N_0} \times \sin\left(\frac{\pi}{M}\right)\right) = 2Q\left(\sqrt{2\gamma_s} \sin\left(\frac{\pi}{M}\right)\right) \quad (4.17)$$

Όπως δείξαμε για την περίπτωση της QPSK, αυτή η προσέγγιση διαφέρει από την ακριβή τιμή της P_s . Είναι ευκολότερο και πολύ πιο απλούστερο να υπολογιστεί από το να υπολογιστεί το ολοκλήρωμα της 4.16 για να αποκτηθεί η ακριβής τιμή της P_s . Μία πιο αυστηρή προσέγγιση για την εύρεση της P_s μπορεί να γίνει με την προσέγγιση της

$p(\theta)$ με τον ακόλουθο τρόπο :

$$p(\theta) \approx \sqrt{\gamma_s} \pi \cos(\theta) e^{\gamma_s \sin^2(\theta)} \quad (4.18)$$

Χρησιμοποιώντας αυτή την προσέγγιση για το αριστερό μέρος της 4.16 έχουμε

$$P_s \approx 2Q\left(\sqrt{2\gamma_s} \sin\left(\frac{\pi}{M}\right)\right) \quad (4.19)$$

Η εύρεση για την πιθανότητα λάθους για την περίπτωση της MPSK υποθέτει ότι η φάση του φορέα είναι απολύτως γνωστή στον δέκτη.

Για τον υπολογισμό σφάλματος της φάσης, η κατανομή της $p(\theta)$, που χρησιμοποιείται για να βρεθεί η P_s , πρέπει να ενσωματώσει την κατανομή της περιστροφής της φάσης η οποία σχετίζεται με την μετατόπιση της φάσης του φορέα. Αυτή η κατανομή είναι τυπικά μία συνάρτηση της τεχνικής υπολογισμού της φάσης του φορέα και του SNR. Έχει αποδειχθεί ότι σημαντική μετατόπιση της φάσης οδηγεί σε αδυναμία ελάττωσης της bit error probability. Επιπλέον οι μη δυαδικές διαμορφώσεις, είναι περισσότερο ευαίσθητες απ' ότι η BPSK, μεταξύ των στις μετατοπίσεις φάσεων εξαιτίας της εναλλαγής μεταξύ των στοιχείων in-phase και quadrature.

Η επίδραση στον υπολογισμό του phase error μπορεί να γίνει εξαιρετικά δύσκολη όταν έχουμε fast fading, όπου η φάση έχει πολύ γρήγορες μεταβολές εξαιτίας των εμποδίων που υπάρχουν στη διαδρομή του σήματος και της παρεμβολής λόγω πολυδιαδρομών (multipath interference). Ακόμα και στη περίπτωση των διαφορικών διαμορφώσεων, οι αλλαγές φάσης κατά τη διάρκεια και στο χρονικό διάστημα μεταξύ συμβόλων μπορούν να παράγουν λάθη μη ελαττώσιμα. Ένας επιπλέον παράγον που μπορεί να μειώσει την απόδοση ενός συστήματος είναι τα λάθη χρονισμού (timing errors)

4.2.4 Error Probability for MPAM and MQAM

Το διάγραμμα αστερισμού για σήματα MPAM έχει σημεία με $A_i = (2i - 1 - M)d$, με $i = 1, 2, \dots, M$. Κάθε ένα από τα $M - 2$ inner σημεία του constellation έχει δύο άλλα γειτονικά σημεία σε απόσταση $2d$. Η πιθανότητα να εμφανιστεί λάθος όταν στέλνεται ένα από τα παραπάνω σημεία είναι ουσιαστικά η πιθανότητα ο θόρυβος να ξεπεράσει την απόσταση d σε μία από τις κατευθύνσεις δηλαδή : $P_s(s_i) = p(|\mathbf{n}| > d)$, $i = 2, \dots, M - 1$.

Για τα outer σημεία του constellation υπάρχει μόνο ένα κοντινό σημείο, έτσι σ' αυτή τη περίπτωση λάθος θα συμβεί εάν ο θόρυβος υπερβεί την απόσταση d σε μία κατεύθυνση μόνον δηλαδή : $P_s(s_i) = p(\mathbf{n} > d) = .5p(|\mathbf{n}| > d)$, $i = 1, M$.

Η πιθανότητα λάθους είναι :

$$P_s = \frac{1}{M} \sum_{i=1}^M P_s(s_i) = \frac{M-2}{M} 2Q\left(\sqrt{\frac{2d^2}{N_0}}\right) + \frac{2}{M} Q\left(\sqrt{\frac{2d^2}{N_0}}\right) = \frac{2(M-1)}{M} Q\left(\sqrt{\frac{2d^2}{N_0}}\right) \quad (4.20)$$

Η μέση ενέργεια ανά σύμβολο για MPAM σήματα είναι :

$$\overline{E_s} = \frac{1}{M} \sum_{i=1}^M A_i^2 = \frac{1}{M} \sum_{i=1}^M (2i - 1 - M)^2 d^2 = \frac{1}{3} (M^2 - 1) d^2 \quad (4.21)$$

Έτσι μπορούμε να γράψουμε την P_s με όρους μέσης τιμής ενέργειας E_s

$$P_s = \frac{2(M-1)}{M} Q\left(\sqrt{\frac{6\overline{E_s}}{M^2-1}}\right) \quad (4.22)$$

Για την περίπτωση της MQAM διαμόρφωσης θεωρούμε ότι έχουμε διάγραμμα αστερισμού ένα τετράγωνο με $M=L^2$. Στο διάγραμμα αστερισμού τα σημεία στο inphase και quadrature έχουν τιμές ίσες με $A_i = (2i - 1 - L)d$, $i = 1, 2, \dots, L$. Η πιθανότητα λάθους συμβόλου (symbol error probability) για κάθε τμήμα ενός MQAM συστήματος δίνεται από την εξίσωση 4.22 θεωρώντας ότι $L = \sqrt{M}$ και γ_s ίσο με την μέση τιμή της ενέργειας κάθε συμβόλου, δηλαδή :

$$P_s = \frac{2(\sqrt{M}-1)}{\sqrt{M}} Q\sqrt{\frac{3\overline{E_s}}{M-1}} \quad (4.23)$$

Στη παραπάνω εξίσωση φαίνεται ότι το γ_s πολλαπλασιάζεται επί 3 αντί του 6 που υπάρχει στην 4.22. Αυτό οφείλεται στο γεγονός ότι στην MQAM η συνολική μέση ενέργεια γ_s μοιράζεται μεταξύ των in-phase και quadrature. Έτσι η πιθανότητα λάθους συμβόλου (probability of symbol error) για ένα MQAM σύστημα δίνεται από την ακόλουθη εξίσωση :

$$P_s = 1 - \left(1 - \frac{2(\sqrt{M} - 1)}{\sqrt{M}} Q \left(\sqrt{\frac{3\gamma_s}{M-1}} \right) \right)^2 \quad (4.24)$$

Η μέθοδος προσέγγισης του κοντινότερου γείτονα για την εύρεση της symbol error probability εξαρτάται από το εάν τα σημεία στο διάγραμμα αστερισμού είναι inner ή outer.

Για μη τετράγωνα διαγράμματα αστερισμού, είναι σχετικά απλό ναδειχθεί ότι η πιθανότητα λάθους συμβόλου (symbol error probability) είναι στο πάνω όριο

$$P_s \leq 1 - \left[1 - 2Q \left(\sqrt{\frac{3\gamma_s}{M-1}} \right) \right]^2 \leq 4Q \left(\sqrt{\frac{3\gamma_s}{M-1}} \right) \quad (4.25)$$

Η προσέγγιση του κοντινότερου σημείου δίνει, για μη τετράγωνα διαγράμματα :

$$P_s \approx M_{d_{min}} Q \left(\frac{d_{min}}{\sqrt{2N_0}} \right) \quad (4.26)$$

Όπου $M_{d_{min}}$ είναι ο μεγαλύτερος αριθμός γειτόνων για κάθε σημείο του διαγράμματος και d_{min} είναι η ελάχιστη απόσταση στο διάγραμμα.

Συγκρινώντας την P_s για διαμορφώσεις τύπου MPSK και MQAM παρατηρούμε ότι για την περίπτωση των MPSK η P_s είναι μεγαλύτερη έναντι των MQAM (για ίδιο M και γ_b) εξαιτίας του γεγονότος ότι η MQAM χρησιμοποιεί και το πλάτος και τη φάση για να κωδικοποιήσει τα δεδομένα, ενώ η MPSK χρησιμοποιεί μόνον τη φάση. Έτσι για την ίδια ενέργεια συμβόλου ή bit, η MQAM χρησιμοποιεί πιο αποδοτικά την ενέργεια και έχει καλύτερη απόδοση. Ένας MQAM αποδιαμορφωτής απαιτεί υπολογισμό και του πλάτους και της φάσης του καναλιού έτσι ώστε οι περιοχές απόφασης που χρησιμοποιεί για την ανίχνευση του εκπεμπόμενου Bit δεν είναι ασύμμετρες ούτε ως προς τη φάση ούτε ως προς το πλάτος.

Η ανάλυση για τη μείωση της απόδοσης, εξαιτίας του λάθους της φάσης είναι όμοια με την περίπτωση της MPSK. Το πλάτος του καναλιού χρησιμοποιείται για να βαθμονομηθούν οι περιοχές απόφασης ώστε να ανταποκρίνονται στα εκπεμπόμενα σύμβολα, αυτή η βαθμονόμηση καλείται Automatic Gain Control (AGC). Εάν το κέρδος του καναλιού (channel gain) υπολογίζεται με λάθη, τότε το AGC λαθεμένα αυξάνει το λαμβανόμενο σήμα, το οποίο οδηγεί σε λάθος αποδιαμόρφωση ακόμα και όταν δεν υπάρχει θόρυβος. Το κέρδος καναλιού ευρίσκεται, τυπικά, από τα pilot symbols τα οποία βοηθούν στον σωστό υπολογισμό του κέρδους του καναλιού στον

δέκτη. Παρ' όλα αυτά τα pilot symbols δεν οδηγούν σε τέλειο υπολογισμό του καναλιού και ο λάθος αυτός υπολογισμός, μπορεί να οδηγήσει σε bit errors.

4.3 Κανάλια Fading

Σε κανάλια θορύβου AWGN η πιθανότητα του symbol error εξαρτάται από το SNR ή το ισodύναμο του γ_s . Σε περιβάλλον με εξασθένιση (fading) η λαμβανόμενη ισχύς του σήματος, μεταβάλλεται τυχαία ανάλογα με την απόσταση ή το χρόνο εξαιτίας των φαινομένων της σκίασης ή της πολυδιαδρομικότητας του σήματος. Έτσι, στο fading το γ_s είναι μία τυχαία μεταβλητή με κατανομή $p_{\gamma_s}(\gamma)$, και η $P_s(\gamma_s)$ είναι επίσης τυχαία.

Το μέτρο της απόδοσης, όταν η γ_s είναι τυχαία εξαρτάται από τον ρυθμό της μεταβολής του fading. Υπάρχουν τρία διαφορετικά κριτήρια απόδοσης που μπορούν να χρησιμοποιηθούν για να χαρακτηρίσουν την μεταβλητή P_s :

- Η outage probability, P_{out} , η οποία ορίζεται ως η πιθανότητα του ότι το γ_s θα πέσει κάτω από μία δεδομένη τιμή, αντίστοιχη στη μέγιστη επιτρεπτή P_s .

- Η μέση πιθανότητα λάθους, $\overline{P_s}$, στρογγυλεμένη στο μέσο όρο της κατανομής της γ_s .

- Η συνδυασμένη μέση πιθανότητα λάθους και outage probability, οριζόμενες ως η μέση πιθανότητα λάθους η οποία μπορεί να επιτευχθεί σε κάποιο ποσοστό του χρόνου ή της θέσης.

Η μέση πιθανότητα του symbol error εφαρμόζεται όταν η εξασθένιση του σήματος είναι της τάξης της διάρκειας του συμβόλου ($T_s \approx T_c$), έτσι ώστε το επίπεδο της εξασθένισης του σήματος να είναι σχεδόν σταθερό στη διάρκεια ενός συμβόλου. Επειδή υπάρχουν πολλές τεχνικές κωδικοποίησης διόρθωσης λαθών (error correction coding techniques) οι οποίες μπορούν να ανακτήσουν λάθη σε μερικά bit, και η απόδοση σ' ένα σύστημα end-to-end δεν μειώνεται σημαντικά από λάθη σε μερικά bits, έτσι η μέση πιθανότητα λάθους είναι ένα καλό μέγεθος για να μετρηθεί η ποιότητα του καναλιού κάτω από τις δεδομένες συνθήκες.

Στη περίπτωση που η ισχύς του καναλιού μεταβάλλεται αργά, δηλαδή ($T_s \ll T_c$), τότε μία μεγάλη εξασθένιση θα επηρεάσει ταυτόχρονα πολλά σύμβολα. Έτσι, η εξασθένιση (fading), οδηγεί σε μεγάλο αριθμό λαθών τύπου burst, τα οποία δεν μπορούν να διορθωθούν με coding λογικής πολυπλοκότητας. Ως εκ τούτου αυτού

του τύπου τα λάθη μπορούν να μειώσουν εξαιρετικά σημαντικά την συνολική απόδοση του συστήματος.

Σ' αυτή τη περίπτωση δεν υπάρχει εγγύηση για αποδεκτή απόδοση του συστήματος καθ' όλη τη διάρκεια της λειτουργίας χωρίς δραματική αύξηση της ισχύς εκπομπής.

Υπό αυτές τις συνθήκες, ορίζεται, μία outage πιθανότητα θεωρώντας ότι το κανάλι για κάποιο ποσοστό χρόνου δεν είναι λειτουργικό. Η outage και η μέση πιθανότητα λάθους συχνά, συνδυάζονται όταν το κανάλι μοντελοποιείται ως συνδυασμός γρήγορου και αργού fading, π.χ log-normal σκίαση με fast Rayleigh fading. Όταν το $T_c \ll T_s$, το fading θα είναι εκτός μέσου όρου από το matched filter στον αποδιαλωτή. Για πολύ γρήγορη εξασθένιση (fast fading), η απόδοση γίνεται ίδια μ' αυτή ενός AWGN καναλιού.

4.3.1 Outage Probability

Η outage πιθανότητα σε σχέση με το γ_0 ορίζεται ως

$$P_{out} = P(\gamma_s < \gamma_0) = \int_0^{\gamma_0} p_{\gamma_s}(\gamma) d\gamma \quad (4.27)$$

Όπου γ_0 τυπικά ορίζεται ως η ελάχιστη τιμή του SNR που απαιτείται για αποδεκτή απόδοση. Για παράδειγμα, θεωρώντας ψηφιοποιημένη φωνή, η $P_b = 10^{-3}$ είναι αποδεκτή τιμή, από τη στιγμή που γενικά δεν γίνεται αντιληπτή από το ανθρώπινο αυτί. Για ένα BPSK σήμα σε Rayleigh fading, $\gamma_b < 7$ dB θα δηλώνει την outage πιθανότητα, και έτσι θέτουμε $\gamma_0 = 7$ dB.

Σε Rayleigh fading η outage πιθανότητα γίνεται

$$P_{out} = \int_0^{\gamma_0} \frac{1}{\gamma_s} e^{-\frac{\gamma_s}{\gamma_s}} d\gamma_s = 1 - e^{-\frac{\gamma_0}{\gamma_s}} \quad (4.28)$$

Αντιστρέφοντας τον παραπάνω τύπο και για μία δεδομένη outage πιθανότητα, μπορούμε να βρούμε το απαιτούμενο μέσο SNR $\overline{\gamma_s}$ το οποίο είναι

$$\overline{\gamma_s} = \frac{\gamma_0}{-\ln(1 - P_{out})} \quad (4.29)$$

Σε dB αυτό σημαίνει ότι $10 \log \overline{\gamma_s}$ θα πρέπει να υπερβαίνει το στόχο $10 \log \gamma_0$ κατά $F_d = -10 \log[-\ln(1 - P_{out})]$ ώστε να συντηρήσουμε την απόδοση περισσότερο από $100 * (1 - P_{out})$ ποσοστό του χρόνου. Η ποσότητα F_d ονομάζεται **fade margin** και δίνεται σε db.

4.3.2 Μέση Πιθανότητα Λάθους (Average Probability of Error)

Η μέση πιθανότητα λάθους χρησιμοποιείται ως ένα μέτρο της απόδοσης όταν $T_s \approx T_c$. Εδώ θεωρούμε ότι η γ_s είναι σταθερή για την διάρκεια του συμβόλου. Η μέση πιθανότητα σφάλματος υπολογίζεται, ολοκληρώνοντας την πιθανότητα σφάλματος σε AWGN με fading distribution:

$$\overline{P_s} = \int_0^{\infty} P_s(\gamma) p_{\gamma_s}(\gamma) d\gamma \quad (4.30)$$

Όπου το $P_s(\gamma)$ είναι η πιθανότητα λάθους σε σύμβολο σε AWGN με SNR γ , το οποίο μπορεί να υπολογιστεί.

Για μία δεδομένη κατανομή του πλάτους εξασθένισης r για παράδειγμα Rayleigh, Rician, log-normal, κλπ, υπολογίζουμε την $p_{\gamma_s}(\gamma)$ αλλάζοντας μεταβλητές ως

$$p_{\gamma_s}(\gamma) d\gamma = p(r) dr \quad (4.31)$$

Για παράδειγμα, σε εξασθένιση Rayleigh το πλάτος του λαμβανόμενου σήματος r έχει κατανομή Rayleigh και δίνεται από τον ακόλουθο τύπο :

$$p(r) = \frac{r}{\sigma^2} e^{-\frac{r^2}{2\sigma^2}}, \quad r \geq 0 \quad (4.32)$$

Η ισχύς του σήματος είναι εκθετικά κατανομημένη με μέση τιμή $2\sigma^2$. Το SNR ανά σύμβολο για δεδομένο πλάτος r είναι :

$$\gamma = \frac{r^2 T_s}{2\sigma_n^2} \quad (4.33)$$

Όπου $\sigma_n^2 = \frac{N_0}{2}$ είναι η PSD μεταβλητή του θορύβου και στους δύο κλάδους συμφασικούς (in-phase) και ορθογώνιους (quadrature) . Διαφορίζοντας και τις δύο πλευρές του τύπου (4.33)

$$d\gamma = \frac{r T_s}{\sigma_n^2} dr \quad (4.34)$$

Αντικαθιστώντας την (4.33) και (4.34) στην (4.32) τότε η (4.31) γίνεται

$$p_{\gamma_s}(\gamma) = \frac{\sigma_n^2}{\sigma^2 T_s} e^{-\frac{\gamma \sigma_n^2}{\sigma^2 T_s}} \quad (4.35)$$

Το μέσο SNR ανά σύμβολο $\overline{\gamma_s}$ είναι ίσο με $\frac{\sigma^2 T_s}{\sigma_n^2}$ και μπορούμε να ξαναγράψουμε

την 4.35 ως :

$$p_{\gamma_s}(\gamma) = \frac{1}{\gamma_s} e^{-\frac{\gamma}{\gamma_s}} \quad (4.36)$$

η οποία είναι εκθετική κατανομή.

Για δυαδικά σήματα έχουμε :

$$p_{\gamma_b}(\gamma) = \frac{1}{\gamma_b} e^{-\frac{\gamma}{\gamma_b}} \quad (4.37)$$

Ολοκληρώνοντας την 4.30 στην 4.37, έχουμε την μέση πιθανότητα σφάλματος για σήματα BPSK σε εξασθένιση Rayleigh

$$\text{BPSK} \quad \overline{P_b} = \frac{1}{2} \left[1 - \sqrt{\frac{\gamma_b}{1 + \gamma_b}} \right] \approx \frac{1}{4\gamma_b} \quad (4.38)$$

Όπου η προσέγγιση ισχύει για μεγάλες τιμές του $\overline{\gamma_b}$.

Ανάλογα έχουμε προσέγγιση και για την QPSK, όπου η μέση πιθανότητα λάθους για Rayleigh Fading δίνεται από τον ακόλουθο τύπο :

$$\text{QPSK} \quad \overline{P_b} \approx \frac{2}{3} \left[1 - \frac{1}{4} \left(1 + \sqrt{\frac{\gamma_b}{1 + \gamma_b}} \right)^2 \right] \approx \frac{1}{3\gamma_b} \quad (4.39)$$

Όπου πάλι η προσέγγιση ισχύει για μεγάλες τιμές του $\overline{\gamma_b}$.

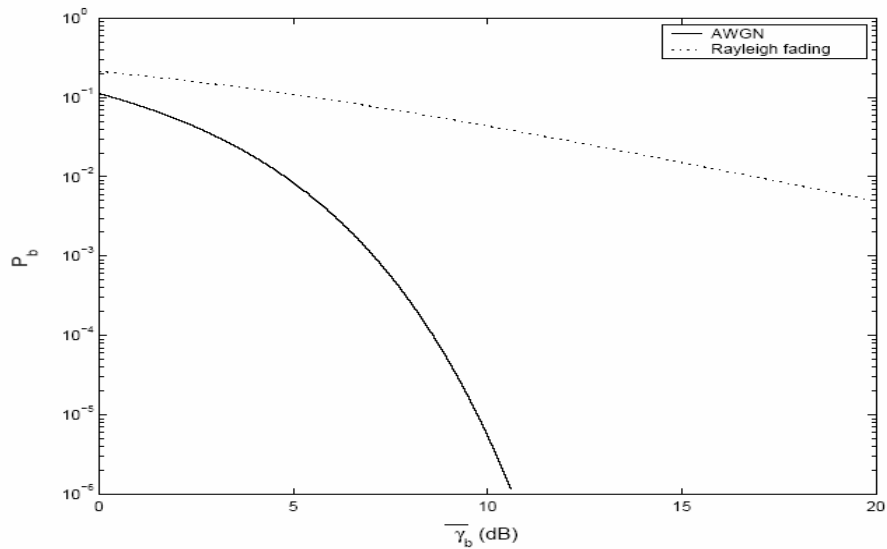
Είναι ενδιαφέρον να συγκρίνουμε την πιθανότητα σφάλματος Bit (bit error probability) στα διαφορετικά σχήματα διαμορφώσεων σε AWGN και fading.

Εάν χρησιμοποιήσουμε την γενική προσέγγιση $P_s \approx a_M Q(\sqrt{\beta_M \gamma_s})$ τότε η μέση πιθανότητα σφάλματος για σύμβολο σε Rayleigh fading μπορεί να προσεγγιστεί ως

$$\overline{P_s} \approx \int_0^\infty a_M Q(\sqrt{\beta_M \gamma}) \cdot \frac{1}{\gamma_s} e^{-\frac{\gamma}{\gamma_s}} d\gamma_s = \frac{\alpha_m}{2} \left[1 - \sqrt{\frac{0.5 \beta_m \overline{\gamma_s}}{1 + 0.5 \beta_m \overline{\gamma_s}}} \right] \approx \frac{\alpha_m}{2 \beta_m \overline{\gamma_s}} \quad (4.40)$$

Όπου η τελευταία προσέγγιση είναι στο όριο του υψηλότερου SNR.

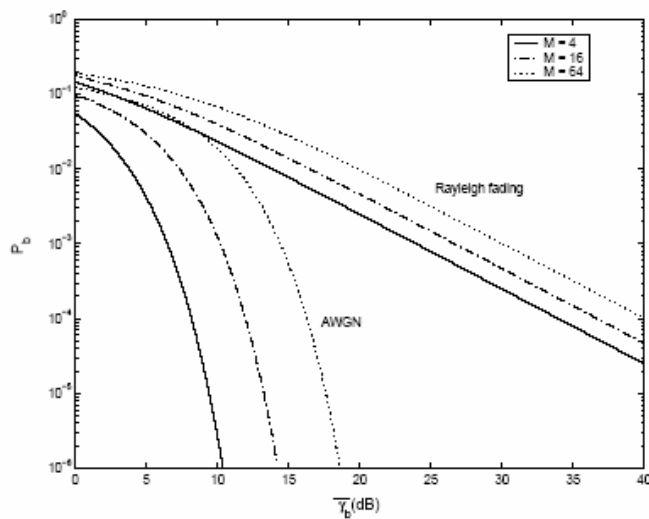
Στο σχήμα 4.1 σχεδιάσαμε την πιθανότητα σφάλματος (error probability) ενός BPSK σήματος σε AWGN και σε flat Rayleigh fading.



Σχήμα 4.1 : Μέση P_b για BPSK σε Rayleigh Fading & AWGN

Παρατηρούμε ότι απαιτείται περίπου 8db SNR για να έχουμε 10^{-3} bit error rate σε AWGN , ενώ απαιτείται περίπου 24 dB SNR για να διατηρήσουμε το ίδιο ρυθμό σφάλματος (error rate) σε fading.

Ανάλογο σχήμα (Σχ. 4.2) έχουμε για error probabilities για διαμόρφωση MQAM , το οποίο βασίζεται στη προσέγγιση της 4.23 και της 4.40



Σχήμα 4.2 Μέση P_b για MQAM σε Rayleigh Fading & AWGN

Από τα παραπάνω σχήματα είναι φανερό ότι για να διατηρήσουμε χαμηλή ισχύ απαιτούνται κάποιες τεχνικές για να αναιρέσουμε την επίδραση του φαινομένου της εξασθένησης. Μερικές από τις τεχνικές αυτές είναι η diversity combining, spread spectrum, RAKE receivers.

Η Rayleigh fading είναι μία από τις χειρότερες περιπτώσεις εξασθένησης .

4.4 Moment Generating Function Approach to Average Error Probability

Η **moment generating function** (MGF) για μία μη αρνητική τυχαία μεταβλητή γ με pdf $p_\gamma(\gamma)$, $\gamma \geq 0$, ορίζεται ως

$$M_\gamma(s) = \int_0^{\infty} P_\gamma(\gamma) e^{s\gamma} d\gamma \quad (4.41)$$

Αυτή η συνάρτηση είναι ο μετασχηματισμός Laplace της pdf $p_\gamma(\gamma)$ με ανεστραμμένο το πρόσημο : $L[p_\gamma(\gamma)] = M_\gamma(-s)$. Έτσι η MGF για τις περισσότερες ενδιαφέρουσες κατανομές εξασθένησης - fading distributions –μπορεί να υπολογιστεί είτε σε κλειστή μορφή χρησιμοποιώντας τον κλασικό μετασχηματισμό Laplace ή μέσω αριθμητικών ολοκληρώσεων. Συγκεκριμένα η MGF για τις διάφορες κατανομές fading δίνεται από τους ακόλουθους τύπους

Rayleigh:

$$\text{MGF } M_{\gamma_s}(s) = (1 - s\overline{\gamma}_s)^{-1} \quad (4.42)$$

Η moments $E[\gamma^n]$ του γ μπορεί να βρεθεί από την $M_\gamma(s)$ ως

$$E[\gamma^n] = \frac{\theta^n}{\theta^n} \left[M_{\gamma_s}(s) \right] \Big|_{s=0} \quad (4.43)$$

Η MGF είναι ένα πολύ ισχυρό εργαλείο στην ανάλυση της απόδοσης των διαμορφώσεων σε περιβάλλον εξασθένησης με ή χωρίς diversity.

Η βασικότερη υπόσχεση της MGF προσέγγισης για τον υπολογισμό της μέσης πιθανότητας σφάλματος σε fading είναι να εκφράσει την πιθανότητα του σφάλματος P_s σε AWGN για την διαμόρφωση ενδιαφέροντος σαν εκθετική συνάρτηση του γ_s ,

$$P_s = c_1 \exp[-c_2 \gamma_s] \quad (4.44)$$

Για σταθερά c_1 and c_2 , ή σαν οριστικό ολοκλήρωμα μίας τέτοιας εκθετικής

$$\text{συνάρτησης : } P_s = \int_A^B c_1 \exp[-c_2(x)\gamma] dx \quad (4.45)$$

Όπου η σταθερά $c_2(x)$ μπορεί να εξαρτάται από το ολοκλήρωμα αλλά το SNR γ δεν είναι στα όρια της ολοκλήρωσης. Αυτές οι φόρμες επιτρέπουν στη μέση πιθανότητα σφάλματος να εκφραστεί με όρους της MGF για την κατανομή fading.

Ειδικότερα εάν $P_s = a \exp[-\beta\gamma_s]$ τότε

$$\bar{P}_s = \int_0^{\infty} c_1 \exp[-c_2\gamma] P_{\gamma_s}(\gamma) d\gamma = c_1 M_{\gamma_s}(-c_2) \quad (4.46)$$

Για την περίπτωση της DPSK στη παραπάνω φόρμα το $c_1 = 1/2$ και $c_2 = 1$, έχουμε ότι η μέση πιθανότητα bit σφάλματος για οποιοδήποτε τύπο εξασθένισης είναι

$$\bar{P}_b = \frac{1}{2} M_{\gamma_s}(-1) \quad (4.47)$$

Όπου $M_{\gamma_s}(s)$ είναι η MGF της κατανομής εξασθένισης. Για παράδειγμα, χρησιμοποιώντας την $M_{\gamma_s}(s)$ για την κατανομή Rayleigh, η οποία δίνεται από την 4.42 με $s = -1$ δίνει $\bar{P}_b = [2(1 + \bar{\gamma}_b)]^{-1}$. Εφόσον η P_s δίνεται από το ολοκλήρωμα της 4.45 τότε

$$\bar{P}_s = \int_0^B \int_A c_1 \exp[-c_2(x)\gamma] p_{\gamma_s}(\gamma) d\gamma = c_1 \int_A^B \left[\int_0^{\infty} \exp[-c_2(x)\gamma] p_{\gamma_s}(\gamma) d\gamma \right] dx = c_1 \int_A^B M_{\gamma_s}(-c_2(x)) dx \quad (4.48)$$

Στη τελευταία περίπτωση, η μέση πιθανότητα σφάλματος συμβόλου είναι ένα μόνο οριστικό ολοκλήρωμα της MGF της κατανομής εξασθένισης, η οποία τυπικά μπορεί να εξαχθεί σε κλειστή μορφή ή ευκολότερα να αποτιμηθεί αριθμητικά. Παρακάτω θα εφαρμόσουμε την προσέγγιση της MGF για συγκεκριμένη διαμόρφωση και κατανομή εξασθένισης. Η γενική έκφραση για την P_s για την περίπτωση της συμφασικής διαμόρφωσης σε κανάλι με AWGN, με όρους της συνάρτησης Gaussian Q δίνεται από τον τύπο $P_s(\gamma_s) \approx a_M Q(\sqrt{\beta_M \gamma_s})$.

Στην παραπάνω θέτοντας $\alpha = a_M$ and $g = .5\beta_M$ έχουμε ότι :

$$P_s(\gamma) = a Q(\sqrt{2g\gamma_s}) \quad (4.49)$$

όπου a και g είναι σταθερές και εξαρτώνται μόνον από τη διαμόρφωση. Η αλλαγή έγινε για να ευρεθεί η πιθανότητα σφάλματος σαν μία ακριβή MGF.

Χρησιμοποιώντας την Q συνάρτηση έχουμε :

$$P_s = \frac{a}{\pi} \int_0^{\frac{\pi}{2}} \exp\left[-\frac{g\gamma}{\sin^2 \phi}\right] d\phi \quad (4.50)$$

η οποία είναι η επιθυμητή εξίσωση (4.45). Έτσι η μέση πιθανότητα σφάλματος σε εξασθένιση για διαμορφώσεις με $P_s = aQ(\sqrt{2g\gamma_s})$ σε AWGN δίνεται από

$$\overline{P_s} = \frac{\alpha}{\pi} \int_0^{\frac{\pi}{2}} \int_0^{\infty} \exp\left[-\frac{g\gamma}{\sin^2 \phi}\right] d\phi p_{\gamma_s} d\gamma = \frac{\alpha}{\pi} \int_0^{\frac{\pi}{2}} \int_0^{\infty} \exp\left[-\frac{g\gamma}{\sin^2 \phi}\right] p_{\gamma_s}(\gamma) d\gamma d\phi = \frac{\alpha}{\pi} \int_0^{\frac{\pi}{2}} M_{\gamma_s}\left(-\frac{g}{\sin^2 \phi}\right) d\phi \quad (4.51)$$

όπου η M_{γ_s} είναι η MGF η σχετική με την pdf $p_{\gamma_s}(\gamma)$ όπως ορίζεται στην (4.41). Οι προσεγγίσεις για την πιθανότητα σφάλματος σε AWGN για πολλές διαμορφώσεις ενδιαφέροντος με $P_s \approx aQ(\sqrt{2g\gamma_s})$ δίνονται στο Πίνακα 4.1, ενώ από την (4.51) έχουμε μία προσέγγιση για την μέση πιθανότητα σφάλματος αυτών των διαμορφώσεων σε fading. Επιπλέον η ακριβής μέση τιμή της πιθανότητας για σφάλμα συμβόλου για συμφασική MPSK μπορεί να δοθεί σε τύπο ανάλογο της (4.51). Έτσι η ακριβής μέση πιθανότητα σφάλματος για MPSK γίνεται :

$$\begin{aligned} \overline{P_s} &= \int_0^{\frac{(M-1)\pi}{2}} \frac{1}{\pi} \int_0^M \exp\left[-\frac{g\gamma_s}{\sin^2 \phi}\right] d\phi p_{\gamma_s} d\gamma = \frac{1}{\pi} \int_0^{\frac{(M-1)\pi}{2}} \left[\int_0^{\infty} \exp\left[-\frac{g\gamma_s}{\sin^2 \phi}\right] p_{\gamma_s}(\gamma) d\gamma \right] d\phi = \\ &= \frac{1}{\pi} \int_0^{\frac{(M-1)\pi}{2}} M_{\gamma_s}\left(-\frac{g}{\sin^2 \phi}\right) d\phi \end{aligned} \quad (4.52)$$

Όπου $g = \sin^2(\pi/M)$ και εξαρτάται από το μέγεθος του MPSK constellation. Η MGF $M_{\gamma_s}(s)$ για την κατανομή Rayleigh δίνεται από τον τύπο 4.42.

Αντικαθιστώντας

$s = -\frac{g}{\sin^2 \phi}$ στην έκφραση 4.42 έχουμε ότι για Rayleigh κατανομή ισχύει :

$$M_{\gamma_s}\left(-\frac{g}{\sin^2 \phi}\right) = \left(1 + \frac{g\gamma_s}{\sin^2 \phi}\right)^{-1}$$

Modulation	$P_s(\gamma_s)$	$P_b(\gamma_b)$
BFSK:		$P_b = Q(\sqrt{2\gamma_b})$
BPSK:		$P_b \approx Q(\sqrt{2\gamma_b})$
QPSK,4QAM:	$P_s \approx 2Q(\sqrt{\gamma_s})$	$P_b \approx Q(\sqrt{2\gamma_b})$
MPAM:	$P_s \approx \frac{2(M-1)}{M} Q\left(\sqrt{\frac{6\gamma_s}{M^2-1}}\right)$	$P_b \approx \frac{2(M-1)}{M \log_2 M} Q\left(\sqrt{\frac{6\gamma_b \log_2 M}{M^2-1}}\right)$
MPSK:	$P_s \approx 2Q(\sqrt{3\gamma_s} \sin(\pi/M))$	$P_b \approx \frac{2}{\log_2 M} Q(\sqrt{2\gamma_b \log_2 M} \sin(\pi/M))$
Rectangular MQAM:	$P_s \approx \frac{4(\sqrt{M}-1)}{\sqrt{M}} Q\left(\sqrt{\frac{3\gamma_s}{M-1}}\right)$	$P_b \approx \frac{4(\sqrt{M}-1)}{\sqrt{M} \log_2 M} Q\left(\sqrt{\frac{3\gamma_b \log_2 M}{M-1}}\right)$
Nonrectangular MQAM:	$P_s \approx 4Q\left(\sqrt{\frac{3\gamma_s}{M-1}}\right)$	$P_b \approx \frac{4}{\log_2 M} Q\left(\sqrt{\frac{3\gamma_b \log_2 M}{M-1}}\right)$

Πίνακας 4.1 Προσεγγιστικοί τύποι για Symbol & Bit Error Probability για coherent modulations

ΚΕΦΑΛΑΙΟ 5

ΑΠΟΔΟΣΗ ΤΩΝ ΨΗΦΙΑΚΩΝ ΔΙΑΜΟΡΦΩΣΕΩΝ ΣΤΑ

ΑΣΥΡΜΑΤΙΚΑ ΚΑΝΑΛΙΑ ΜΕ CODING

5.1 Γενικά

Στο προηγούμενο κεφάλαιο , αναπτύξαμε την απόδοση που έχουν τα συστήματα , δηλαδή πως μεταβάλλεται το BER(Bit Error Rate) και το SER(Symbol Error Rate) σε σχέση με το SNR. Αυτό εξετάστηκε για διάφορες διαμορφώσεις, όπως για την BPSK, QPSK, MQAM . Όλα τα παραπάνω εξετάστηκαν με τις ακόλουθες παραδοχές , όσο αφορά το μοντέλο του καναλιού :

- (1) Κανάλια με θόρυβο AWGN
- (2) Κανάλια με flat fading
- (3) Χωρίς κωδικοποίηση

Στο παρόν κεφάλαιο , θα εξετάσουμε τις ανάλογες παραμέτρους , αλλά με την προϋπόθεση ότι τώρα εφαρμόζεται και κωδικοποίηση, δηλαδή έχουμε και channel coding. Έτσι ουσιαστικά θα εξετάσουμε την απόδοση της κωδικοποίησης , σε διάφορες διαμορφώσεις . Ειδικά για τους συνελκτικούς κώδικες (convolutional codes) η απόδοση αυτή ορίζεται κατά κύριο λόγο από το rate του κώδικα και το constraint length. Δεν έχει αξία να καθοριστεί η πιθανότητα λάθους στο block , από τη στιγμή που αυτό μπορεί να είναι πολύ μεγάλο. Έχει αξία , και είναι αυτό που είναι χρήσιμο να βρεθεί, η πιθανότητα σφάλματος στο bit , ή το bit error rate , δηλαδή ο ρυθμός σφάλματος των bit, ο οποίος είναι η μέση τιμή των bits που είναι λάθος για μία δοσμένη ακολουθία bits , διαιρούμενη με τον συνολικό αριθμό των bits του μηνύματος στην ακολουθία.

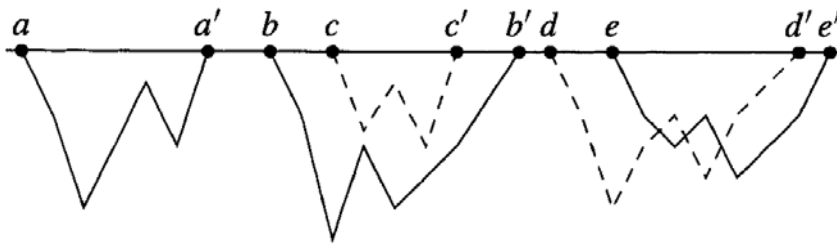
Ορίζουμε αυτή την πιθανότητα ως P_b . Σ' αυτό το κεφάλαιο θα αναπτύξουμε το άνω όριο της P_b .

5.2 Ανάλυση Λάθους για convolutional codes

Ας εξετάσουμε πόσα λάθη μπορούν να συμβούν κατά την διαδικασία της αποκωδικοποίησης. Ο μηχανισμός απόφασης του viterbi αλγορίθμου λειτουργεί όταν δύο διαδρομές ενώνονται. Εάν δύο διαδρομές ενώνονται και η διαδρομή με το

καλύτερο (χαμηλότερο) metric είναι στη πραγματικότητα η λανθασμένη διαδρομή , τότε μία μη σωστή απόφαση θα ληφθεί σ' αυτό το σημείο. Καλούμε ένα τέτοιο λάθος , node error και λέμε ότι ένα τέτοιο λάθος συμβαίνει τη στιγμή που οι διαδρομές για πρώτη φορά αποκλίνουν. Δηλώνουμε την πιθανότητα του node error ως P_e . Ένα node error μπορεί με τη σειρά του να οδηγήσει , σ' έναν αριθμό εισερχομένων Bits να αποκωδικοποιηθούν λανθασμένα.

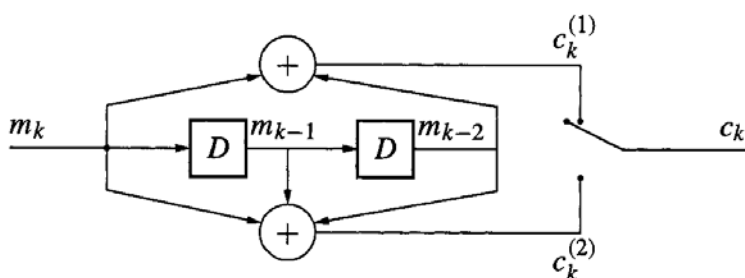
Από τη στιγμή που ο κώδικας είναι γραμμικός , είναι επαρκές να υποθέσουμε ότι η all-zero codeword στέλνεται : με $d_H(r,c)$ την απόσταση Hamming μεταξύ c , r , έχουμε ότι $d_H(r,c)=d_H(r+c,c+c)=d_H(r+c,0)$. Ας θεωρήσουμε τα λάθη που συμβαίνουν όπως αυτά απεικονίζονται στο Σχήμα 5.1



Σχήμα 5.1 Εμφάνιση σφαλμάτων εξαιτίας της συγχώνευσης διαδρομών

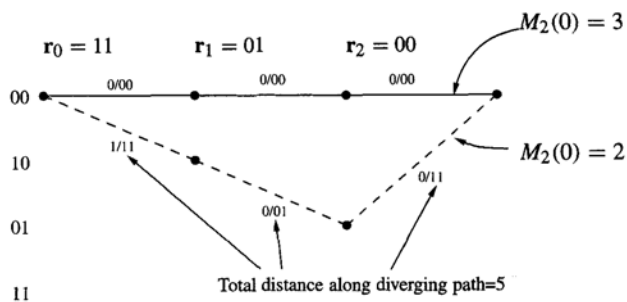
Η οριζόντια γραμμή του σχήματος αναπαριστά την διαδρομή all-zero του trellis διαγράμματος. Υποθέτουμε ότι η αποκλίνουσα διαδρομή από την διαδρομή all-zero στο a έχει την καλύτερη (χαμηλότερη) metric όταν οι διαδρομές συγχωνεύονται στο a' . Αυτό δίνει ένα σφάλμα στο a . Υποθέτουμε ότι υπάρχουν επίσης σφάλματα στο b και d . Αν δούμε τη διαδρομή που αποκλίνει στο c : ακόμα και εάν το Metric είναι χαμηλότερο (καλύτερο) στο c' , η αποκλίνουσα διαδρομή από το c μπορεί τελικά να μην επιλεγεί εάν η Metric είναι χειρότερη από τη διαδρομή που ενώνεται στο b . Ομοια η διαδρομή που ενώνεται στο b , μπορεί να μην επιλεγεί απαραίτητα επειδή η διαδρομή που ενώνεται στο e θα πάρει προτεραιότητα . Αυτή η επικάλυψη των αποφάσεων των διαδρομών , κάνει την ακριβή ανάλυση του bit error rate δύσκολη. Θα πρέπει να αρκεστούμε σε όρια και προσεγγίσεις.

Παρακάτω θα δώσουμε ένα παράδειγμα για να δείξουμε την αλήθεια των παραπάνω. Ας θεωρήσουμε τον παρακάτω κώδικα με $G(x) = [1+x^2, 1+x+x^2]$

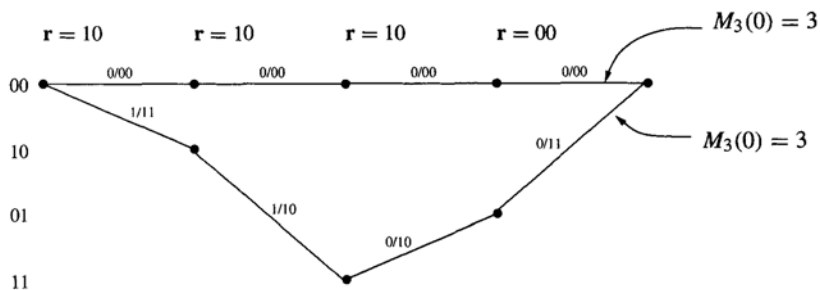


Σχήμα 5.2 Convolutional Encoder rate R=1/2

Υποθέτουμε ότι η ακολουθία των bit εισόδου είναι $x=[0,0,0,0,\dots]$, και η μεταδιδόμενη ακολουθία αντίστοιχα είναι $c=[00,00,00,00,\dots]$, αλλά η λαμβανόμενη ακολουθία, μέσω ενός καναλιού BSC (Binary Symmetric Channel) είναι $r=[11,01,00,\dots]$. Ένα τμήμα του Trellis διαγράμματος δεικνύεται στο παρακάτω σχήμα (Σχήμα 5.3) Μετά από τρία στάδια του trellis, όταν οι διαδρομές συγχωνεύονται, το metric για τη χαμηλότερη διαδρομή (δεικνύεται με διακεκομμένη γραμμή) είναι χαμηλότερο από το metric όλων των zero διαδρομών (σταθερές γραμμές).



(a) Diverging path of distance 5.



(b) Diverging path of distance 6.

Σχήμα 5.3 Decoding

Ο αλγόριθμος Viterbi επιλέγει, την λανθασμένη διαδρομή, όπως φαίνεται, και έτσι έχουμε ένα node error στο πρώτο κόμβο. Τα αποτελέσματα της απόφασης αυτής οδηγούν σε τρεις λανθασμένους κλάδους της διαδρομής στο trellis, η ακολουθία εισόδου ανταποκρίνεται στην επιλεγμένη διαδρομή [1,0,0] και έτσι μόνο ένα Bit είναι λανθασμένο κατά την αποκωδικοποίηση εξαιτίας της απόφασης αυτής. Όπως φαίνεται και στο σχήμα (5.3) υπάρχει μία διαδρομή με Metric 5 η οποία παρεκκλίνει από την διαδρομή all-zero. Η πιθανότητα της λανθασμένης απόφασης για την επιλογή αυτής της διαδρομής δηλώνεται ως P_5 . Αυτό το λάθος θα συμβεί όταν η λαμβανόμενη ακολουθία έχει τρία ή περισσότερα λάθη (1s) σ' αυτήν. Γενικά δηλώνουμε ως

P_d την πιθανότητα της λανθασμένης αποκωδικοποίησης της διαδρομής με Metric =d.

Για μία παρεκκλίνουσα διαδρομή περιττού βάρους, θα υπάρχει λάθος, εάν περισσότερα από τα μισά Bits είναι λάθος. Η πιθανότητα αυτή για ένα BSC κανάλι με crossover probability p_c είναι :

$$P_d = \sum_{i=(d+1)/2}^d \binom{d}{i} p_c^i (1-p_c)^{d-i} \quad \text{με } d \text{ περιττό} \quad (5.1)$$

Υποθέτουμε, ότι το λαμβανόμενο σήμα είναι το $r=[10,10,10,00]$ και το trellis διάγραμμα είναι αυτό του σχήματος 5.3b. Σ' αυτή τη περίπτωση τα path metrics είναι ίσα, και το μισό χρόνο ο decoder επιλέγει την λάθος διαδρομή. Εάν επιλεγεί η λανθασμένη διαδρομή τα εισερχόμενα bits στον κωδικοποιητή θα είναι [1,1,0,0], με δύο bits αποκωδικοποιημένα λάθος. Η πιθανότητα του γεγονότος να επιλεγεί η λανθασμένη διαδρομή στη περίπτωση αυτή είναι P_6 και η P_d ίση με :

$$P_d = \frac{1}{2} \binom{d}{d/2} p_c^{d/2} (1-p_c)^{d/2} + \sum_{i=d/2+1}^d \binom{d}{i} p_c^i (1-p_c)^{d-i} \quad \text{με } d \text{ άρτιο} \quad (5.2)$$

Η P_d είναι η πιθανότητα περισσότερα από τα μισά bits να είναι λάθος συν την μισή (1/2) πιθανότητα ότι ακριβώς τα μισά bits είναι λάθος.

Από το παραπάνω παράδειγμα μπορούμε να εξάγουμε τις ακόλουθες πληροφορίες :

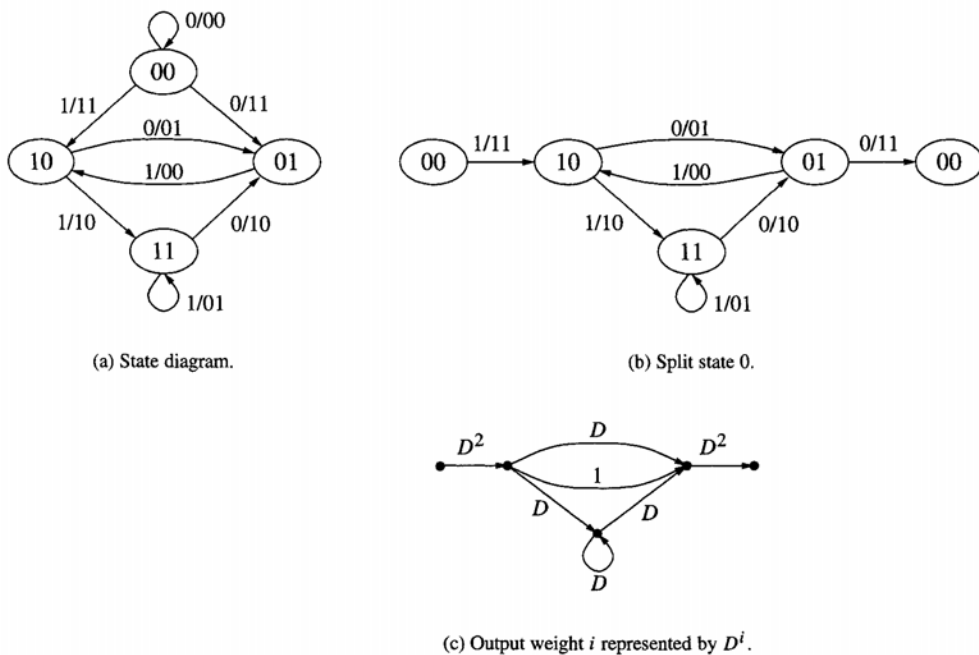
- Σφάλματα μπορούν να συμβούν στον αλγόριθμο αποκωδικοποίησης όταν οι διαδρομές ενώνονται. Εάν η λανθασμένη διαδρομή έχει χαμηλότερο (καλύτερο) Path metric απ' ότι η σωστή διαδρομή, ο αλγόριθμος την επιλέγει.

- Συνενώσεις διαδρομών μπορεί να είναι από διαφορετικά μήκη (αριθμός σταδίων)
- Όταν επιλέγεται μία λάθος διαδρομή, ο αριθμός των bits εισόδου που έχουν αποκωδικοποιηθεί λανθασμένα εξαρτάται από τη συγκεκριμένη διαδρομή.
- Η πιθανότητα του συγκεκριμένου λάθους υπολογίζεται και ορίζεται ως P_d .

5.3 Απαρίθμηση των Διαδρομών μέσα στο Trellis

Κατά τον υπολογισμό της ολικής πιθανότητας σφάλματος του decoder, είναι σκόπιμο να έχουμε μία μέθοδο για την απαρίθμηση όλων των διαδρομών στο trellis. Από τη στιγμή που ο convolutional code είναι γραμμικός είναι επαρκές και ικανοποιητικό να θεωρήσουμε μόνον εκείνες τις διαδρομές οι οποίες αποκλίνουν από την διαδρομή zero και μετά επανενώνονται για να υπολογίσουμε την πιθανότητα σφάλματος.

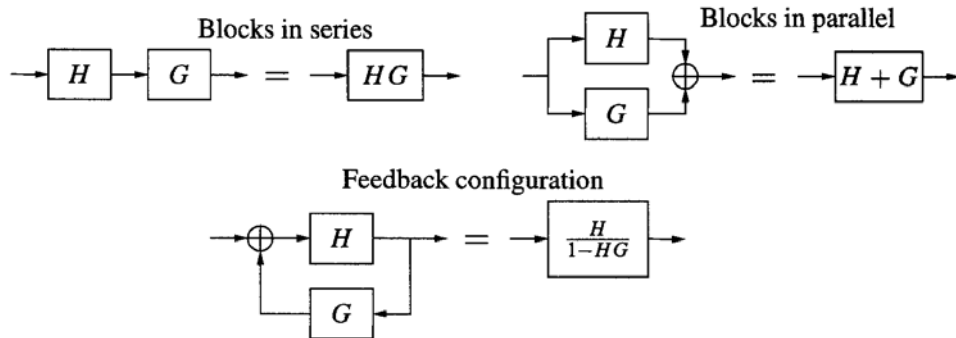
Εχει αναπτυχθεί μία μέθοδος συνάρτησης μεταφοράς, η οποία απαριθμεί όλες τις διαδρομές οι οποίες αποκλίνουν από τη διαδρομή all-zero και μετά επανενώνονται. Αυτή η συνάρτηση μεταφοράς καλείται απαριθμητής διαδρομής (path enumerator). Θα εξετάσουμε αυτήν τεχνική στο παράδειγμα που ήδη έχουμε δώσει και για τον encoder του σχήματος 5.2. Το Σχήμα 5.4 δείχνει το διάγραμμα κατάστασης γι' αυτόν τον encoder.



Σχήμα 5.4 Διάγραμμα κατάστασης και γράφημα για αποκλίνουσες /συγκλίνουσες διαδρομές

Στο παραπάνω σχήμα (5.4b), η κατάσταση 00 έχει χωριστεί, η αντιγραφεί. Η μετάβαση από την κατάσταση 00 στην κατάσταση 00 έχει παραληφθεί, από τη στιγμή που ενδιαφερόμαστε για διαδρομές οι οποίες αποκλίνουν από την κατάσταση 00. Οποιοδήποτε διαδρομή του σχήματος 5.4b από τον κόμβο 00 στα αριστερά στον κόμβο 00 στα δεξιά αναπαριστά μία διαδρομή η οποία αποκλίνει από τη διαδρομή zero και μετά επανενώνεται. Στο σχήμα 5.4c η codeword εξόδου βάρους i κατά μήκος κάθε άκρου, αναπαρίσταται με το D^i . Για παράδειγμα μία έξοδος 11 αναπαρίσταται με D^2 , ενώ μία έξοδος 10 με $D^1=D$ και μία 00 με $D^0=1$.

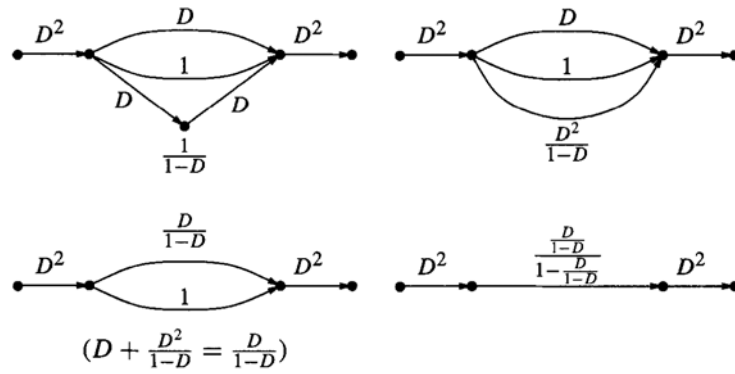
Τα αναγραφόμενα γράμματα στο γράφημα είναι οι συναρτήσεις μεταφοράς. Για την απλοποίηση του διαγράμματος ροής, εφαρμόζουμε του συμβατικούς κανόνες, όπως αυτοί δεικνύονται στο σχήμα 5.5



Σχήμα 5.5 Κανόνες για απλοποίηση του διαγράμματος ροής.

- Blocks στη σειρά πολλαπλασιάζουν τις συναρτήσεις μεταφοράς.
- Blocks παράλληλα προσθέτουν τις συναρτήσεις μεταφοράς.
- Blocks σε feedback υπηρετούν τον κανόνα «forward gain over 1 minus loop again»

Για το διάγραμμα κατάστασης του σχήματος 5.4 παίρνουμε κάθε κόμβο σαν ένα αθροιστικό κόμβο. Η ακολουθία των βημάτων από την επιτυχή εφαρμογή των κανόνων απλοποίησης δεικνύεται στο σχήμα 5.6



Σχήμα 5.6 Βήματα απλοποίησης για το διάγραμμα ροής ενός convolutional code

Απλοποιώντας το τελικό διάγραμμα , έχουμε ότι :

$$T(D) = D^2 \frac{\frac{D}{1-D}}{1 - \frac{D}{1-D}} D^2 = \frac{D^5}{1-2D}$$

Χρησιμοποιώντας την $\frac{1}{1-D} = 1 + D + D^2 + D^3 + \dots$ έχουμε ότι :

$$T(D) = D^5 (1 + 2D + (2D)^2 + (2D)^3 + \dots) = D^5 + 2D^6 + 4D^7 + \dots + 2^k D^{k+5} + \dots$$

Μεταφράζοντας την παραπάνω βλέπουμε ότι :

- Μία αποκλίνουσα/συγκλίνουσα διαδρομή λάθους με Metric 5 από την διαδρομή zero .
- Δύο λανθασμένες διαδρομές με metric 6
- Τέσσερις λανθασμένες διαδρομές με metric 5 .
- Η συντομότερη λανθασμένη διαδρομή έχει metric 5

Η ελάχιστη Metric μίας διαδρομής αποκλίνουσας από και μετά συνενώνοντας με την zero διαδρομή καλείται free distance ενός convolutional code , και δηλώνεται ως d_{free} . Ο αριθμός των διαδρομών μ' αυτό το metric δηλώνεται ως N_{free} .

Γενικά έχουμε

$$T(D) = \sum_{d=d_{free}}^{\infty} a(d)D^d \quad \text{όπου} \quad a(d_{free}) = N_{free}$$

5.4 Node Error Probability P_e και Bit Error Rate P_b

Θεωρούμε ότι ως P_j δηλώνεται το σύνολο των λανθασμένων διαδρομών οι οποίες αποκλίνουν από τον κόμβο j απ' όλη τη zero διαδρομή στο Trellis και μετά επανενώνονται και κάθε μία απ' αυτές τις διαδρομές δηλώνεται ως $p_{i,j} \in P_j$. Επίσης θεωρούμαι ότι το $\Delta M(p_{i,j}, 0)$ δηλώνει την διαφορά ανάμεσα στην υπολογισμένη metric κατά μήκος της διαδρομής $p_{i,j}$ και όλης της zero διαδρομής. Λάθος συμβαίνει στον κόμβο j εξαιτίας της διαδρομής $p_{i,j}$ εάν $\Delta M(p_{i,j}, 0) < 0$. Το $P_e(j)$ δηλώνει την πιθανότητα της εμφάνισης σφάλματος στο κόμβο j και ισχύει ότι :

$$P_e(j) \leq P_r \left[\bigcup_i \{ \Delta M(p_{i,j}, 0) \leq 0 \} \right] \quad (5.3)$$

Η ανισότητα στην παραπάνω ισχύει δεδομένου ότι σφάλμα μπορεί να μην συμβεί όταν $\Delta M(p_{i,j}, 0) = 0$. Οι διαδρομές $p_{i,j} \in P$ δεν είναι όλες ασύνδετες από τη στιγμή που μπορεί να μοιράζονται κλάδους, έτσι το γεγονός ότι $\{ \Delta M(p_{i,j}, 0) \leq 0 \}$ ισχύει. Αυτό όμως οδηγεί στο γεγονός ότι ο ακριβής υπολογισμός της 5.3 είναι πολύ δύσκολος. Μπορεί όμως να δοθεί ως :

$$P_e(j) \leq \sum_i P_r(\Delta M(p_{i,j}, 0) \leq 0) \quad (5.4)$$

Κάθε όρος στο παραπάνω άθροισμα είναι ένα γεγονός μεταξύ των διαδρομών $p_{i,j}$ και της zero. Η 5.4 ισχύει για την περίπτωση που το κανάλι είναι «χωρίς μνήμη» (memoryless). Για ένα τέτοιο κανάλι έχουμε ότι η $\Delta M(p_{i,j}, 0)$ εξαρτάται μόνον από εκείνους τους κλάδους για τους οποίους η $p_{i,j}$ είναι μη μηδενική. Θεωρούμαι ότι d είναι το βάρος Hamming του κάθε $p_{i,j}$ και P_d η πιθανότητα του γεγονότος ότι η διαδρομή αυτή έχει το χαμηλότερο (καλύτερο) metric απ' όλη τη μηδενική (zero) διαδρομή. Θεωρώντας $a(d)$ να είναι ο αριθμός των διαδρομών σε απόσταση d από την μηδενική, η πιθανότητα εμφάνισης σφάλματος στον κόμβο μπορεί να ξαναγραφεί ως εξής :

$$\begin{aligned} P_e(j) &\leq \sum_{d=d_{free}}^{\infty} P_r(\text{σφάλμα που προκαλείται από οποιοδήποτε } a(d) \text{ λανθασμένη διαδρομή απόστασης } d) \\ &= \sum_{d=d_{free}}^{\infty} a(d) P_d \end{aligned} \quad (5.5)$$

Κάθε περαιτέρω χαρακτηρισμός της $P_e(j)$ απαιτεί χαρακτηρισμό της P_d . Θα δείξουμε παρακάτω ότι τα όρια της P_d μπορούν να δοθούν από την

$$P_d < Z^d \quad (5.6)$$

για κάποιο κανάλι – εξαρτημένη συνάρτηση Z , και για το BSC και AWGN ανάπτυξη ακριβών εκφράσεων. Στη παρούσα φάση, απλώς εκφράζουμε τα αποτελέσματα με όρους Z . Μ' αυτό το όριο μπορούμε να γράψουμε

$$P_e(j) < \sum_{d=d_{free}}^{\infty} a(d) Z^d$$

Γνωρίζοντας ότι ο απαριθμητής διαδρομών για τον κωδικοποιητή είναι

$$T(D) = \sum_{d=d_{free}}^{\infty} a(d) D^d \quad \text{έχουμε μία κλειστή φόρμα έκφρασης για το όριο:}$$

$$P_e(j) < T(D) \Big|_{D=Z} \quad (5.7)$$

Το όριο της 5.7 είναι το όριο της πιθανότητας του σφάλματος του κόμβου. Από αυτό μπορούμε να έχουμε ένα όριο για το bit error rate, απαριθμώντας τον αριθμό των Bits που είναι λάθος για κάθε λανθασμένο κόμβο.

Παίρνοντας την παράγωγο $\frac{\partial}{\partial N} T(D, N)$ έχουμε τον αριθμό των error bit κάθε error node της λανθασμένης διαδρομής.

Ο μέσος αριθμός των error bits, κατά μήκος των κλάδων του trellis είναι:

$$\frac{\partial}{\partial N} T(D, N) \Big|_{N=1, D=Z}$$

Για ένα κώδικα με rate $R=k/n$, κάθε κλάδος αντιστοιχεί σε k bits μηνύματος, και έτσι

$$P_b < \frac{1}{k} \frac{\partial}{\partial N} T(D, N) \Big|_{N=1, D=Z} \quad (5.8)$$

Μία προσέγγιση για την πιθανότητα του bit error rate μπορεί να αποκτηθεί αφήνοντας μόνον τους πρώτους όρους στη σειρά (5.8) και κάνοντας χρήση της

$$P_d < Z^d \quad \text{έχουμε:}$$

$$T(D, N) = D^{d_{free}} (N^{n_1} + N^{n_2} + \dots) + \dots$$

$$\frac{\partial}{\partial N} T(D, N) \Big|_{N=1} = D^{d_{free}} (n_1 + n_2 + \dots) + \dots$$

Ο αριθμός $n_1+n_2+\dots$ είναι ο αριθμός των μη μηδενικών bit που σχετίζονται με τις codewords βάρους d_{free} . Ο παραπάνω αριθμός $b_{d_{free}} = n_1+n_2+\dots$

5.5 Όριο για την P_d για Discrete Channels

Θεωρούμε την $p(r_i | 1)$ ως την πιθανότητα να λάβουμε το σήμα r_i , δοθέντος ότι το σύμβολο 1 έχει σταλθεί μέσω του καναλιού. Αντίστοιχα έχουμε για το σύμβολο 0 $p(r_i | 0)$. Τότε :

$$P_d = P(\Delta M(p_{i,j}, 0) \leq 0 \text{ και } d_H(p_{i,j}, 0) = d) = P\left[\sum_{i=1}^d \log P(r_i | 1) - \sum_{i=1}^d \log P(r_i | 0) \geq 0\right]$$

όπου $\{r_1, r_2, \dots, r_d\}$ είναι τα λαμβανόμενα σήματα στο d και $p_{i,j}$ είναι μη μηδενική.

Συνεχίζοντας ,

$$P_d = P\left[\sum_{i=1}^d \log \frac{p(r_i | 1)}{p(r_i | 0)} \geq 0\right] = P\left[\prod_{i=1}^d \frac{p(r_i | 1)}{p(r_i | 0)} \geq 1\right]$$

Αν θεωρήσουμε R' ένα ανύσμα που αποτελείται από τα στοιχεία

$r = (r_1, r_2, \dots, r_d)$ τέτοια ώστε :

$$\prod_{i=1}^d \frac{p(r_i | 1)}{p(r_i | 0)} \geq 1 \quad (5.9)$$

Για παράδειγμα , για $d=5$ πάνω στο BSC , R' είναι ένα σύνολο ανυσμάτων όπου το 3^ο , το 4^ο , το 5^ο από τα r_i είναι ίσα με 1. Η πιθανότητα οποιουδήποτε απ' αυτά τα στοιχεία είναι $\prod_{i=1}^d p(r_i | 0)$, από τη στιγμή που υποθέτουμε ότι στέλνονται 0 . Ετσι η πιθανότητα του κάθε ανύσματος R' είναι $\prod_{i=1}^d p(r_i | 0)$. Η πιθανότητα P_d μπορεί να αποκτηθεί προσθέτοντας όλους τα ανύσματα στο R' :

$$P_d = \sum_{r \in R'} \prod_{i=1}^d p(r_i | 0)$$

Εξαιτίας του γεγονότος ότι το αριστερό μέρος της 5.9 είναι ≥ 1 έχουμε

$$P_d \leq \sum_{r \in R'} \prod_{i=1}^d p(r_i | 0) \left[\frac{p(r_i | 1)}{p(r_i | 0)} \right]^s \quad \text{για κάθε } s \text{ τέτοιο ώστε } 0 \leq s < 1. \text{ Ένα πιο}$$

αυστηρό όριο μπορεί να αποκτηθεί μειώνοντας σε σχέση με το s :

$$P_d \leq \min_{0 \leq s < 1} \sum_{r \in R'} \prod_{i=1}^d p(r_i | 0) \left[\frac{p(r_i | 1)}{p(r_i | 0)} \right]^s = \min_{0 \leq s < 1} \sum_{r \in R'} \prod_{i=1}^d p(r_i | 0)^{1-s} p(r_i | 1)^s$$

Προσθέτοντας μόνον το σύνολο των R , r_1, r_2, \dots, r_d , έχουμε

$$P_d < \min_{0 \leq s < 1} \sum_{r \in R} \prod_{i=1}^d p(r_i|0)^{1-s} p(r_i|1)^s$$

Η σειρά του αθροίσματος και του γινομένου μπορεί να αλλάξει και έτσι να έχουμε :

$$P_d < \min_{0 \leq s < 1} \prod_{i=1}^d \sum_{r_i} p(r_i|0)^{1-s} p(r_i|1)^s$$

Η παραπάνω είναι γνωστή ως Όριο Chernoff

5.6 Όριο για την P_d για BPSK σε κανάλι AWGN

Τα κωδικοποιημένα bits $c_i^{(i)}$ γίνονται mapping σ' ένα διάγραμμα αστερισμού διαμόρφωσης BPSK από $a_i^{(i)} = \sqrt{E_c} (2c_i^{(i)} - 1)$, και το E_c είναι η ενέργεια του κωδικοποιημένου σήματος με $E_c = RE_b$ και E_b είναι η ενέργεια ανά bit. Εάν στέλνεται η ακολουθία με όλα μηδενικά, τότε στέλνονται τα ακόλουθα πλάτη $(-\sqrt{E_c}, -\sqrt{E_c}, -\sqrt{E_c}, \dots)$. Μία ακολουθία η οποία αποκλίνει από αυτή τη διαδρομή σε d θέσεις είναι σε τετραγωνική απόσταση $2dE_c$ απ' αυτήν. Τότε η πιθανότητα P_d είναι η πιθανότητα όπου μία ακολουθία συμβόλων d αποκωδικοποιείται λανθασμένα, συγκρινόμενη με την ακολουθία των συνεχόμενων μηδενικών. Είναι το πρόβλημα της διάκρισης μεταξύ $p_1 = (-\sqrt{E_c}, -\sqrt{E_c}, -\sqrt{E_c}, \dots)$ από το $p_2 = (\sqrt{E_c}, \sqrt{E_c}, \dots)$ και το κάθε άνυσμα απ' αυτά έχουν d στοιχεία. Η Ευκλείδεια απόσταση μεταξύ δύο σημείων είναι

$$d_{Euclidean}(p_1, p_2) = 2[dE_c]^{1/2}$$

Η πιθανότητα της ανίχνευσης λάθους είναι

$$P_d = Q\left(\sqrt{\frac{2dE_c}{N_0}}\right)$$

Για να εκφράσουμε το παραπάνω σε μορφή Z^d , χρησιμοποιούμε το όριο

$$Q(x) < \frac{1}{2} e^{-\frac{x^2}{2}} \text{ και έτσι η } P_d < \frac{1}{2} e^{-d \frac{E_c}{N_0}}$$

Τότε η 5.7 και η 5.8 δίνουν

$$P_e(j) < \frac{1}{2} T(D) \Big|_{D=e^{-\frac{E_c}{N_0}}} \quad P_b < \frac{1}{2} \frac{1}{k} \frac{\partial}{\partial N} T(D, N) \Big|_{n=1, D=e^{-\frac{E_c}{N_0}}} \quad (5.10)$$

Ξέροντας ότι :

$$Q(\sqrt{x+y}) \leq Q(\sqrt{x})e^{-\frac{y}{2}} \quad x \geq 0, y \geq 0$$

Μπορούμε να γράψουμε για την P_d

$$\begin{aligned} P_d &= Q\left(\sqrt{\frac{2dE_c}{N_0}}\right) = Q\left(\sqrt{\frac{2d_{free}E_c}{N_0} + \frac{2(d-d_{free})E_c}{N_0}}\right) \leq Q\left(\sqrt{\frac{2d_{free}E_c}{N_0}}\right) e^{d_{free}E_c/N_0} e^{-dE_c/N_0} \\ &= Q\left(\sqrt{\frac{2d_{free}E_c}{N_0}}\right) e^{d_{free}E_c/N_0} (e^{-E_c/N_0})^d \end{aligned}$$

Η παραπάνω πιθανότητα ονομάζεται pairwise error probability επειδή είναι η πιθανότητα σφάλματος που σχετίζεται πάντα με την σύγκριση δύο διαδρομών που διαφέρουν κατά d bits .

Τότε η $P_e(j)$ δίνεται από την ακόλουθη σχέση

$$\begin{aligned} P_e(j) &\leq e^{d_{free}E_c/N_0} Q\left(\sqrt{\frac{2d_{free}E_c}{N_0}}\right) T(D) \Big|_{D=e^{-E_c/N_0}} \quad \text{και} \\ P_b &\leq \frac{1}{k} e^{d_{free}E_c/N_0} Q\left(\sqrt{\frac{2d_{free}}{N_0}}\right) \frac{\partial}{\partial N} T(D, N) \Big|_{N=1, D=e^{-E_c/N_0}} \end{aligned} \quad (5.11)$$

Επίσης ένα χαμηλό όριο για την P_b

$$P_b > \frac{1}{k} b_{d_{free}} Q\left(\sqrt{\frac{2d_{free}E_c}{N_0}}\right) \quad (5.12)$$

5.7 Asymptotic Coding Gain

Το χαμηλότερο όριο για την πιθανότητα του bit error rate ενός κωδικοποιημένου (coded) σήματος (για soft-decision decoding) είναι :

$$P_b > \frac{1}{k} a(d_{d_{free}}) n_{d_{free}} Q\left(\sqrt{\frac{2d_{free}E_c}{N_0}}\right)$$

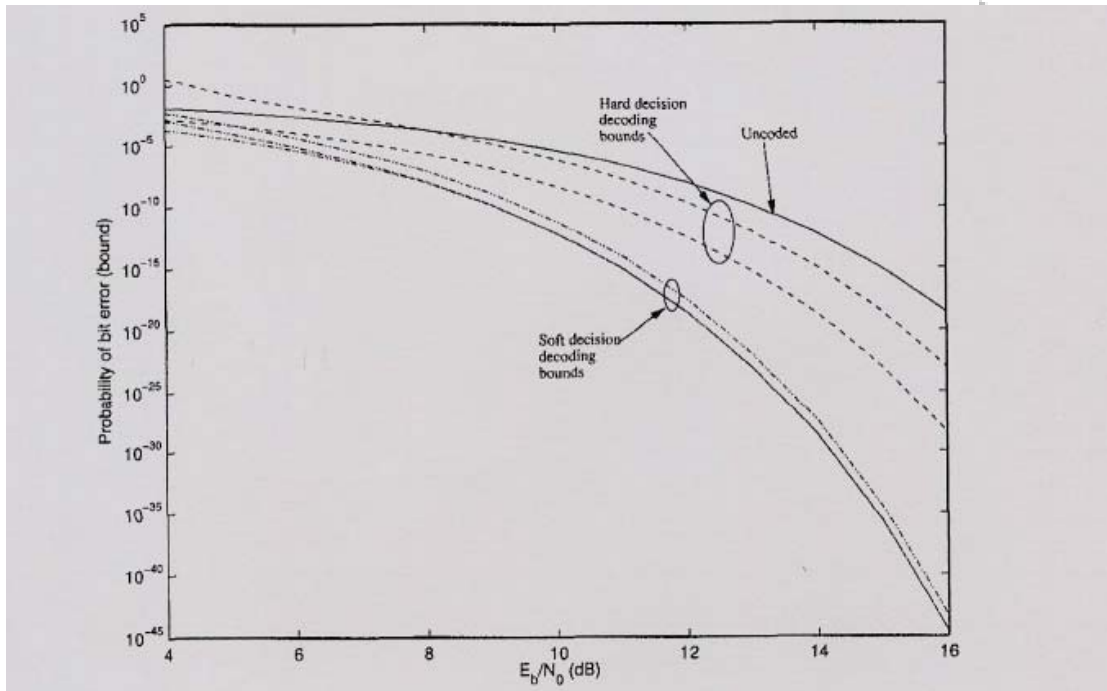
Και μπορεί να προσεγγιστεί χρησιμοποιώντας το όριο της $Q(x) < \frac{1}{2}e^{-x^2/2}$ και

έτσι να γίνει

$$P_b \approx \frac{1}{k} \frac{1}{2} a(d_{d_{free}}) n_{d_{free}} e^{-d_{free}E_c/N_0} \quad (5.13)$$

Η bit error πιθανότητα για uncoded μετάδοση είναι :

$$P_b = Q\left(\sqrt{\frac{2E_b}{N_0}}\right) < \frac{1}{2}e^{-E_b/N_0} \quad (5.14)$$



Σχήμα 5.7 Απόδοση ενός (3,1) convolutional code με $d_{free} = 5$

Ο κυρίαρχος παράγοντας στις δύο εξισώσεις (5.13 & 5.14) για μεγάλες τιμές του SNR καθορίζεται από τα εκθετικά. Συγκρίνοντας τα εκθετικά μέρη των δύο και χρησιμοποιώντας τη σχέση $E_c = RE_B$, βλέπουμε ότι το εκθετικό στην περίπτωση της πιθανότητας του bit error για την κωδικοποίηση είναι ένας παράγοντας του Rd_{free} μεγαλύτερος απ' ό τι στην περίπτωση του μη κωδικοποιημένου σήματος. Η ποσότητα

$$\gamma_{soft} = 10 \log_{10} Rd_{free}$$

Ονομάζεται asymptotic coding gain του κώδικα. Για αρκετά μεγάλο SNR, απόδοση ουσιαστικά ισοδύναμη μ' αυτή του uncoded μπορεί να αποκτηθεί με λιγότερο SNR κατά γ db απ' όταν έχουμε κώδικα. Ένα όμοιο όρισμα, δείχνει ότι το asymptotic coding gain για την περίπτωση της hard decision είναι ίσο με :

$$\gamma_{hard} = 10 \log_{10} \frac{Rd_{free}}{2}$$

Αυτό δείχνει ότι, η αποκωδικοποίηση με soft decision είναι κατά 3 db καλύτερη απ' ό τι με hard. Καθώς το SNR αυξάνει, ο κυρίαρχος όρος για τον υπολογισμό του bit

error rate είναι ο πρώτος όρος στην $T(x,N)$. Η free distance επηρεάζει σημαντικά την απόδοση του κώδικα.

5.8 Παράγοντες στην εφαρμογή των Convolutional Codes

Οι convolutional codes είναι ευρέως χρησιμοποιούμενοι στις σύγχρονες εφαρμογές στα συστήματα επικοινωνίας. Η αποκωδικοποίηση με Viterbi είναι δεσπόζουσα και χρησιμοποιείται για μικρά constraint length ($K \leq 10$), ενώ η sequential χρησιμοποιείται για τις περιπτώσεις που έχουμε κώδικες με μεγάλο constraint length και η πολυπλοκότητα του Viterbi γίνεται απαγορευτική. Η επιλογή του constraint length γίνεται με βάση το επιθυμούμενο coding gain.

Από την πιθανότητα σφάλματος για την soft decision decoding η οποία δίνεται

από την σχέση $P_b < \sum_{d=d_{free}}^{\infty} \beta_d Q(\sqrt{2\gamma_b R_c d})$ είναι φανερό ότι το coding gain που

επιτυγχάνεται από ένα convolutional code σ' ένα uncoded BPSK ή QPSK είναι :

$$\text{Coding Gain} \leq 10 \log_{10} (R_c d_{free})$$

Γνωρίζουμε επίσης ότι η ελάχιστη free distance d_{free} μπορεί να αυξηθεί είτε μειώνοντας την code rate ή αυξάνοντας το constraint length ή και τα δύο. Ο Πίνακας 5.1 παρέχει μία λίστα των άνω ορίων στο coding gain για μερικούς convolutional codes. Για λόγους σύγκρισης παρατίθεται και ο Πίνακας 5.2 ο οποίος δίνει πραγματικά coding gain για διάφορους convolutional code με μικρά constrain length και Viterbi decoding. Θα πρέπει να σημειωθεί ότι το coding gain αυξάνει άμεσα το ασυμπτωτικό όριο καθώς το SNR /bit αυξάνει.

Αυτά τα αποτελέσματα βασίζονται στο soft-decision Viterbi decoding. Εάν χρησιμοποιηθεί hard-decision decoding το coding gain μειώνεται προσεγγιστικά κατά 2 db για ένα AWGN κανάλι.

Μεγαλύτερα coding gain απ' αυτά που αναγράφονται στους Πίνακες 5.1 και 5.2, επιτυγχάνονται με convolutional codes οι οποίοι έχουν μεγάλα constraint length π.χ $K=50$, και αποκωδικοποίηση με sequential διαδικασία.

Rate $\frac{1}{2}$ codes			Rate $\frac{3}{4}$ codes		
Constrain Length K	d_{free}	Upper bound db	Constrain Length K	d_{free}	Upper bound db
3	5	3.98	3	8	4.26
4	6	4.77	4	10	5.23
5	7	5.44	5	12	6.02
6	8	6.02	6	13	6.37
7	10	6.99	7	15	6.99
8	10	6.99	8	16	7.27
9	12	7.78	9	18	7.78
10	12	7.78	10	20	8.24

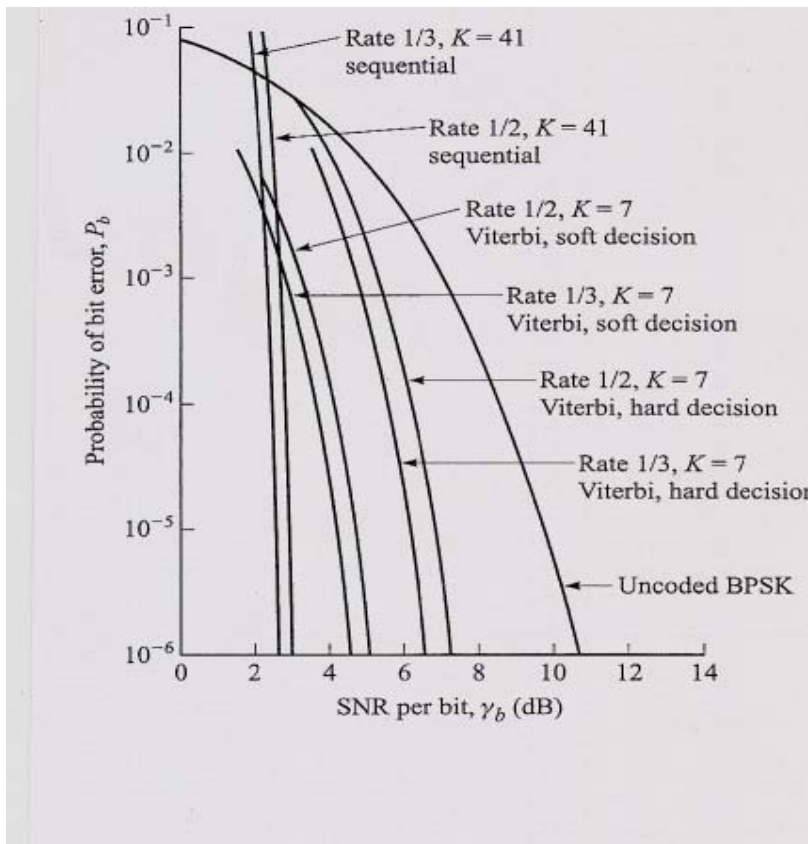
Πίνακας 5.1 Ανω όρια στο coding gain convolutional codes με soft decision

P_b	E_b/N_0 uncoded db	$R_c=1/3$		$R_c=1/2$			$R_c=2/3$		$R_c=3/4$	
		K=8	K=8	K=5	K=6	K=7	K=6	K=8	K=6	K=9
10^{-3}	6.8	4.2	4.4	3.3	3.5	3.8	2.9	3.1	2.6	2.6
10^{-5}	9.6	5.7	5.9	4.3	4.6	5.1	4.2	4.6	3.6	4.2
10^{-7}	11.3	6.2	6.5	4.9	5.3	5.8	4.7	5.2	3.9	4.8

Πίνακας 5.2 Coding gain (db) για Soft decision Viterbi decoding

Σταθεροί, sequential decoders, αναπτύσσονται για τις περιπτώσεις της hard decision decoding για να μειωθεί η πολυπλοκότητα. Στο σχήμα 5.8 δεικνύεται η error performance για convolution codes με constrain length $K=7$ και code rates $\frac{1}{2}$ και $\frac{1}{3}$ και με sequential decoding (hard decisions) με rate $\frac{1}{2}$ και $\frac{1}{3}$ και constrain length $K=41$. Σημειώνετε ότι με $K=41$ επιτυγχάνεται ένα error rate της τάξης των 10^{-6} στο 2.5 db και 3db. Αντίστοιχα για Viterbi decoding με soft decision και rate $\frac{1}{2}$ και $\frac{1}{3}$, $K=7$ επιτυγχάνεται 5 και 4.4db για error rate της τάξης των 10^{-6} . Αυτά τα μικρά

constrain length επιτυγχάνουν ένα coding gain των 6db περίπου στα 10^{-6} , ενώ για μεγάλα constrain length το gain είναι της τάξης των 7.5-8db.



Σχήμα 5.8 Απόδοση για rate 1/2 & 1/3 Viterbi και Sequential decoding

Υπάρχουν δύο σημαντικά θέματα στην ανάπτυξη του Viterbi decoding :

1. Η επίδραση της μείωσης της διαδρομής μνήμης, η οποία είναι επιθυμητή, διότι εισάγει σταθερή καθυστέρηση στη αποκωδικοποίηση.
2. Ο βαθμός της αποκωδικοποίησης του εισαγόμενου σήματος στον Viterbi decoder.

ΚΕΦΑΛΑΙΟ 6

ΣΥΣΤΗΜΑ & ΛΟΓΙΣΜΙΚΟ ΠΡΟΣΟΜΟΙΩΣΗΣ

6.1 Γενικά

Η παρούσα διπλωματική στόχο είχε να δημιουργήσει ένα θεωρητικό μοντέλο συστήματος ασύρματης ψηφιακής επικοινωνίας, για το οποίο θα έπρεπε να υλοποιηθεί η προσομοίωση του με κατάλληλο λογισμικό (S/W), ώστε να εξεταστεί η απόδοσή του με βάσει τις διάφορες παραμέτρους του. Το εργαλείο που χρησιμοποιήθηκε για την ανάπτυξη του προγράμματος είναι το MATLAB.

Με βάσει τα ανωτέρω, στα επόμενα τμήματα του κεφαλαίου αυτού θα παρουσιαστούν και θα αναλυθούν :

(1) Η σχεδίαση του θεωρητικού μοντέλου ψηφιακής μετάδοσης με τα διάφορα υποσυστήματα του, και οι παραδοχές που έχουν γίνει. (§ 6.2)

(2) Η ανάπτυξη, του λογισμικού προσομοίωσης (S/W) του ανωτέρω μοντέλου – συστήματος ασύρματης ψηφιακής επικοινωνίας, θα αναφερθούν τα δεδομένα εισαγωγής, θα περιγραφούν οι χρησιμοποιούμενες συναρτήσεις, θα αναλυθούν αυτές που έχουν γραφεί για τις ανάγκες της διπλωματικής, και απλώς θα αναφερθούν αυτές που είναι ενσωματωμένες στο MATLAB και χρησιμοποιήθηκαν αυτούσιες. (§ 6.3)

(3) Θα δοθούν τα διάφορα δεδομένα εισαγωγής για τα οποία «έτρεξε» το πρόγραμμα και τα επιμέρους αποτελέσματα. (§ 6.4)

(4) Τέλος θα δοθούν οι σχετικές εξισώσεις βάσει των οποίων υπολογίστηκαν τα θεωρητικά μοντέλα καθώς και οι σχετικές γραφικές παραστάσεις, ώστε να είναι δυνατή η σύγκριση. (§ 6.5)

6.2 Σχεδίαση συστήματος

Η φιλοσοφία με την οποία σχεδιάστηκε το σύστημα ήταν να δίνεται η δυνατότητα να ελεγχθούν διαφορετικές αναπτύξεις του συστήματος-μοντέλου. Έτσι υπάρχουν οι εξής περιπτώσεις για τις οποίες θα τρέξουμε τον αλγόριθμο προσομοίωσης :

(1) Υπολογισμός του BER & SER για την περίπτωση όπου στο σύστημα δεν υπάρχει η τεχνική του channel estimation, δηλαδή έχουμε 'ideal', όπως

καθορίζεται μέσα στο πρόγραμμα . Σ' αυτή την περίπτωση έχουμε flat fading (μόνον το πλάτος του σήματος επηρεάζεται) . Για την τεχνική του channel estimation και των pilot symbols θα αναφερθούμε αναλυτικά στην επόμενη παράγραφο.

(2) Υπολογισμός των ίδιων μεγεθών , αλλά με channel estimation. Σ' αυτή τη περίπτωση χρησιμοποιούμε fading , όπου και η φάση επηρεάζεται.

Και στις δύο περιπτώσεις , η παλμοσειρά εισόδου (information bits) που στέλνεται, δίνεται η δυνατότητα είτε να είναι “uncoded” δηλαδή να μεταδίδεται χωρίς να έχει πραγματοποιηθεί κωδικοποίηση, και παράλληλα την άλλη με κωδικοποίηση.

Το μοντέλο που αναπτύχθηκε περιλαμβάνει τα ακόλουθα :

(1) Δημιουργία του information bit stream.

Εδώ αναφέρουμε ότι το μήκος αυτού του stream είναι καθορισμένο και ίσο με 200 bits για όλες τις περιπτώσεις που θα εξετάσουμε. Αυτό το stream είναι ουσιαστικά η καθαρή πληροφορία. Η δημιουργία αυτού του information bits γίνεται διαφορετικά κατά περίπτωση, ανάλογα με το εάν το σύστημα θα χρησιμοποιεί channel coding ή όχι και εάν θα χρησιμοποιεί την τεχνική του channel estimation . Γι' αυτή την τεχνική θα αναφερθούμε παρακάτω αναλυτικά.

(2) Διαδικασία της Διαμόρφωσης

Οι τύποι των διαμορφώσεων για τους οποίους θα εξεταστεί η απόδοση είναι τρεις και είναι :

- Η πιο απλή μορφή ψηφιακής PSK διαμόρφωσης η BPSK , για την οποία γνωρίζουμε ότι το bit error rate είναι ίσο με το symbol error.
- Την QPSK , που είναι πιο περίπλοκη μορφή διαμόρφωσης από την BPSK και τέλος
- Την 16 QAM για την οποία γίνεται η διαδικασία mapping και demapping.

Για τον τρόπο με τον οποίο υλοποιούνται οι κωδικοποιήσεις του bit stream σε σύμβολα , ανάλογα με την διαμόρφωση θα αναφέρουμε αναλυτικά παρακάτω.

(3) Για την περίπτωση , όπου έχουμε δημιουργία του encoded bit stream , με τη χρήση ενός convolutional encoder . Ο encoder που χρησιμοποιήθηκε είναι αυτός που διαθέτει το MATLAB ως έτοιμη συνάρτηση.

(4) Χρήση της τεχνικής interleaver , σύμφωνα με την οποία γίνεται μία «αναδιάταξη» των Bits , ώστε να βελτιωθεί η δυνατότητα για διόρθωση λαθών. Αυτό συμβαίνει διότι συνεχόμενα σύμβολα (Bits) τα οποία ενδεχομένως να ληφθούν λανθασμένα στο δέκτη , εξαιτίας του θορύβου (AWGN) στο κανάλι και του φαινομένου fading, με την τεχνική αυτή απομακρύνονται και δεν είναι πλέον συνεχόμενα. . Αυτό έχει σαν αποτέλεσμα , ο decoder , ο οποίος λειτουργεί πολύ καλά στις περιπτώσεις των random bit errors , αλλά όχι των συνεχόμενων (burst) λαθών.

(5) Χρήση της τεχνικής του punctured convolutional codes , ώστε να αυξηθεί το code rate του encoder.

(6) Χρήση square root raised cosine φίλτρου για το διαμορφωμένο σε βασική ζώνη σήμα (με χρήση κάποιου roll-off factor)

(7) Το μοντέλο που χρησιμοποιείται για την προσομοίωση του διαύλου επικοινωνίας είναι μοντέλο με AWGN και fading με κατανομή Rayleigh. Επισημαίνεται ότι δίνεται η δυνατότητα για δύο ειδών fading, αλλά σε κάθε περίπτωση για one-path . Το flat fading , όπου υφίσταται μεταβολή μόνον το πλάτος του σήματος , και η περίπτωση όπου έχουμε μεταβολή και στη φάση.

(8) Δημιουργία του δέκτη , ο οποίος εκτελεί την αντίστροφη διαδικασία δηλαδή :

(α) Matched filtering με το ίδιο raised cosine filter.

(β) Downsampling για να έχουμε τη στατιστική ανίχνευσης για κάθε σύμβολο στην έξοδο του προσαρμοσμένου φίλτρου . Θα πρέπει να επισημάνουμε την καθυστέρηση που υπάρχει στο 1^ο επιθυμητό δείγμα συμβόλου λόγω της ύπαρξης των δύο εν σειρά φίλτρων.

(γ) Αποδιαμόρφωση , η οποία γίνεται για τις αναφερόμενες τρεις διαμορφώσεις (BPSK, QPSK, 16QAM) .

(δ) Διαδικασία De-interleaving

(ε) Viterbi Decoding –όπου εφαρμόζεται η κατά Viterbi αποκωδικοποίηση.

(9) Χρήση της τεχνικής του channel estimation και των Pilot Symbols. Σύμφωνα μ' αυτήν τεχνική εξετάζεται η δυνατότητα με βάσει τον έλεγχο συγκεκριμένων symbols που στέλνονται από τον πομπό , και που είναι γνωστά στο δέκτη , να διορθώνεται η επίδραση του καναλιού στα bits πληροφορίας.

(10) Στο τελικό στάδιο , συγκρίνονται τα δύο streams , αυτό του αρχικού bit stream εκπομπής , με αυτό που έχουμε κατά τη διαδικασία της λήψης . Ανάλογα με τον αν συγκρίνονται Bits ή Symbols , έχουμε BER ή SER . Επίσης έχουν ληφθεί μετρήσεις και για FER (Frame Error Rate) .

6.3 Λογισμικό Προσομοίωσης

Το λογισμικό που αναπτύχθηκε, σε περιβάλλον matlab , επισυνάπτεται στο ΠΑΡΑΡΤΗΜΑ Ι και δίνονται όλες οι συναρτήσεις που καλούνται μέσα από το πρόγραμμα. Παρακάτω θα αναφερθούν αναλυτικά όλες οι παράμετροι που έχουν ορισθεί, τα δεδομένα εισαγωγής , τα εξαγωγή, καθώς επίσης και οι συναρτήσεις (function) που χρησιμοποιήθηκαν , είτε αυτές που το Matlab προσφέρει έτοιμες , είτε αυτές που αναπτύχθηκαν για τις ανάγκες του συγκεκριμένου προγράμματος.

Το πρόγραμμα περιλαμβάνει τις ακόλουθες ενότητες :

- (1) Preparation Part
- (2) Convolutional Encoding Initialization
- (3) Filter Initialization
- (4) Fading Initialization
- (5) Start Calculations
- (6) Fading Channel
- (7) Automatic Gain Control
- (8) Receiver with Matched Filter
- (9) Calculations of BER/SER/FER
- (10) Plotting figures

6.3.1 Παράμετροι Εισαγωγής

Στο πρόγραμμα ορίζονται οι μεταβλητές βάσει των οποίων υλοποιούνται οι διάφορες συναρτήσεις. Σύμφωνα με την δομή του προγράμματος , θα παρουσιαστούν παρακάτω οι διάφορες μεταβλητές για κάθε τμήμα του. Στο πρόγραμμα έχουν συγκεκριμένες τιμές , και μπορούν να αλλάξουν από τον χρήστη. Με βάση τον παραπάνω διαχωρισμό εννοιών έχουμε και τις αντίστοιχες παραμέτρους.

- (1) Preparation Part

➤ “**system**” - είναι η μεταβλητή που μπορεί να λάβει δύο τιμές την “**uncoded**” και την “**coded**” . Στη μία περίπτωση αυτή της uncoded τα

Information bits δεν εισέρχονται σε convolutional encoder , και η προσομοίωση ανταποκρίνεται στα σχήματα 7.1 και 7.3.

➤ **“estimation”** - Η μεταβλητή αυτή καθορίζει τον τρόπο με τον οποίο θα πραγματοποιηθεί η προσομοίωση. Οι τιμές που μπορεί να λάβει είναι δύο , η μία η **“ideal”** προσομοιάζει το σύστημα θεωρώντας ότι κατά τη διάρκεια της μετάδοσης έχουμε φαινόμενο flat fading, δηλαδή έχουμε επηρεασμό μόνον στο πλάτος του σήματος και όχι στη φάση. Με την τιμή αυτή , η προσομοίωση ανταποκρίνεται στα σχήματα 7.2 και 7.4. Η άλλη τιμή είναι η **“pilot”**, σύμφωνα με την οποία , γίνεται χρήση της συνάρτησης channel estimation , σύμφωνα με την οποία γίνεται χρήση των pilot symbols με σκοπό την διόρθωση του προβλήματος του fading , που σ’ αυτή τη περίπτωση δεν είναι flat , αλλά επηρεάζει και τη φάση του σήματος .

➤ **“ f_d ”** - είναι η Maximum Doppler Frequency [Hz] - Η συχνότητα αυτή είναι η μέγιστη ολίσθηση Doppler που παρατηρείται εξαιτίας του φαινομένου Doppler στη συχνότητα του φορέα. Η f_d δίνεται από τον τύπο v/λ , όπου v = η ταχύτητα του οχήματος και λ το μήκος κύματος της αντίστοιχης συχνότητας . Στο πρόγραμμα η $f_d = 30$ και θεωρώντας ότι έχουμε συχνότητα της τάξης των 1800 Mhz (κινητή τηλεφωνία) τότε η ταχύτητα του χρήστη είναι περίπου 30Km/h , σχετικά αργή.

➤ **“M”** - δηλώνει το αριθμό των συμβόλων και κατά επέκταση το είδος της διαμόρφωσης. Με την χρήση αυτής της μεταβλητής παρέχεται η ευκολία να τρέχουμε το πρόγραμμα για τις τρεις ψηφιακές διαμορφώσεις τις οποίες έχουμε επιλέξει να ελέγξουμε την BPSK, QPSK, 16QAM . Ετσι για $M=2$ έχουμε διαμόρφωση BPSK , για $M=4$ την QPSK και για $M=16$ την 16QAM .

➤ **“N_bits”** – δηλώνει τον αριθμό των information bits τα οποία χρησιμοποιούνται σε κάθε loop . Για την περίπτωση μας η μεταβλητή έχει την τιμή =200bits .

➤ **“codeRate”** – δηλώνει την code rate του convolution encoder .Μπορεί να πάρει διάφορες τιμές, στο πρόγραμμα παίρνει δύο τιμές τις $\frac{1}{2}$, και $\frac{3}{4}$.

➤ **“pattern”** – μοναδιαίος vector [1] . Χρησιμοποιείται στην διαδικασία του puncture.

➤ **“EsNodBmin”** , **“EsNodBmax”** , **“nloop”** – και οι τρεις είναι μεταβλητές οι οποίες ορίζουν , οι δύο πρώτες τις ελάχιστες και μέγιστες τιμές

του λόγου Ενέργεια Συμβόλου/Ισχύς Θορύβου για τις οποίες θα τρέξει το πρόγραμμα με βήμα ,(step) που καθορίζεται από το χρήστη. Οι τιμές θεωρούνται ότι δίνονται σε db.

➤ **“nloops”** – vector που περιέχει τον αριθμό των loops που θα τρέξει το πρόγραμμα για κάθε τιμή των **EsNodBmin** , **“EsNodBmax**.

➤ **“pilotlen”** – η μεταβλητή αυτή , η οποία δηλώνεται και ως global , και έτσι μπορεί να χρησιμοποιηθεί σ’ οποιαδήποτε function μέσα στο πρόγραμμα , χωρίς να χρειάζεται να δηλώνεται συνεχώς. Η τιμή της δείχνει τον αριθμό των pilot symbols που χρησιμοποιούμε , και είναι ίση με 12 (symbols) για όλες τις περιπτώσεις .

➤ **“br”** – η παράμετρος αυτή δηλώνει το bit rate των Information bits .Για όλες τις περιπτώσεις υλοποίησης του συστήματος η τιμή κρατείται σταθερή και ίση με 256Kbit/sec.

➤ **“fs”**- Είναι ο αριθμός των oversamples /symbol

(2) Convolutional Encoding Initialization

➤ **“constlen”** , **“codegen”** , **“tblen”**– είναι μεταβλητές που αφορούν την διαδικασία της κωδικοποίησης/αποκωδικοποίησης. Οι δύο η **constlen** η οποία είναι η μεταβλητή για το constraint length του encoder και είναι ίση με 7 , και η **codegen** είναι η μεταβλητή που δηλώνει το generator polynomial του encoder[171,133] και χρησιμοποιούνται για την δημιουργία του trellis . Επίσης το **tblen** είναι το traceback length για την διαδικασία του viterbi decoding (=32) .

➤ **”st2”**- Η μεταβλητή αυτή , χρησιμοποιείται για την διαδικασία του interleaving στη συνάρτηση **“randintrlv”** , και τίθεται ίση με έναν αριθμό (=4831) .

➤ **“k_rate”** - είναι η μεταβλητή που λαμβάνει υπόψη την επίδραση του coding

(3) Filter Initialiazation

➤ **“irfn”** - Είναι η μεταβλητή που καθορίζει τον αριθμό των συμβόλων μέσα στο φίλτρο.

➤ **“alfs”** – η μεταβλητή αυτή είναι ο Rolloff factor του φίλτρου.

❖ Fading initialization

- “**tstp**” – μεταβλητή που καθορίζει τον ελάχιστο χρόνο προσομοίωσης για το fading . Η τιμή αυτής της μεταβλητής εξαρτάται από το symbol rate και είναι ίση με $1/sr/4$
- “**n0**” - συμβολίζει τον αριθμό των κυμάτων που δημιουργούν fading . Γι’ αυτή την προσομοίωση έχουμε μόνον 1 –path Rayleigh fading
- “**itnd0**” – Αριθμός του fading counter που δείχνει την έναρξη του χρόνου για το γέννηση του φαινομένου fading σ’ ένα fading generator .
- “**itnd1**” – Αρχική τιμή του fading counter
- “**flat**” – η μεταβλητή καθορίζει εάν το fading θα είναι flat , δηλαδή εάν έχουμε επηρεασμό μόνον του πλάτους του σήματος οπότε flat=1 ή flat=0 και έχουμε επηρεασμό και της φάσης

6.3.2 Χρησιμοποιούμενες συναρτήσεις

Σ’ αυτήν την ενότητα θα περιγράψουμε αναλυτικά τις συναρτήσεις οι οποίες χρησιμοποιήθηκαν στο πρόγραμμα , και γράφτηκαν , ενώ θα γίνει απλή αναφορά σ’ αυτές που υπάρχουν έτοιμες στο matlab. Η περιγραφή θα γίνει όπως αυτές εμφανίζονται στα διάφορα τμήματα του προγράμματος

(α) Convolutional Encoding Initialization

Η μόνη συνάρτηση που καλείται εδώ είναι μία συνάρτηση ενσωματωμένη στο matlab και είναι η **poly2trellis(constlen, codegen)**. Η συνάρτηση αυτή έχει 2 μεταβλητές οι οποίες δηλώνονται, την constlen και την codegen . Μετατρέπει την πολυωνυμική έκφραση του convolutional code σε trellis διάγραμμα.

(β) Filter Initialization

Σ’ αυτό το τμήμα υπάρχει μία συνάρτηση που καλείται και ονομάζεται **hrollfcoef(irfn,fs,sr,alfs,1)** . Η συνάρτηση αυτή επιστρέφει (vectors) τους συντελεστές των Nyquist φίλτρων [hTransmitFilter] για το filter transmit και receive[hReceiveFilter] αντίστοιχα. Οι παράμετροι που δέχεται είναι :

- (i) Η “irfn” και η “alfs” οι οποίες όπως έχει ήδη αναφερθεί καθορίζουν η μεν πρώτη τον αριθμό των συμβόλων που χρησιμοποιούνται για filtering , ενώ η δεύτερη είναι ο συντελεστής rolloff του φίλτρου.

(u) Η “fs”, “sr”, που δηλώνουν αντίστοιχα τον αριθμό των oversamples / symbol , ενώ η δεύτερη δηλώνει το symbol rate .

(iii) Η τελευταία μεταβλητή δηλώνει εάν πρόκειται για το transmit filter (1) ή το receive filter (0) .

(γ) Start Calculations

Σ’ αυτό το τμήμα του προγράμματος καλούνται οι ακόλουθες συναρτήσεις :

(1) “**randsrc**” – η συνάρτηση αυτή , είναι συνάρτηση του matlab , και δημιουργεί μία τυχαία σειρά, ένα vector, από 1’ και ‘0’. Οι μεταβλητές που δέχεται είναι τρεις συνολικά και είναι :

(i) “**N_bits**” που είναι ο αριθμός των Bits ,

(ii) η δεύτερη παράμετρος είναι ο αριθμός των στηλών του vector που στη προκειμένη περίπτωση είναι μία στήλη (=1) και

(iii) η τελευταία παράμετρος είναι οι ακέραιοι που υπάρχουν στον vector , στη περίπτωση μας είναι 0 και 1 .

(2) “**convenc**” – συνάρτηση η οποία υπάρχει ενσωματωμένη στο matlab και είναι υλοποιεί ένα convolutional encoder . Δέχεται δύο μεταβλητές :

(i) τα information bits και

(ii) τα αποτελέσματα της μετατροπής σε trellis και παράγει coded bits.

(3) “**puncture**” – η συνάρτηση αυτή η οποία έχει υλοποιηθεί για τις ανάγκες του προγράμματος εκτελεί την διαδικασία του puncture . Όπως έχουμε αναφέρει η διαδικασία αυτή αφαιρεί κάποια bits για να αυξήσει το code rate. Οι μεταβλητές που δέχεται είναι κατ’ αρχήν τα κωδικοποιημένα bits (coded bits) και μία μεταβλητή που την ονομάζουμε pattern βάση της οποίας γίνεται η αφαίρεση των Bits .

(4) “**randintrlv**” – η συνάρτηση αυτή υπάρχει ενσωματωμένη μέσα στο matlab . Ως εκ τούτου χρησιμοποιείται ως έχει με δύο μεταβλητές. Ο τύπος αυτός είναι random interleaver και επιλέγει ένα πίνακα σύμφωνα με τον οποίο ανακατεύει τα bits . Η πρώτη μεταβλητή που δέχεται αυτή η συνάρτηση είναι τα bits που πρόκειται να «ανακατέψει» και η δεύτερη μεταβλητή είναι η αρχική κατάσταση εισόδου που δίνεται από το χρήστη. Όπως γνωρίζουμε ένας random interleaver ανακατεύει τα bits πολλαπλών codewords προτού δύο

γειτονικά codewords χαλάσουν από τρία συνεχόμενα λάθη. Έτσι με τον interleaver βελτιώνεται το error rate σ' ένα τηλεπικοινωνιακό σύστημα γιατί πρακτικά κατανέμονται τα λάθη μεταξύ διαφορετικών codewords .

(5) “**qammod**” - πρόκειται για την συνάρτηση που υλοποιεί τη διαμόρφωση για την περίπτωση της BPSK και της 16 QAM και υπάρχει μέσα στο matlab . Δύο είναι οι μεταβλητές που πρέπει να δοθούν , τα bits , και τον αριθμό των συμβόλων , π.χ για BPSK $M=2$, και για 16QAM $M=16$.

(6) “**QPSKEncoder**” – η συνάρτηση αυτή , εκτός matlab , μετατρέπει τα bits σε σύμβολα , δημιουργώντας τα I και Q και υλοποιώντας έτσι την διαμόρφωση QPSK .

(7) “**bi2de**” – μετατρέπει ένα άνυσμα δυαδικών (0,1) σε δεκαδικό μη αρνητικό ακέραιο και είναι συνάρτηση του matlab.

(8) “**TransmitFilter**” - Η συνάρτηση αυτή δεν είναι ενσωματωμένη στο matlab. Με τη συνάρτηση αυτή ουσιαστικά υλοποιούμε τον πομπό για την μετάδοση του σήματος. Οι μεταβλητές που δέχεται σαν είσοδο είναι :

(i) Τα ich και qch , που είναι ουσιαστικά τα σύμβολα προς μετάδοση .

(ii) Την $hTransmitFilter$, η οποία περιέχει τους συντελεστές του φίλτρου του πομπού.

(iii) τον αριθμό των oversample/symbol που για το συγκεκριμένο πρόγραμμα είναι σταθερό και ίσο με 8. Οι έξοδοι που δίνει αυτή η συνάρτηση είναι οι κυματομορφές μετάδοσης (I & Q) . Κατά την εκτέλεση αυτού του προγράμματος μετατρέπουμε πρώτα το εισερχόμενο symbol stream σε κυματομορφή. Σ' αυτό το σημείο καλείται μία άλλη συνάρτηση , που ονομάζεται “**SymbolToWaveform**” και ουσιαστικά είναι αυτή που μετατρέπει τα σύμβολα της I & Q σε αντίστοιχες κυματομορφές. Κατόπιν αυτές οι κυματομορφές φιλτράρονται από το φίλτρο ώστε να έχουμε το σήμα ζώνης προς μετάδοση. Το φιλτράρισμα αυτό γίνεται με την εντολή conv (convolution-συνέλιξη) της κάθε κυματομορφής με τους συντελεστές του φίλτρου. Τα αποτελέσματα που δίνει η συνάρτηση **TransmitFilter** είναι τα πλάτη του σήματος που εκπέμπονται .

(γ) Fading Channel

Σ' αυτό το τμήμα υπάρχει μόνον μια συνάρτηση η οποία καλείται “**fade**” . Η συνάρτηση αυτή υλοποιεί την διαδικασία του fading στο

λαμβανόμενο σήμα είτε πρόκειται για flat fading είτε normal, αυτό επιλέγεται από μία μεταβλητή την flat η οποία εάν είναι =1, τότε έχουμε μεταβολή μόνον στο πλάτος, εάν flat=0 τότε έχουμε μεταβολή και στη φάση. Θα πρέπει να αναφερθεί ότι η συνάρτηση υλοποιεί το fading θεωρώντας ότι ακολουθείται η Rayleigh κατανομή για one-path wave. Το πρόγραμμα ουσιαστικά υλοποιεί την ακόλουθη έκφραση η οποία είναι η συνάρτηση που παράγει το Rayleigh Fading Envelope με $N=n$ και $N=n1$.

$$r(t) = x(t) + j \bullet y(t) = \left[\sqrt{\frac{2}{N_1 + 1}} \sum_{n=1}^{N_1} \sin\left(\frac{\pi n}{N_1}\right) \cos\{2f_d \cos\left(\frac{2\pi n}{N_1}\right)t\} + \frac{1}{\sqrt{N_1 + 1}} \cos(2f_d t) \right] + j \left[\sqrt{\frac{2}{N_1 + 1}} \sum_{n=1}^{N_1} \sin\left(\frac{\pi n}{N_1}\right) \cos\{2f_d \cos\left(\frac{2\pi n}{N_1}\right)t\} \right]$$

N_1 low frequency oscillators με συχνότητες ίσες με Doppler shifts $f_d * \cos(2\pi n / N)$, $n=1,2,\dots,N_1$ συν μία συχνότητα f_d , χρησιμοποιούνται για να δημιουργήσουν σήματα frequency shifted από την συχνότητα του carrier. Τα πλάτη όλων των συνιστωσών

είναι ίσα με $\sqrt{\frac{2}{N_1 + 1}}$ εκτός από αυτό στην συχνότητα f_d , που είναι $\frac{1}{\sqrt{N_1 + 1}}$. Οι

συντελεστές αυτοί επιλέγονται έτσι ώστε η μέση ισχύς της διάλειψης να είναι ίση με 1. Οι φάσεις των σημάτων επιλέγονται έτσι ώστε η κατανομή της φάσης να είναι ομοιόμορφη, $1/2\pi$. Γι αυτό έχουμε τις φάσεις ως $2\pi n/N$, $n=1,2,\dots,N_1$.

Οι παράμετροι που δέχεται αυτή η συνάρτηση είναι :

- (i) Είναι τα πλάτη του σήματος στο I και Q
- (ii) Το μήκος του vector που περιέχει τα πλάτη του I
- (iii) Το time resolution για το fading simulation
- (iv) Το f_d , που όπως έχουμε πει η max Doppler Frequency
- (v) Το flat που δηλώνει το είδος του fading
- (vi) Οι παράμετροι για τον fading counter. Κάθε δείγμα του fading simulator αντιστοιχεί σε πραγματικό χρόνο $t_{stp}=1/sr/numSamplesPerSymbol$ (αριθμός δειγμάτων που χρησιμοποιούνται για να καθορίσουν ένα σύμβολο). Η μεταβλητή counter παίρνει μία αρχική τιμή χρόνου $ic0$ (αντιστοιχεί σε αριθμό δειγμάτων) η οποία πρέπει να είναι διαφορετική κάθε φορά που θέλουμε να

δημιουργήσουμε μία νέα διεργασία Rayleigh Fading η οποί να δίνει ανεξάρτητα δείγματα. Το time resolution (tstp) είναι ο χρόνος μεταξύ δύο δειγμάτων.

(δ) Automatic Gain Control (AGC)

Η συνάρτηση αυτή εξισορροπεί το κέρδος στο πλάτος των στοιχείων In-phase και Q από κοινού ή ανεξάρτητα . Αυτό γίνεται υπολογίζοντας τα πλάτη I και O.

(ε) Receiver with Matched Filter

Έχουμε τις ακόλουθες συναρτήσεις οι οποίες εκτελούνται σ' αυτό το τμήμα :

(1) “ **ReceiveFilter**” – Η συνάρτηση αυτή είναι η αντίστοιχη της Transmit Filter. Πρώτα γίνεται το matched filtering χρησιμοποιώντας την conv . Θεωρούμαι ότι τα φίλτρα transmit , receive έχουν το ίδιο μήκος. Μετατρέπουμε σε symbol stream το I και το Q λαμβάνοντας υπόψη τη καθυστέρηση (Delay) λόγω των κρουστικών αποκρίσεων των φίλτρων. Έτσι το πρώτο χρήσιμο δείγμα που προέρχεται από σύμβολο εκπομπής εμφανίζεται μετά το φίλτρο λήψης στο δείγμα με αριθμό Delay+1. Η συνάρτηση που χρησιμοποιείται εντός της ReceiveFilter είναι η “ **WaveformToSymbol** “ η οποία μετατρέπει ένα waveform σε symbol stream παίρνοντας 1 στοιχείο κάθε 4 (=numSamplesPerSymbol) στοιχεία. Έχει προβλεφθεί (μεταβλητή SymbolRange) ώστε το 1^ο δείγμα να είναι στην πραγματικότητα το Delay+1. Στη συνέχεια παίρνουμε 1 δείγμα κάθε 4 δείγματα από τον vector FilterOutput .

(2) “ **chan_est**” – με την υλοποίηση αυτής της συνάρτησης προσπαθούμε να εξαλείψουμε την επίδραση του καναλιού ,για την περίπτωση που έχουμε fading που επηρεάζει και το πλάτος αλλά και τη φάση του σήματος. Χρησιμοποιούνται τα pilot symbols , τα οποία έχουμε ορίσει να είναι 12 , και είναι από τα ίδια τα μεταδιδόμενα σύμβολα από τον πομπό. Δηλαδή στη περίπτωση που κάνουμε χρήση της τεχνικής του channel estimation και των pilot symbols , τα 12 πρώτα bits απ'αυτά που είχαμε εκπέψει είναι και τα pilot. Ανάλογα με τη διαμόρφωση έχουμε :

Για την BPSK έχουμε ότι

$$i_ce = \sum_{i=1}^{12} rx_ch_i(i) \quad (1)$$

$$q_{ce} = \sum_{i=1}^{12} rx_{ch_q}(i) \quad (2)$$

Για την περίπτωση της QPSK έχουμε ότι

$$i_{ce} = \frac{1}{2} \left[\sum_{i=1}^{12} rx_{ch_i}(i) + rx_{ch_q}(i) \right] \quad (3)$$

$$q_{ce} = \frac{1}{2} \sum_{i=1}^{12} rx_{ch_i}(i) + rx_{ch_q}(i) \quad (4)$$

Και για 16 QAM έχουμε

$$i_{ce} = \frac{1}{6} \sum_{i=1}^{12} rx_{ch_i}(i) + rx_{ch_q}(i) \quad (5)$$

$$q_{ce} = \frac{1}{6} \sum_{i=1}^{12} rx_{ch_i}(i) + rx_{ch_q}(i) \quad (6)$$

Για τον υπολογισμό του καναλιού παίρνουμε την μέση τιμή των pilot symbols η οποία είναι για όλες τις παραπάνω περιπτώσεις :

$$i_{ce} = \frac{i_{ce}}{12} \quad (7)$$

$$q_{ce} = \frac{q_{ce}}{12} \quad (8)$$

Γνωρίζουμε ότι τα λαμβανόμενα σύμβολα είναι $RxData = rx_{ch_i} + tx_{ch_q}$

Και τα pilot symbols είναι $IQ_symbols = (i_{ce} + j * q_{ce})$

Για να διορθώσουμε την επίδραση του καναλιού και χρησιμοποιώντας τα pilot symbols έχουμε ότι τα λαμβανόμενα σύμβολα θα είναι :

$$RX_symbols = RxData * \frac{conj(IQ_ch)}{\sqrt{i_{ce}^2 + q_{ce}^2}}$$

Άρα τα λαμβανόμενα σύμβολα είναι τα παραπάνω , διορθωμένα κατά τον

παράγοντα $\frac{conj(IQ_ch)}{\sqrt{i_{ce}^2 + q_{ce}^2}}$ ο οποίος αντιστοιχεί στις παραμέτρους του καναλιού

και έχει υπολογιστεί με την βοήθεια των pilot symbols .

(3) “**qamdemod**” - πρόκειται για την συνάρτηση που υλοποιεί τη αποδιαμόρφωση, δηλαδή την μετατροπή από σύμβολα σε bit . Εδώ την χρησιμοποιούμε για την περίπτωση της BPSK και της 16 QAM και υπάρχει μέσα στο matlab . Δύο είναι οι μεταβλητές που πρέπει να δοθούν , τα λαμβανόμενα

σύμβολα , και ο αριθμός των συμβόλων , π.χ για BPSK $M=2$, και για 16QAM $M=16$.

(4) “**QPSKDecoder**” – Μετατρέπει τα λαμβανόμενα σύμβολα σε bits . Για κάθε σύμβολο που είναι >0 , τότε είναι ίσο με ‘1’ , εάν είναι <0 τότε είναι ίσο ‘0’ . Ως παράμετροι εισόδου έχουμε μόνον τα λαμβανόμενα σύμβολα `rx_ich`, `rx_qch`.

(5) “**ParallelToSerial**” – Η συνάρτηση αυτή μετατρέπει τα παράλληλα bits σε σειριακά , ώστε να έχουμε την αρχική παλμοσειρά εισόδου για να μπορούμε να κάνουμε την σύγκριση.

(6) “**de2bi**” - εδώ μετατρέπονται οι ακέραιοι σε bits .

(ζ) Calculations of BER/SER/FER

Σ’ αυτό το τμήμα γίνονται οι υπολογισμοί οι οποίοι για :

- (1) BER- Bit Error Rate – ο υπολογισμός για τον ρυθμό σφαλμάτων στο επίπεδο Bit γίνεται συγκρίνοντας τις δύο παλμοσειρές Bit , αυτής της δημιουργούμενης για μετάδοση και αυτής που έχουμε μετά την διαδικασία της λήψης.
- (2) SER- Symbol Error Rate – εδώ υπολογίζονται τα λάθη στα σύμβολα .
- (3) FER –Frame Error Rate - Εδώ υπολογίζονται τα λάθη σε επίπεδο frame . Ως frame ορίζεται το σύνολο των bits ($=200$) που στέλνονται σε κάθε loop . Κάθε ένα λάθος στο frame, δηλαδή στα 200 bits , λογίζεται ως ένα λάθος και το frame θεωρείται λάθος.

Στη περίπτωση που το σύστημα χρησιμοποιεί coding , τότε πριν τον υπολογισμό των παραπάνω θα πρέπει να προηγηθούν οι διαδικασίες της :

- Μετατροπής των παράλληλων λαμβανόμενων συμβόλων I & Q σε σειρά bits .
- Αναστροφή της διαδικασίας του Interleaving χρησιμοποιώντας την συνάρτηση `deintrlv`
- Προετοιμασίας για Soft –decision decoding όπου χρησιμοποιείται η συνάρτηση `quantiz`. Η συνάρτηση αυτή που υπάρχει έτοιμη στο matlab , αντιστοιχίζει τα λαμβανόμενα data στους κατάλληλους decision-value integers μεταξύ 0-7 . Το δεύτερο argument στην συνάρτηση

αυτή είναι ένας partition vector που καθορίζει ποιες data values αντιστοιχούν στα 0,1,2 κλπ.

- Viterbi Decoding , συνάρτηση έτοιμη στο matlab , με παραμέτρους , την έξοδο από την προηγούμενη διαδικασία που είναι οι τιμές soft decoding , τις παραμέτρους από το trellis , τη τιμή του tblen (traceback length) , τη παράμετρο 'term' , τη δήλωση για 'soft' decision .

Τέλος σ' αυτό το τμήμα υπολογίζονται , μέσα στα κατάλληλα loops που γίνονται οι τιμές των διαφόρων σφαλμάτων BER, SER , FER σε συνάρτηση με τις τιμές του E_bN_0 που δίνονται στο αρχικό τμήμα του προγράμματος .

(η) Plotting figures

Στο τμήμα αυτό που είναι και το τελευταίο , γράφονται οι εντολές για την απεικόνιση των γραφικών παραστάσεων και τον σχεδιασμό των σημείων των τιμών του E_bN_0 συναρτήσει των υπολογιζόμενων λαθών , σε Bit , symbol ή σε frame.

6.4 Παράμετροι εισαγωγής & Γραφικές Παραστάσεις

Το πρόγραμμα έτρεξε για τα ακόλουθα δεδομένα εισόδου :

$f_d = 30$

$M = 2, 4, 16$

$N_bits = 200$

codeRate=1/2

step = 2

EsNodBmin=0

EsNodBmax=30

nloop =[1000 2000 3000 4000 5000 6000 7000 8000 10000 15000 20000
25000 30000 40000 50000 60000]

pilotlen=12

br=256000

nd=100

constlen=7

tblen=32

codegen=[171,133]

st2=4831

irfn=8

alfs=0.5

flat=0 ή 1

6.5 Θεωρητικές Εξισώσεις

Τα αποτελέσματα της προσομοίωσης συγκρίθηκαν με αυτά των θεωρητικών εξισώσεων που είναι , τα οποία θεωρούνται ότι αντιπροσωπεύουν τις συνθήκες σε ένα AWGN & one-path Rayleigh fading channel. Θα παραθέσουμε τις εξισώσεις που αφορούν την περίπτωση όπου δεν έχουμε κωδικοποίηση στο σήμα , και την περίπτωση που χρησιμοποιούμε ένα συνελκτικό κώδικα , συγκεκριμένων παραμέτρων και η αποκωδικοποίηση γίνεται με Viterbi .

(1) UNCODED

➤ BPSK

$$(α) BER_{AWGN} = \frac{1}{2} \operatorname{erfc}(E_b / N_0)$$

$$(β) BER_{FADING} = \frac{1}{2} \left[1 - \frac{1}{\sqrt{1 + \frac{1}{E_b / N_0}}} \right]$$

➤ QPSK

$$(α) BER_{AWGN} = \frac{1}{2} \operatorname{erfc}\left(\sqrt{\frac{E_b}{N_0}}\right)$$

$$(β) BER_{FADING} = \frac{1}{2} \left[1 - \frac{1}{\sqrt{1 + \frac{1}{E_b / N_0}}} \right]$$

$$(γ) SER_{AWGN} = 2Q\left(\sqrt{\frac{2E_b}{N_0}}\right) \left[1 - \frac{1}{2}Q\left(\sqrt{\frac{2E_b}{N_0}}\right) \right]$$

➤ 16QAM

$$(α) BER_{AWGN} = \frac{3}{8} \operatorname{erfc} \left[\sqrt{\frac{2 E_b}{5 N_0}} \right] - \frac{9}{64} \operatorname{erfc}^2 \left(\sqrt{\frac{2 E_b}{5 N_0}} \right)$$

$$(β) BER_{FADING} = \frac{3}{8} \left[1 - \frac{1}{\sqrt{1 + \frac{5}{2E_b / N_0}}} \right]$$

$$(γ) SER_{AWGN} = (1 - P_{\sqrt{M}})^2 \text{ και}$$

$$P_{\sqrt{M}} = 2 \left(1 - \frac{1}{\sqrt{M}} \right) Q \left(\sqrt{\frac{3}{M-1} \frac{E}{N_0}} \right)$$

Οι θεωρητικοί υπολογισμοί για το Symbol Error rate και οι καμπύλες των θεωρητικών τιμών έχουν εξαχθεί με χρήση της MGF, υλοποιώντας την με τα σχετικά προγράμματα, τα οποία και επισυνάπτονται

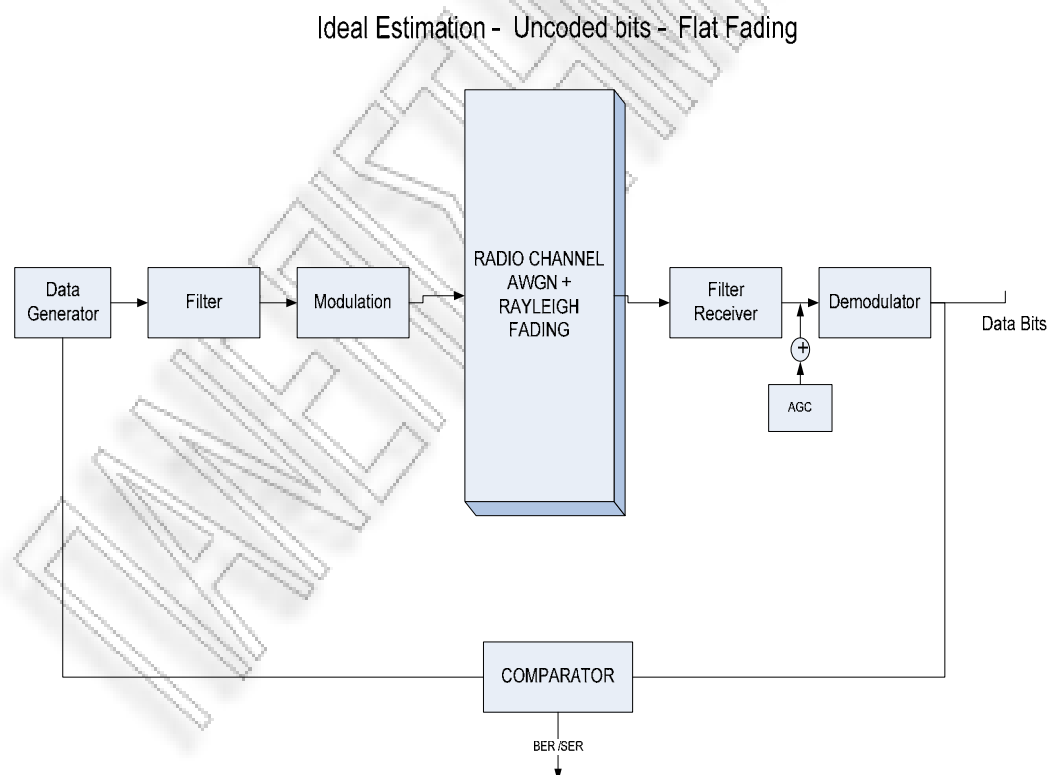
ΚΕΦΑΛΑΙΟ 7

ΑΠΟΤΕΛΕΣΜΑΤΑ ΠΡΟΣΟΜΟΙΩΣΗΣ

Στο παρόν κεφάλαιο παρουσιάζονται τα αποτελέσματα της προσομοίωσης με την μορφή γραφικών παραστάσεων για κάθε περίπτωση συστήματος που παρουσιάζεται. Συνολικά υλοποιήθηκαν τέσσερις περιπτώσεις και προσομοιώθηκαν αντίστοιχα με το λογισμικό. Για όλες τις περιπτώσεις έχουμε διαδικασία για κάθε ένα τύπο διαμόρφωσης ξεχωριστά (BPSK, QPSK, 16QAM) Συνολικά έχουμε () γραφικές παραστάσεις. Αναλυτικά :

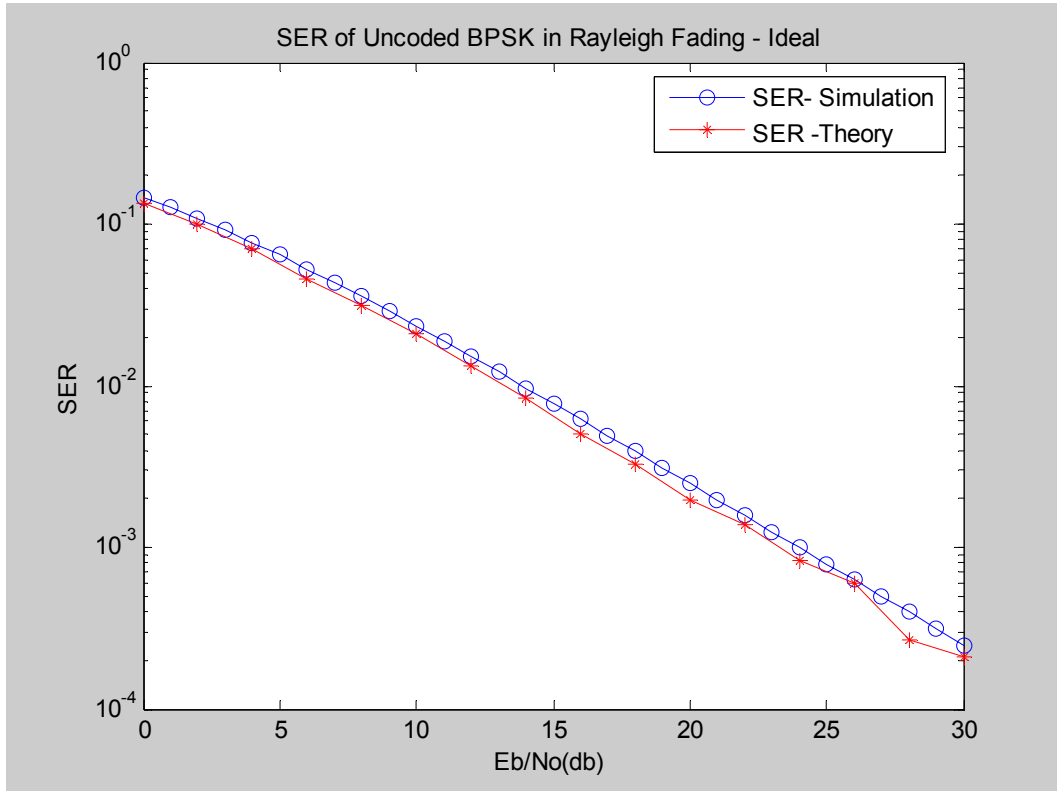
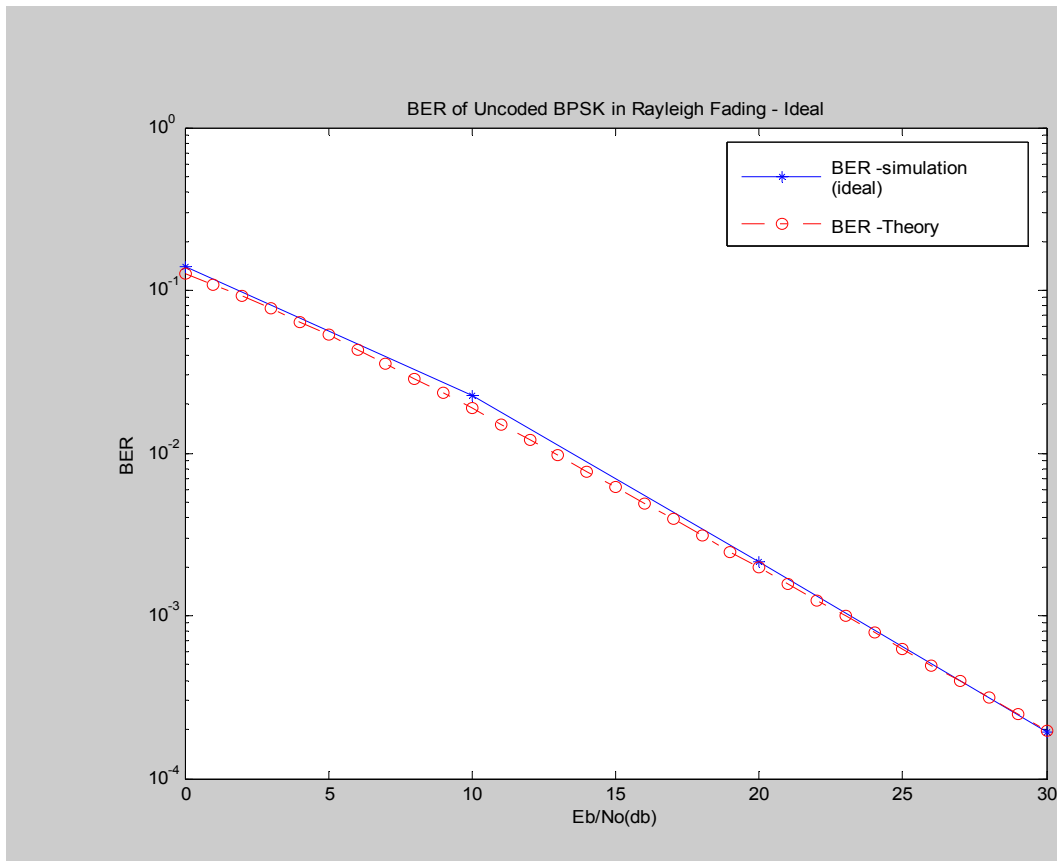
(1) Προσομοίωση **Uncoded Data** σε Rayleigh **Flat** Fading AWGN κανάλι .

Το σχήμα 7.1 απεικονίζει το σύστημα εκπομπής –λήψης που υλοποιείται με το λογισμικό. Τα δεδομένα εισόδου είναι ανάλογα, και ο έλεγχος των σφαλμάτων σε bit και symbol έδωσε τις ακόλουθες γραφικές παραστάσεις.

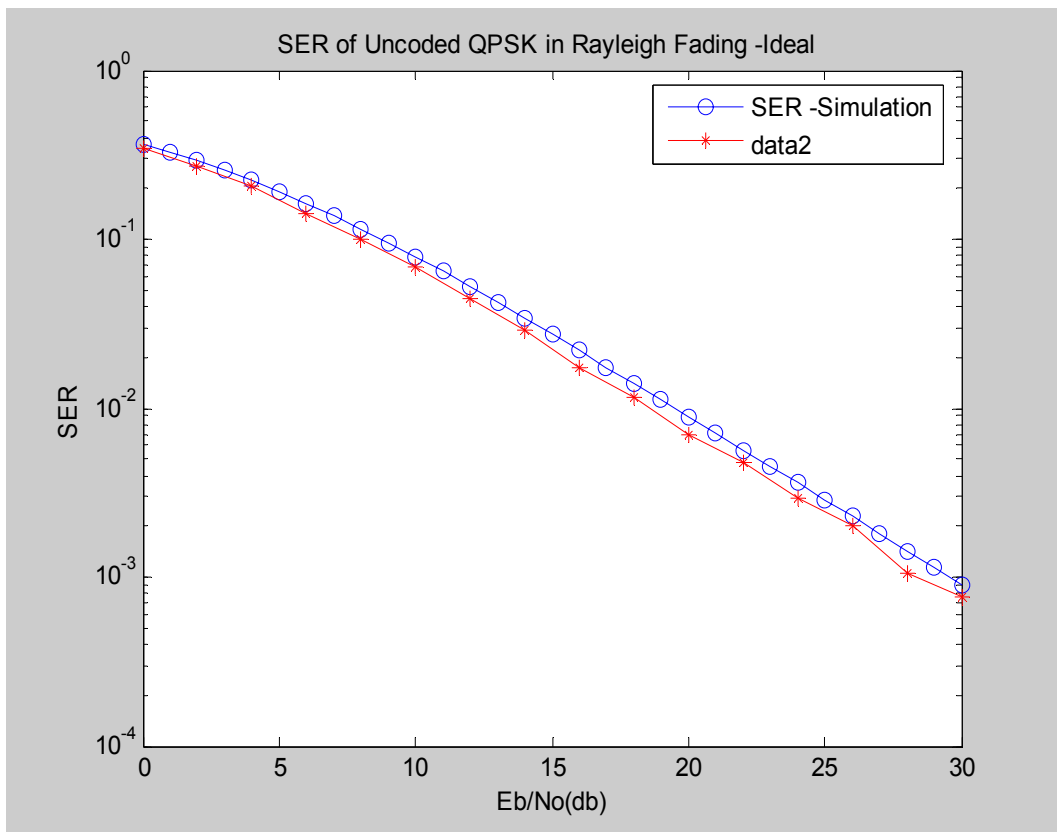
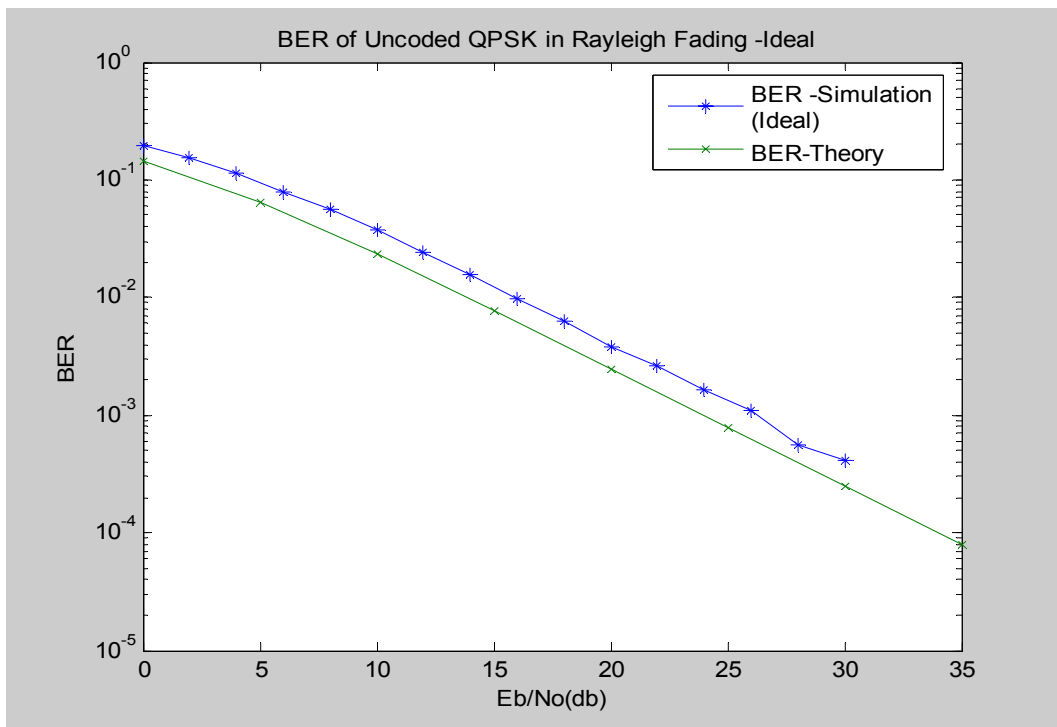


Σχήμα 7.1

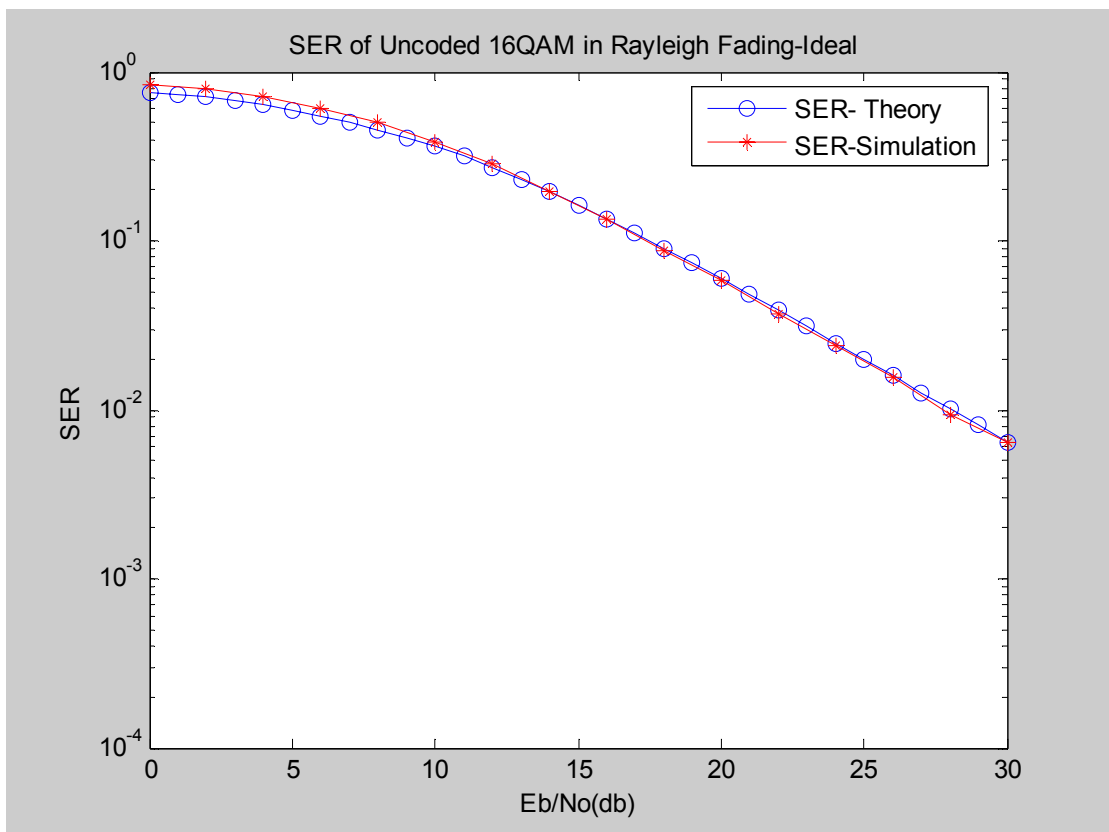
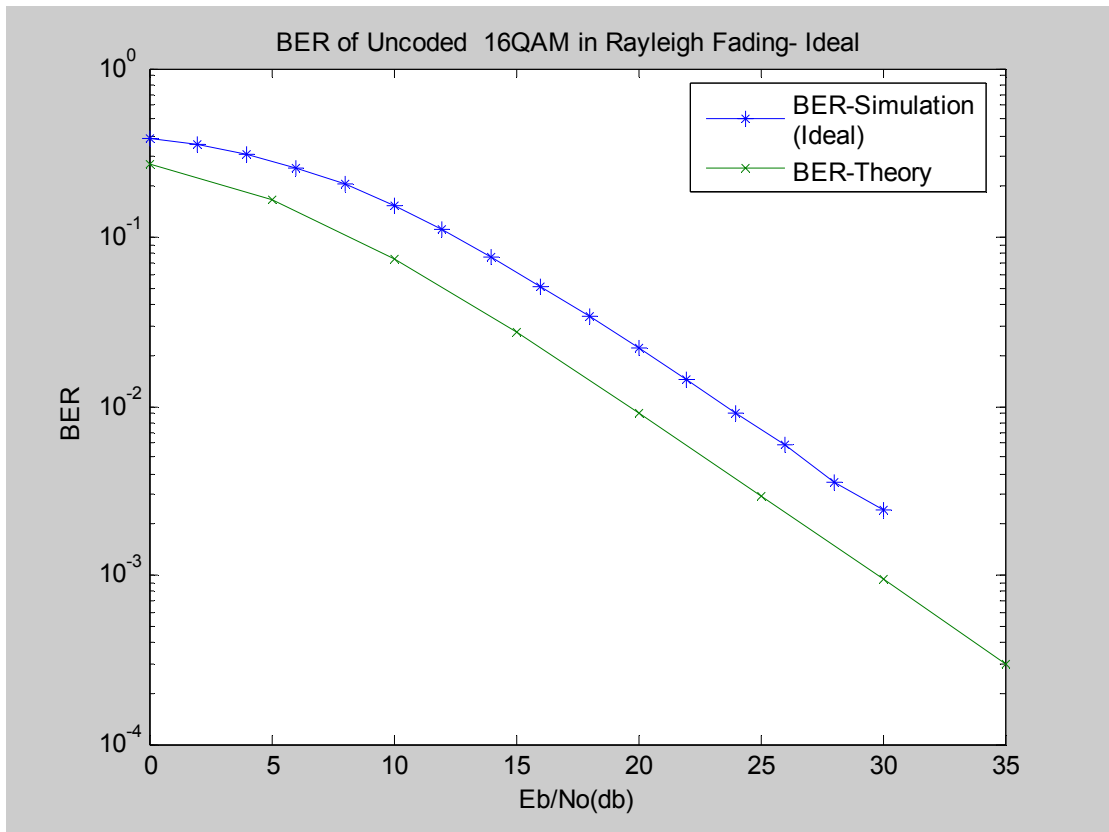
(1) Διαμόρφωση BPSK



(2) Διαμόρφωση QPSK

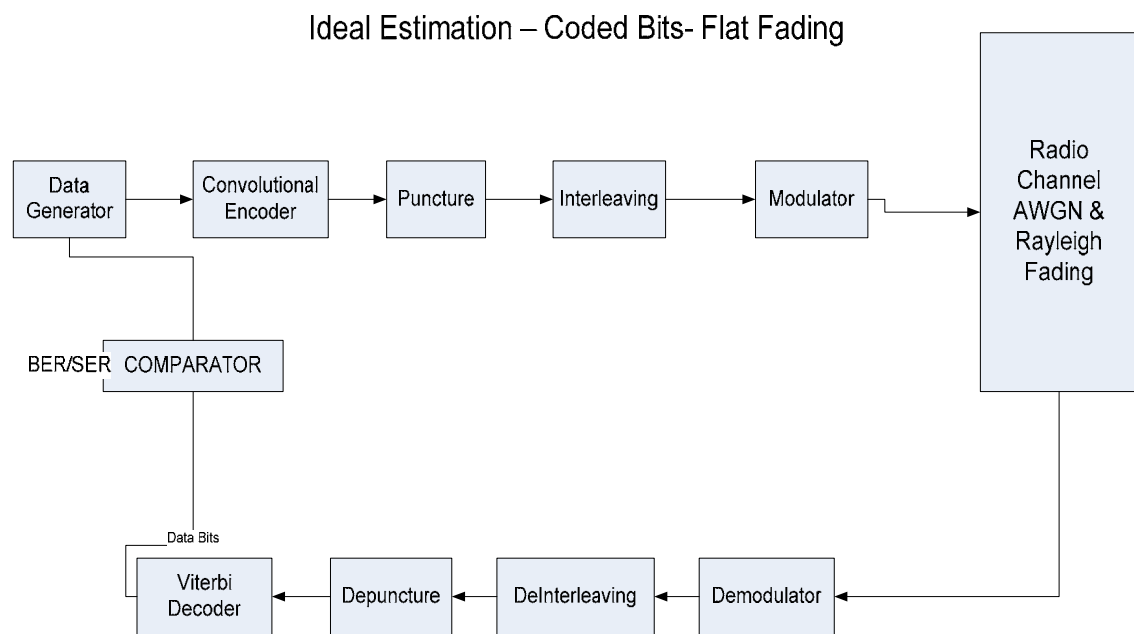


(3) Διαμόρφωση 16QAM



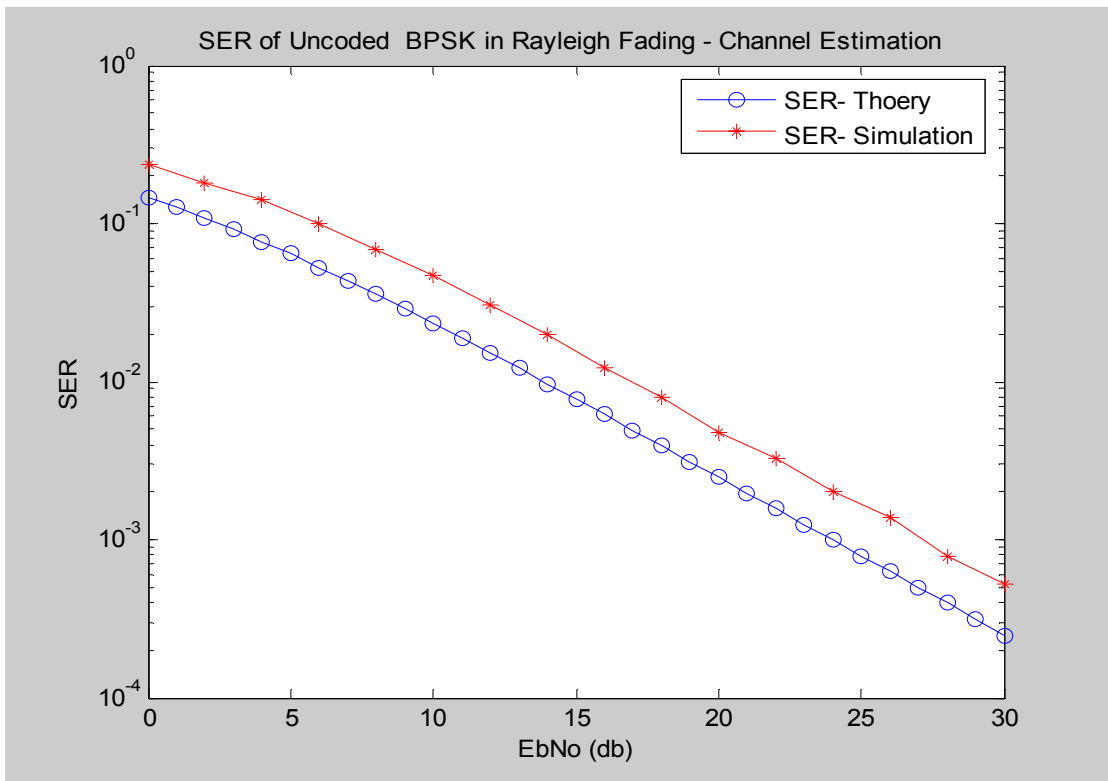
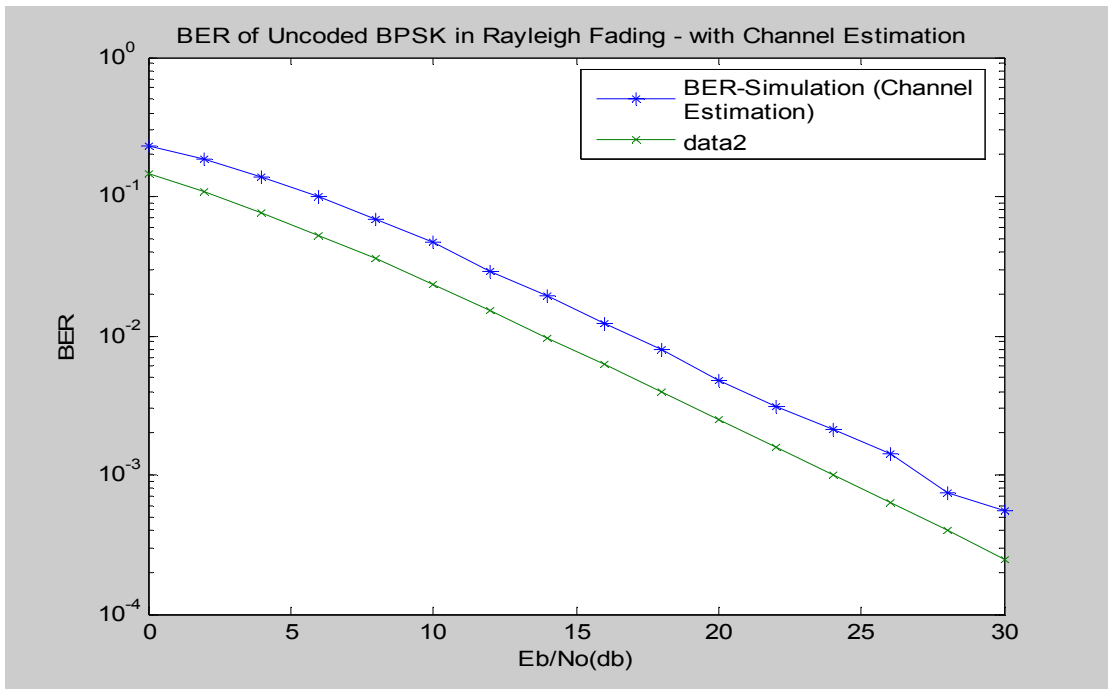
(2) Προσομοίωση **Uncoded Data** σε Rayleigh Fading **Normal** AWGN κανάλι (**Channel Estimation**)

Στο σχήμα 7.2 δεικνύεται το σύστημα το οποίο προσομοιώνεται στην περίπτωση αυτή . Η διαφορά με την προηγούμενη είναι ότι εδώ εφαρμόζεται η τεχνική του channel estimation με την οποία φαίνεται να μειώνεται η επίδραση του φαινόμενου fading (normal) .

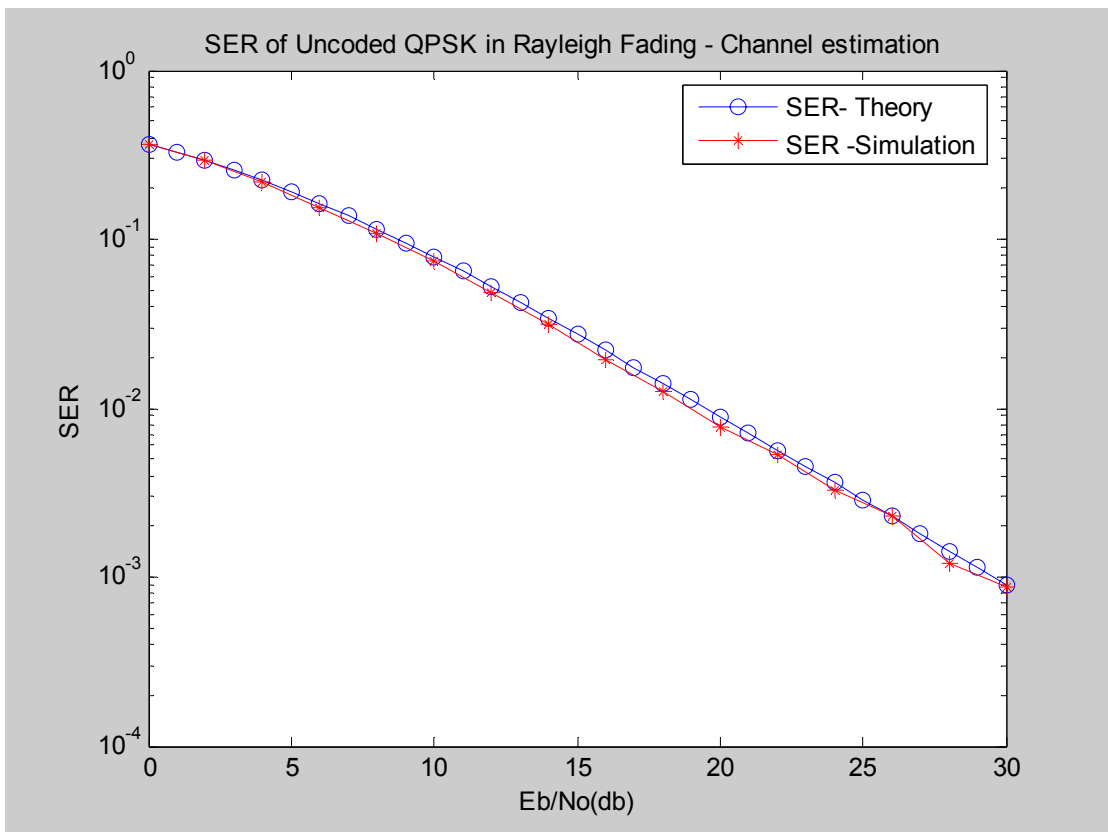
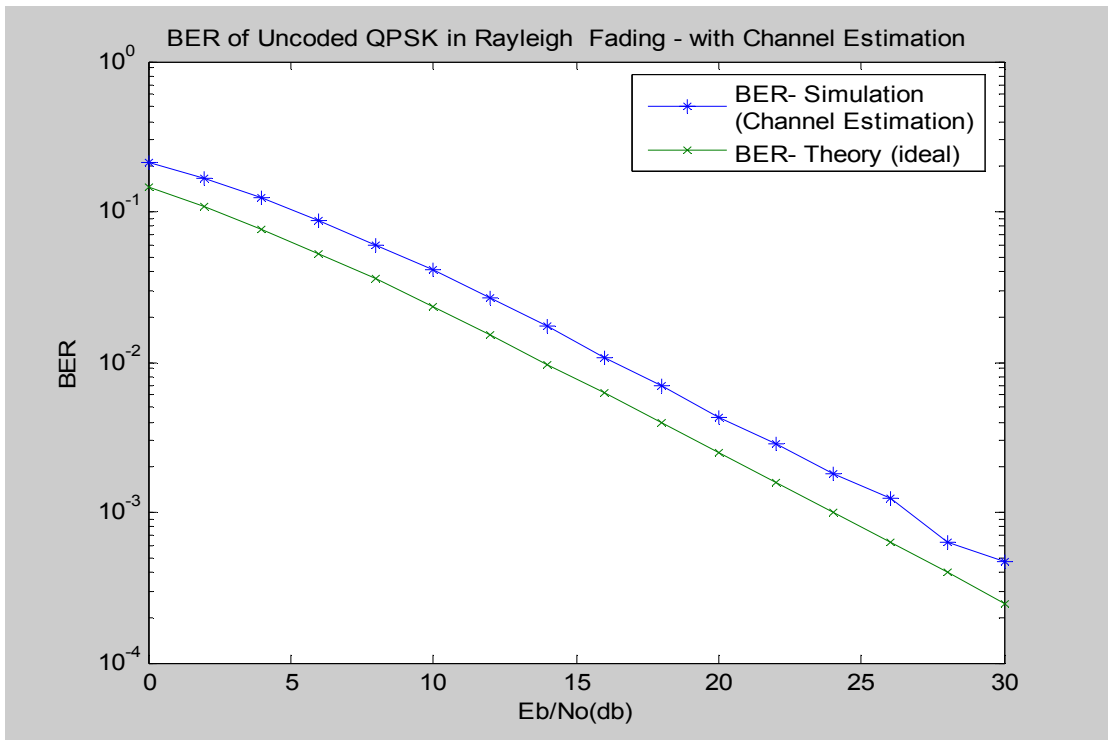


Σχήμα 7.2

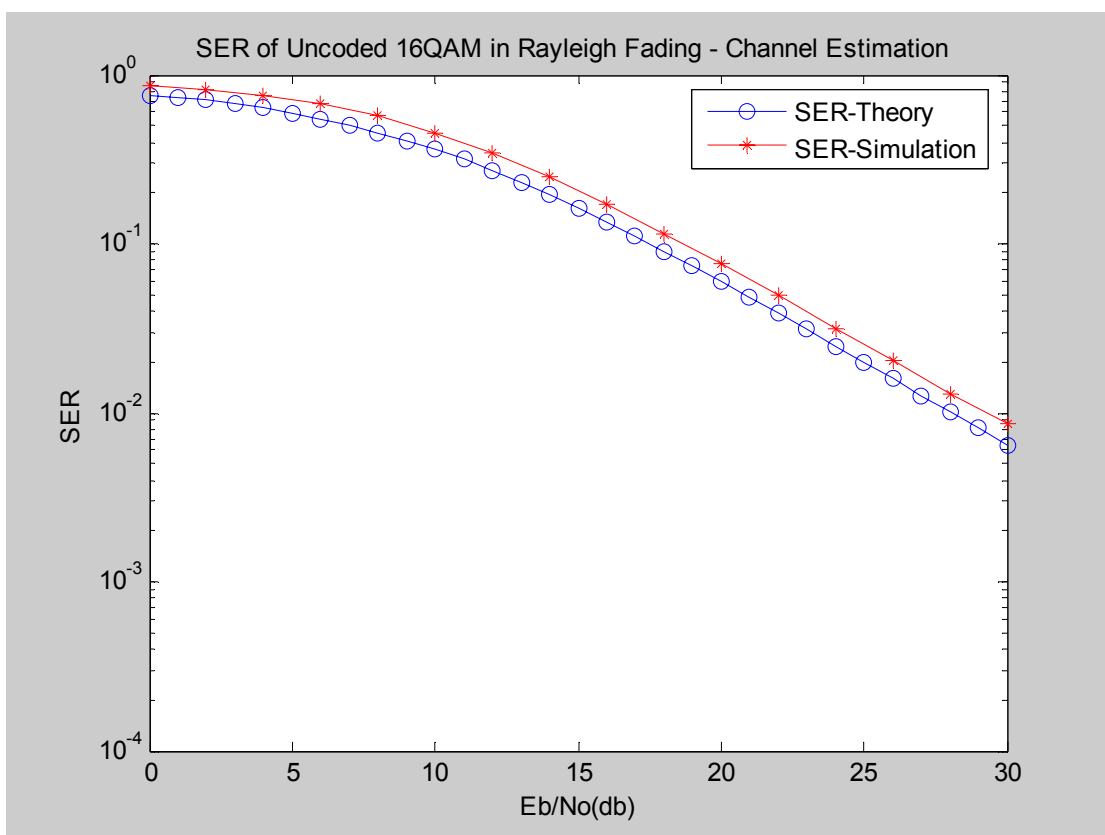
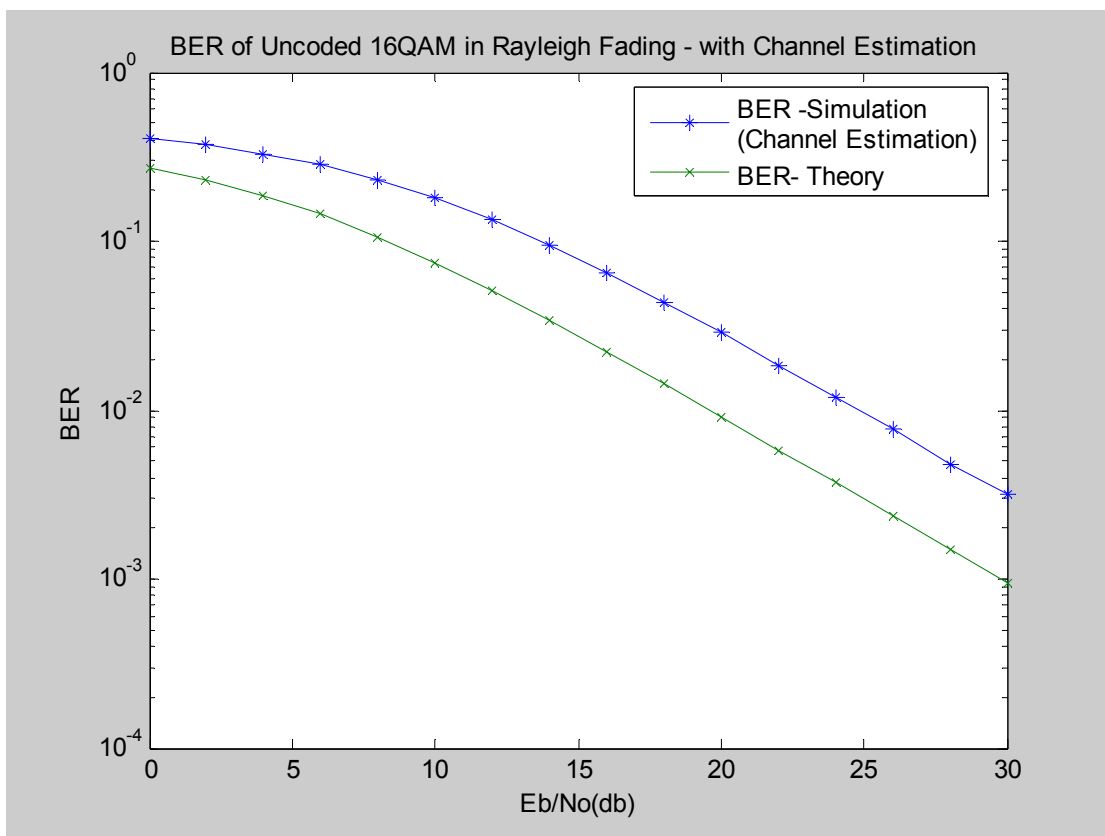
(1) Διαμόρφωση BPSK



(2) Διαμόρφωση QPSK

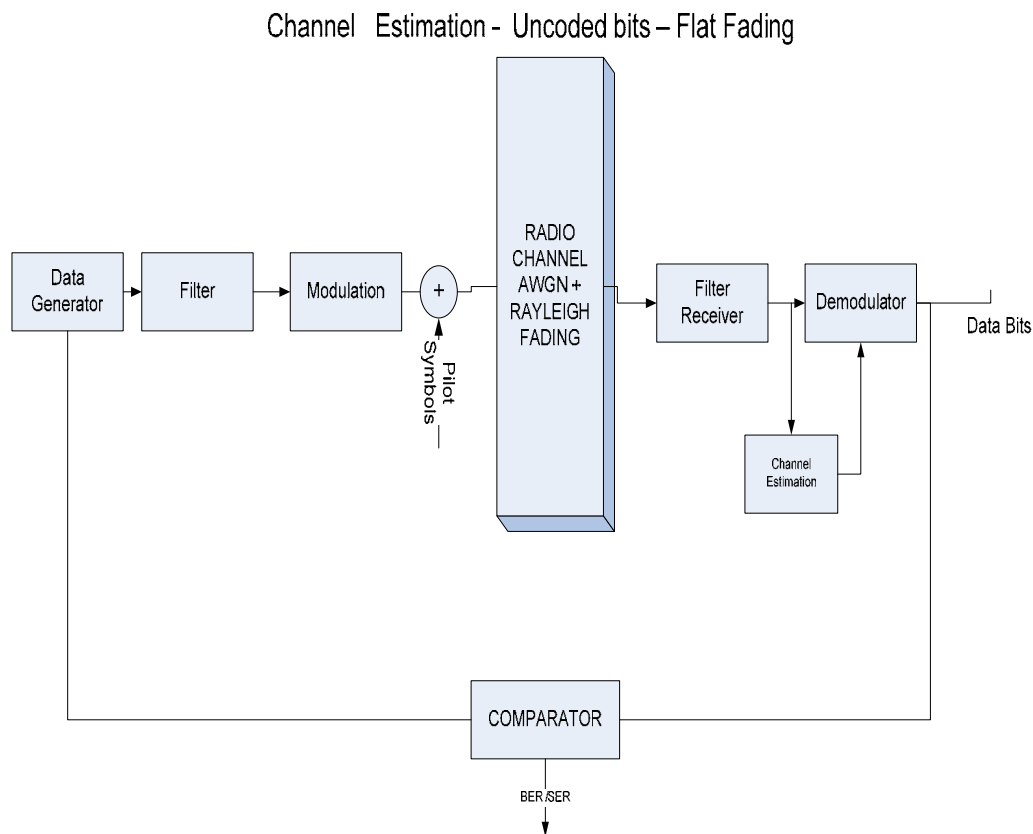


(3) Διαμόρφωση 16QAM



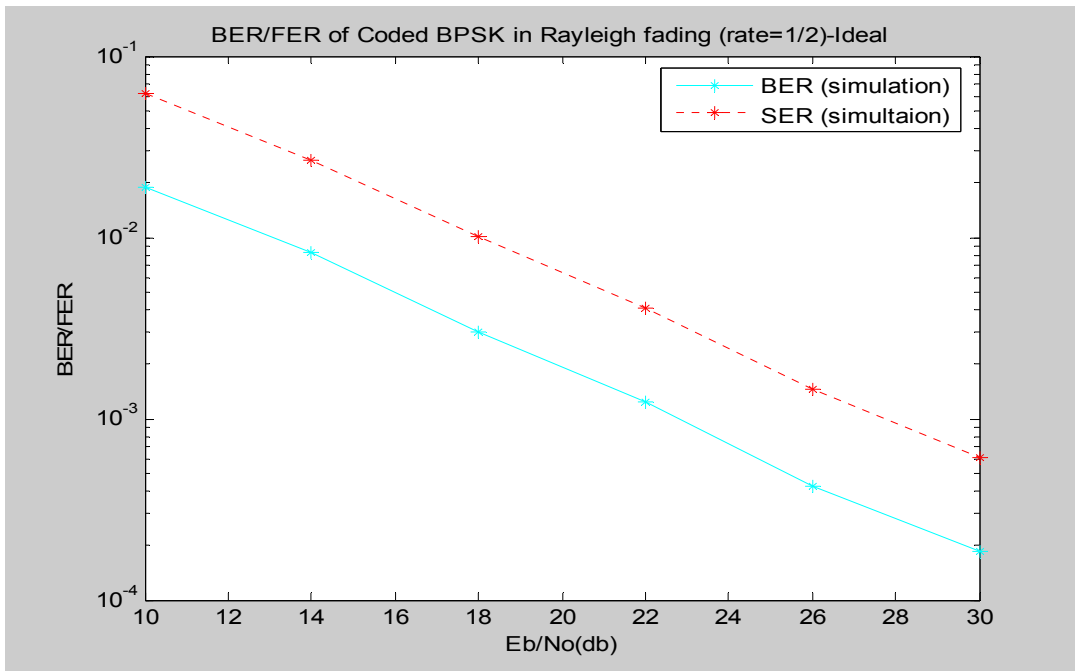
(3) Προσομοίωση **coded Data** σε Rayleigh **Flat** Fading AWGN κανάλι

Η περίπτωση αυτή δεικνύεται στο σχήμα 7.3, και εδώ υπάρχει μία σημαντική διαφορά αυτή των κωδικοποιημένων δεδομένων προς μετάδοση με έναν convolutional encoder, ο οποίος έχει με συγκεκριμένα χαρακτηριστικά (constrain length, generator polynomial κλπ). Αντίστοιχα η αποκωδικοποίηση των δεδομένων γίνεται με Viterbi και εδώ παρουσιάζονται οι σχετικές γραφικές παραστάσεις οι οποίες περιέχουν και το Bit Error Rate (BER), αλλά και το FER (Frame Error Rate) σε μία παράσταση.

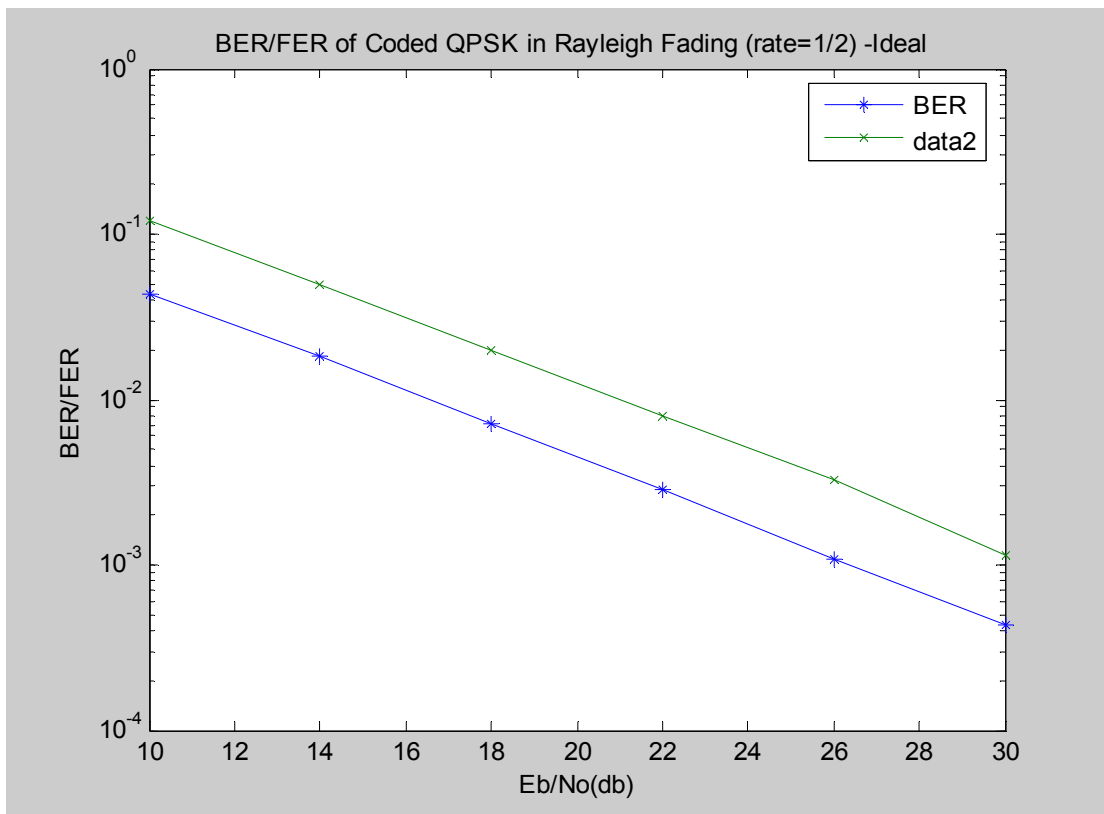


Σχήμα 7.3

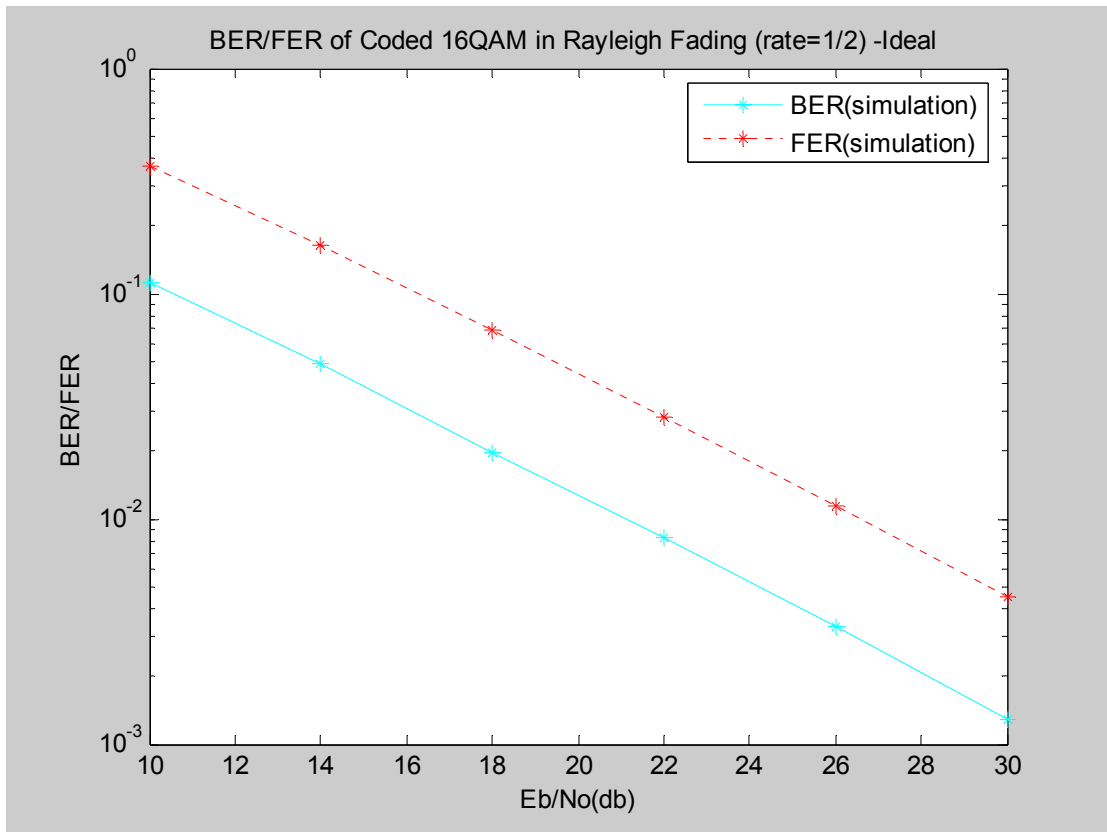
(1) Διαμόρφωση BPSK



(2) Διαμόρφωση QPSK

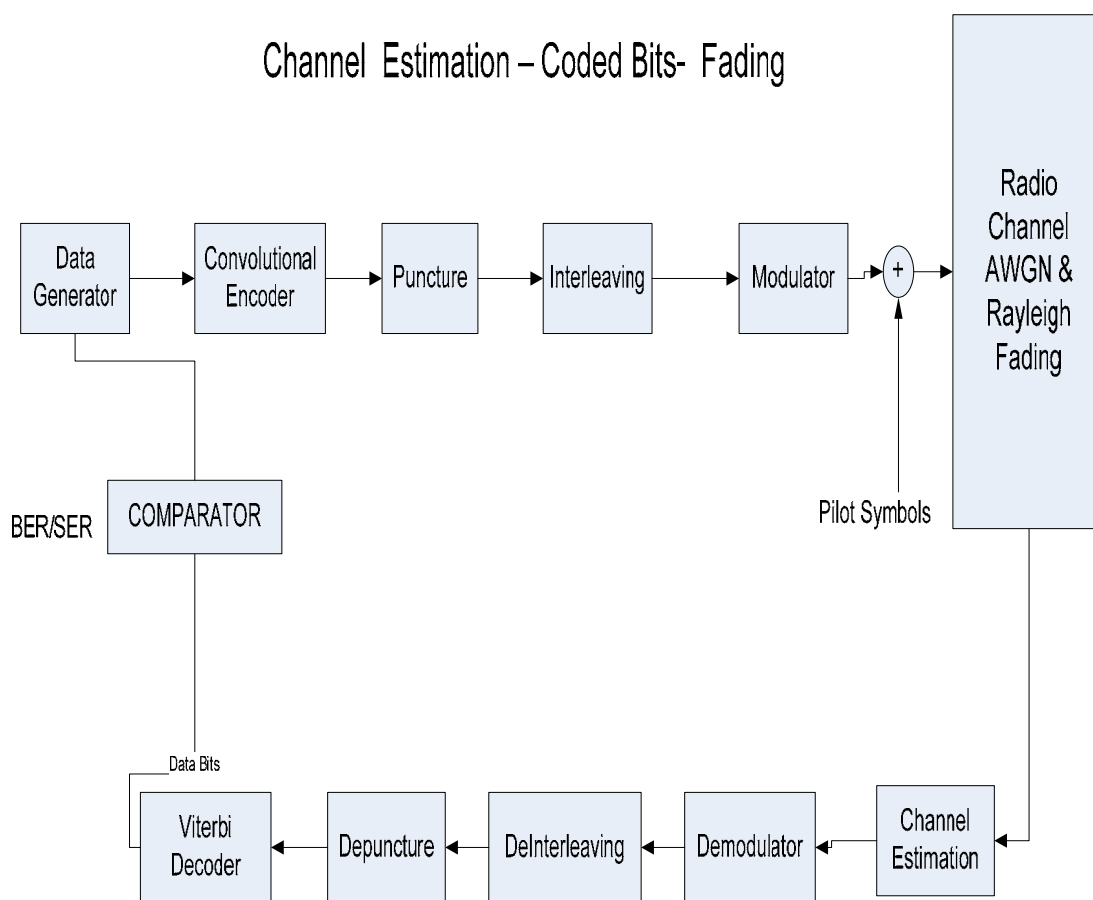


(3) Διαμόρφωση 16 QAM



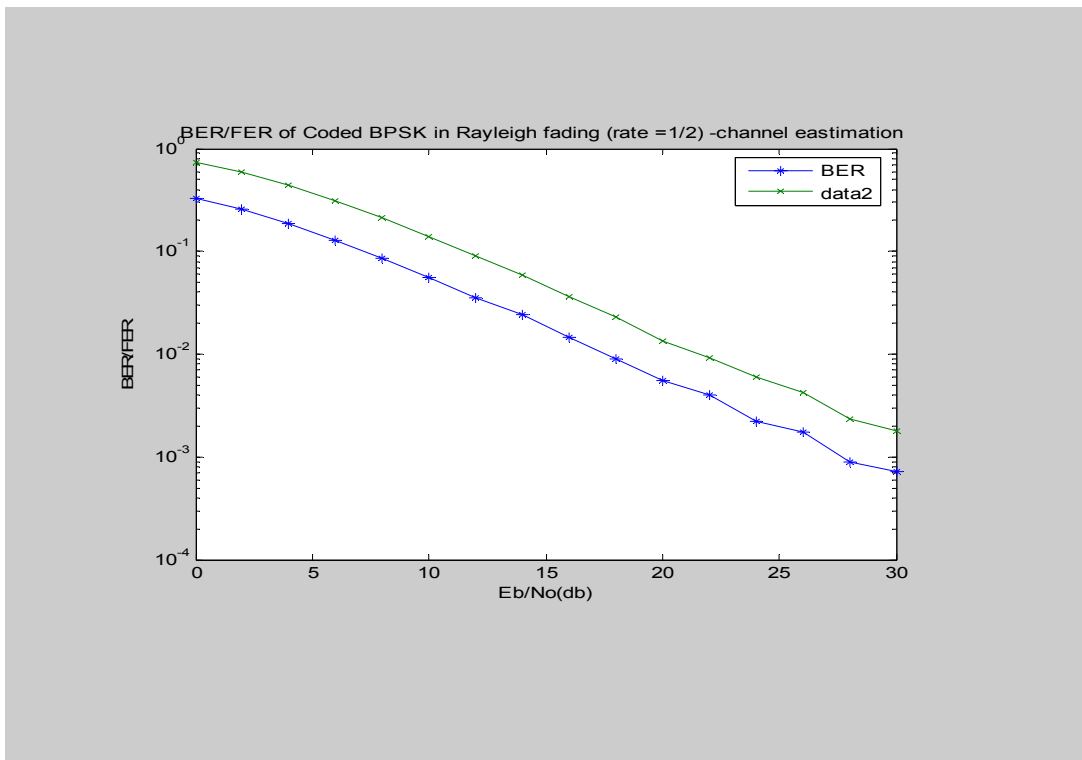
(4) Προσομοίωση **coded Data** σε Rayleigh **Normal** Fading AWGN κανάλι .

Η τελευταία περίπτωση είναι αυτή του σχήματος 7.4 η οποία ουσιαστικά είναι ίδια με την περίπτωση (3) , αλλά εδώ το fading είναι normal , δηλαδή επηρεασμός πλάτους και φάσης.

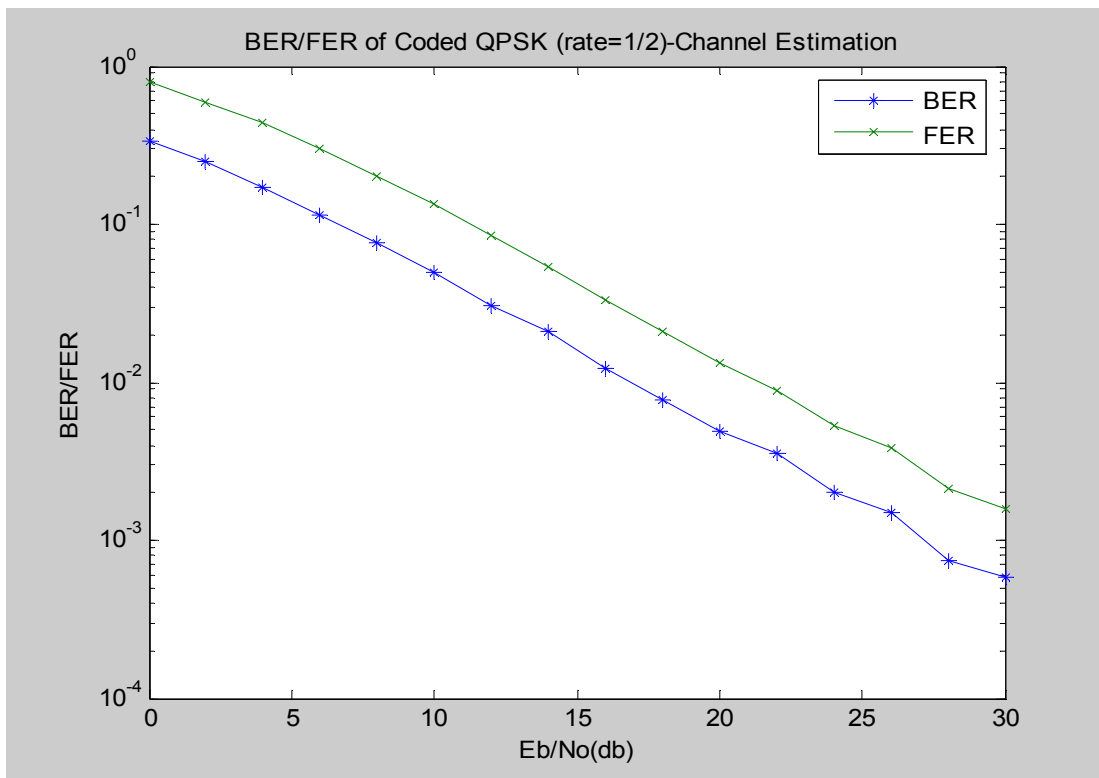


Σχήμα 7.4

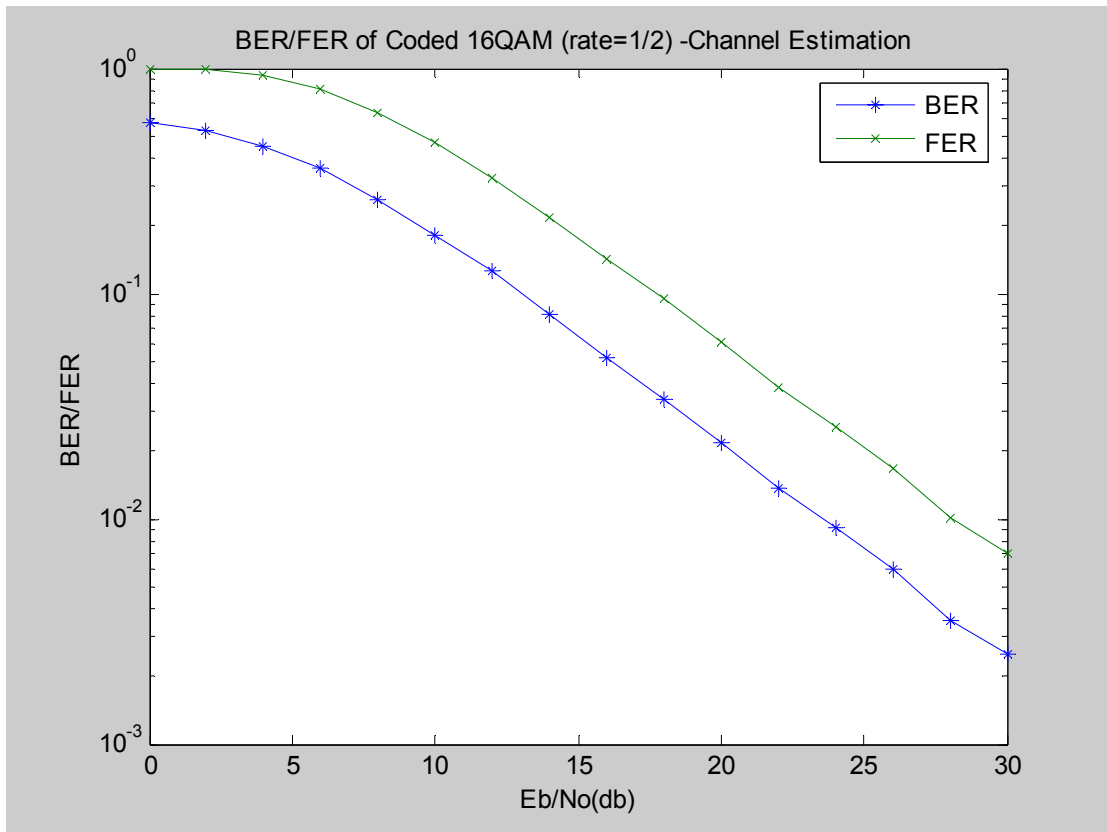
(1) Διαμόρφωση BPSK



(2) Διαμόρφωση QPSK



(3) Διαμόρφωση 16QAM



ΚΕΦΑΛΑΙΟ 8

ΣΥΜΠΕΡΑΣΜΑΤΑ

Η παρούσα εργασία αποσκοπεί στο να μελετήσει σε κάποιο βαθμό και να καταβληθεί προσπάθεια να εφαρμοστεί μία καινούργια τεχνική αυτή των adaptive transmission με τον υπολογισμό των παραμέτρων του καναλιού μετάδοσης.

Σύμφωνα με την δομή της διπλωματικής τα αρχικά κεφάλαια πραγματεύονται το θεωρητικό μέρος και υπόβαθρο , ενώ στο τελευταίο κεφάλαιο αναλύεται η εργασία που πραγματοποιήθηκε.

Με την παρούσα διπλωματική πραγματοποιήθηκαν τα ακόλουθα :

1. Δημιουργία λογισμικού , με βασικό εργαλείο το MATLAB , το οποίο Προσημειώνει ένα ασυρματικό σύστημα μετάδοσης δεδομένων ταχύτητας 256kb/s.

2. Η εξέταση της συμπεριφοράς –απόδοσης του εν λόγω συστήματος όσο αφορά το ρυθμό εμφάνισης σφαλμάτων στο επίπεδο του bit (Bit Error Rate) , στο επίπεδο συμβόλου (Symbol Error Rate) και στο επίπεδο του Frame (Frame Error Rate) . Ο έλεγχος αυτός έγινε κάτω από ορισμένες και συγκεκριμένες προϋποθέσεις όσο αφορά την συμπεριφορά του καναλιού μετάδοσης , το μοντέλο που χρησιμοποιήθηκε για να προσομοιωθεί , καθώς επίσης και των μεταδιδόμενων data (Uncoded, Coded) .

3. Εφαρμογή της τεχνικής για τον υπολογισμό της παραμέτρου του καναλιού (κέρδους) με τη χρήση του λογισμικού Channel_Estimation και τη χρησιμοποίηση των pilot symbols για τον υπολογισμό αυτόν.

4. Εξήχθησαν οι σχετικές γραφικές παραστάσεις εφαρμόζοντας κατά περίπτωση τις κατάλληλες παραμέτρους , δηλαδή :

(α) Εξαγωγή του BER και του SER για την περίπτωση που έχουμε μετάδοση μη κωδικοποιημένων (Uncoded) data και η συμπεριφορά του καναλιού όσο αφορά το fading θεωρείται flat . Η κατανομή που χρησιμοποιείται σε κάθε περίπτωση είναι η Rayleigh .

Από τις γραφικές παραστάσεις παρατηρούμε ότι τα αποτελέσματα που δίνει το λογισμικό προσομοίωσης σχεδόν ταυτίζονται με αυτά που δίνονται από τις σχετικές θεωρητικές εξισώσεις. Οι σχετικές γραφικές παραστάσεις παρουσιάζονται με τον τίτλο BER/SER SIMULATION BPSK/QPSK/16QAM/ UNCODED /IDEAL

(β) Εξαγωγή των BER/SER για την περίπτωση όπου όλα τα χαρακτηριστικά είναι ταυτόσημα με τα ανωτέρω εκτός από δύο σημαντικές διαφορές :

- Της εφαρμογής της τεχνικής υπολογισμού του καναλιού (channel estimation) με την χρήση των Pilot symbols.
- Το κανάλι παρουσιάζει normal fading , δηλαδή έχουμε μεταβολή όχι μόνον του πλάτους του σήματος αλλά και της φάσης. Αυτό έχει σαν αποτέλεσμα να απαιτείται πολύ μεγαλύτερη ισχύ για να μεταδοθούν data , μη κωδικοποιημένα , ίδιου ρυθμού μετάδοσης.

Με τις γραφικές παραστάσεις αυτές, παρατηρούμε ότι με την ίδια ισχύ που χρησιμοποιούμε όταν δεν έχουμε channel estimation , μπορούμε χρησιμοποιώντας αυτήν την τεχνική να μεταδώσουμε data με τον ίδιο ρυθμό , σε περιβάλλον «χειρότερο» απ' αυτό της περίπτωσης που έχουμε flat fading. Ως εκ τούτου βλέπουμε ότι με την τεχνική του υπολογισμού των παραμέτρων του καναλιού σε πραγματικό χρόνο , δηλαδή κατά την μετάδοση , και εδώ συγκεκριμένα για το κέρδος του , μπορούμε να βελτιώσουμε τις συνθήκες μετάδοσης , αλλά και να χρησιμοποιηθεί για την σε real time μεταβολή των παραμέτρων μετάδοσης (διαμόρφωσης , κωδικοποίησης κλπ).

Οι σχετικές γραφικές παραστάσεις παρουσιάζονται με τον τίτλο BER/SER SIMULATION BPSK/QPSK/16QAM /UNCODED / CHANNEL ESTIMATION

(γ) Τέλος όλα τα παραπάνω πραγματοποιήθηκαν και για την περίπτωση των κωδικοποιημένων data (Coded rate =1/2) , ενώ χρησιμοποιήθηκε αποκωδικοποίηση κατά Viterbi. Και εδώ τα αποτελέσματα ήταν ανάλογα , με την έννοια ότι με μικρή αύξηση του ber , αλλά με την ίδια ισχύ μπορούμε να μεταδώσουμε data σε περιβάλλον με fading. Οι σχετικές γραφικές παραστάσεις δίνονται με τον τίτλο BER/FER SIMULATION BPSK/QPSK 16QAM/ CODED /IDEAL / CHANNEL ESTIMATION

ΠΑΡΑΡΤΗΜΑ Ι
ΛΟΓΙΣΜΙΚΟ ΠΡΟΣΟΜΕΙΩΣΗΣ

```
***** Preparation part *****

clear;
global pilotlen M
system = 'coded ';
estimation = 'pilot';
fd = 30;
M = 16; % # of symbols
N_bits = 200; % Number of Bits that simulates in each
loop

codeRate = 1/2;
pattern = [1];
step = 2.0;
EsNodBmin = 0.0;
EsNodBmax = 30.0;
nloop=[1000 2000 3000 4000 5000 6000 7000 8000 10000 15000 20000
25000 30000 40000 50000 60000]; % Number of simulation loops
pilotlen = 12;
br = 2560000.0; % Bit rate
k = log2(M); % # of bits/symbol
sr = br / k; % Symbol rate
N_syms = N_bits/k; % Number of symbols that simulates in
each loop
numSamplesPerSymbol=8; % Number of oversamples per symbol
fs=numSamplesPerSymbol;
nd = 100; % Number of blocks that skips every
loop for the channel

%*****Convolutional Encoding initialization *****

constlen = 7;
codegen = [171 133];
tblen = 32; % traceback length
trellis = poly2trellis(constlen, codegen);
tail = (constlen - 1);
k_rate = codeRate*k; % effect of coding
st2 = 4831; % state for random interleaving

%*****Filter initialization *****

irfn=8; % Number of syms inside filter
alfs=0.5; % Rolloff factor
[hTransmitFilter] = hrollfcoef(irfn,fs,sr,alfs,1); % Transmitter
filter coefficients
[hReceiveFilter] = hrollfcoef(irfn,fs,sr,alfs,0); %Receiver filter
coefficients

%***** Fading initialization *****

% Time resolution
tstp=1/sr/numSamplesPerSymbol;
n0=[6];
itnd0=nd*numSamplesPerSymbol*N_syms;
itnd1=[1000];
if estimation=='ideal'
    flat=1;
else
    flat =0;
```



```
end
%1->flat (only amplitude is fluctuated), 0->nomal(phase and amplitude
are fluctutated

%***** START CALCULATION*****

EsNodB = EsNodBmin:step:EsNodBmax;

for i=1:length(EsNodB)
% Αρχίζει το test για κάποιο SNR
Errors_sym = 0;
Errors_bit = 0;
Errors_frame = 0;
Errors = 0;
Blck_Errors=0;

    for iters = 1:nloop(i)
% Ο Πομπός (Transmitter) στέλνει κάθε φορά πακέτα με N_bits
% Για κάθε test ο συνολικός αριθμός των bits που στέλνονται είναι
% nloop * N_bits

if system == 'uncoded'
BitStreamLength = N_bits;
x = randsrc(N_bits, 1, 0:1);
elseif system == 'coded '
    BitStreamLength = (N_bits-tail)*codeRate;
    msg_orig = randsrc(N_bits, 1, 0:1);
    msg_orig_tail = [msg_orig; zeros(tail, 1)];
    [msg_enc_bi] = convenc(msg_orig_tail, trellis);
    punctcode=puncture(msg_enc_bi,pattern);
    x = randintrlv(punctcode,st2);
end

if (M==2)
    xsym = x;
    y = qammod(xsym, M);
    ich = real(y);
    qch = imag(y);
elseif (M==4)
    [BitStreamOne,BitStreamTwo] = SerialToParallel(x);
    [ich,qch] = QPSKEncoder(BitStreamOne,BitStreamTwo);
    y = ich + j*qch;
    xsym = qamdmod(y, M); % symbol for comparing at receiver
elseif (M==16)
    if system == 'uncoded'
        xsym = bi2de(reshape(x,k,length(x)/k).','left-msb');
        y = qammod(xsym, M);
    else
        mapping = [3 2 0 1 7 6 4 5 15 14 12 13 11 10 8 9].';
        xsym = bi2de(reshape(x,k,length(x)/k).','left-msb');
        sym = mapping(xsym+1);
        y = qammod(sym, M);
    end
    ich = real(y);
    qch = imag(y);
end

if estimation == 'pilot'
pilot=(sqrt(M)-1)*ones(pilotlen,1)+j*(sqrt(M)-1)*ones(pilotlen,1);
```

```
tx=[pilot.' y.'];

ich = real(tx);
qch = imag(tx);
end

[ich2, qch2]=TransmitFilter(ich, qch,
hTransmitFilter,numSamplesPerSymbol);

%***** Fading channel *****

% Generated data are fed into a fading simulator
[ifade, qfade, ramp]=fade(ich2, qch2, length(ich2), tstp, fd, n0,
itnd1, flat);
% Update fading counter σε κάθε loop!!
itnd1 = itnd1+ itnd0;

SNR = 10^(EsNodB(i)/10); % μετατροπή του SNR από dB σε κανονική
τιμή
Es = sum(ich2.^2+qch2.^2)/length(ich2);
No = Es/SNR; % SNR = Es/No
sigma = sqrt(fs*No/2);

lenWaveform = length(ifade);
I_WaveformRx = ifade + sigma*randn(1, lenWaveform);
Q_WaveformRx = qfade + sigma*randn(1, lenWaveform);

%***** Automatic Gain Control (AGC)*****

ramp16 =sum(sqrt(I_WaveformRx.^2 +
Q_WaveformRx.^2))/length(I_WaveformRx);
I_WaveformRx = I_WaveformRx./ramp16;
Q_WaveformRx = Q_WaveformRx./ramp16;

%*****Receiver with Matched Filter*****

Filter_delay = irfn*fs;
[rx_ich, rx_qch] =
ReceiveFilter(I_WaveformRx,Q_WaveformRx,hReceiveFilter,
numSamplesPerSymbol, Filter_delay);

if estimation == 'pilot'
[rx_ich_est, rx_qch_est] = chan_est(rx_ich, rx_qch);
clear rx_ich rx_qch;
rx_ich = rx_ich_est;
rx_qch = rx_qch_est;
end

if system == 'uncoded'
if (M==2)
yrx = 2*(rx_ich + j*rx_qch);
zsym = qamdemod(yrx, M);
zbit = zsym;
elseif (M==4)
yrx = 2*(rx_ich + j*rx_qch);
[ib,qb]=QPSKDecoder(rx_ich,rx_qch);
```

```
    BitStreamOneRX=ib';
    BitStreamTwoRX=qb';
    zsym = qamdemod(yrx, M);
    zbit=ParallelToSerial(BitStreamOneRX,BitStreamTwoRX,2);
elseif (M==16)
    yrx = 3*(rx_ich + j*rx_qch);
    zsym = qamdemod(yrx, M);
    z = de2bi(zsym, 4, 'left-msb'); % Convert integers to bits.
    zbit = reshape(z.',prod(size(z)),1); % Convert z from a matrix to
                                        a vector.
End

%*****Calculations of BER/SER/FER*****
% Symbol Error Rate
if (M==2)
    Errors_sym = Errors_sym + sum(xsym~=zsym');
elseif (M==4)
    Errors_sym = Errors_sym + sum(xsym~=zsym');
elseif (M==16)
    Errors_sym = Errors_sym + sum(xsym~=zsym');
end

% Bit Error Rate
if (M==2)
    Errors_bit = Errors_bit + sum(x~=zbit');
elseif (M==4)
    Errors_bit = Errors_bit + sum(x~=zbit');
elseif (M==16)
    Errors_bit = Errors_bit + sum(x~=zbit');
end

% Frame Error Rate
if (M==16)
    zbit = zbit';
end

if sum(x~=zbit') ~= 0
    Errors_frame = Errors_frame + 1;
end

elseif system == 'coded '

BitStreamRx = ParallelToSerial(rx_ich, rx_qch, k);
deinter = randeintrlv(real(BitStreamRx),st2);
BitStreamRx_unpunct=depuncture(deinter, pattern);

% Quantize to prepare for soft-decision decoding.
qcode = quantiz(real(BitStreamRx_unpunct),[-1.0 -.75, -.25,
0, .25, .5, .75,1.0]);

msg_dec = vitdec(real(qcode), trellis, tble, 'term', 'soft',3);

Errors = Errors + sum(msg_orig(1:end)~=msg_dec(1:end-tail));

% Calculate Frame Error Rate
if (sum(msg_orig(1:end)~=msg_dec(1:end-tail)))~=0
    Blck_Errors = Blck_Errors + 1;
end
```

```
end      % coded

      end      % LOOPS

if system == 'uncoded'
EbNodB(i) = EsNodB(i)
SER_simulation(i) = Errors_sym / (length(xsym)*nloop(i))
BER_simulation(i) = Errors_bit / (length(x)*nloop(i))
FER_simulation(i) = Errors_frame/nloop(i)
elseif system == 'coded '
EbNodB(i) = EsNodB(i)
BER_simulation(i) = Errors / (length(msg_orig(1:end-tblen))*nloop(i))
FER_simulation(i) = Blck_Errors/nloop(i)
end
end      % SNRs
if system == 'uncoded'
    % Theoretically Bit Error Rate
    ebno_max=30;
    ebno_min=0;
    ebno=ebno_min:2:ebno_max;
    if (M==16)
        for j=1:length(ebno);
            w=10^(ebno(j)/10);
            a=sqrt(w);
            BER_fading(j)=3/8*(1-(1/(1+5/(2*w)))));
        end
    else
        for j=1:length(ebno);
            w=10^(ebno(j)/10);
            a=1+(1/w);
            b=sqrt(a);
            m=1/b;
            BER_fading(j)=1/2*(1-m);
        end
    end
end

%*****PLOTTING FIGURES *****

semilogy(EbNodB, BER_simulation, '-*', ebno, BER_fading, '-x')
figure(1)
semilogy(EbNodB, SER_simulation, '-*r')
end
if system == 'coded '
semilogy(EbNodB, BER_simulation, '-*', EbNodB, FER_simulation, '-x')
end
```

*******Παραγωγή συντελεστών φίλτρου Nyquist *******

```
% hrollfcoef.m
%
% Generate coefficients of Nyquist filter
%
% programmed by H.Harada
%

function [xh] = hrollfcoef(irfn,ipoint,sr,alfs,ncc)

%***** variables *****
% irfn   : Number of symbols to use filtering
% ipoint : Number of samples in one symbol
% sr     : symbol rate
% alfs   : rolloff coeficiense
% ncc    : 1 -- transmitting filter  0 -- receiving filter
% *****

xi=zeros(1,irfn*ipoint+1);
xq=zeros(1,irfn*ipoint+1);

point = ipoint;
tr = sr ;
tstp = 1.0 ./ tr ./ ipoint;
n = ipoint .* irfn;
mid = ( n ./ 2 ) + 1;
sub1 = 4.0 .* alfs .* tr;          % 4*alpha*R_s

for i = 1 : n

    icon = i - mid;
    ym = icon;

    if icon == 0.0
        xt = (1.0-alfs+4.0.*alfs./pi).* tr; % h(0)
    else
        sub2 =16.0.*alfs.*alfs.*ym.*ym./ipoint./ipoint;
        if sub2 ~= 1.0
            x1=sin(pi*(1.0-alfs)/ipoint*ym)./pi./(1.0-sub2)./ym./tstp;
            x2=cos(pi*(1.0+alfs)/ipoint*ym)./pi.*sub1./(1.0-sub2);
            xt = x1 + x2; % h(t) plot((1:length(xh)),xh)
        else % (4alphaRst)^2 = 1plot((1:length(xh)),xh)
            xt = alfs.*tr.*((1.0-
2.0/pi).*cos(pi/4.0/alfs)+(1.0+2.0./pi).*sin(pi/4.0/alfs))./sqrt(2.0)
;
        end % if sub2 ~= 1.0
    end % if icon == 0.0

    if ncc == 0 % in the case of receiver
        xh( i ) = xt ./ ipoint ./ tr; % normalization
    elseif ncc == 1 % in the case of transmitter
        xh( i ) = xt ./ tr; % normalization
    else
        error('ncc error');
    end % if ncc == 0

end % for i = 1 : n
```

```
%***** end of file *****
```

*******Μετατροπή παράλληλου BitStream σε Σειριακό

```
function BitStream = ParallelToSerial(BitStreamOne, BitStreamTwo, k)
```

```
lenBitStream = k*length(BitStreamOne);
```

```
if k==1
BitStream(1:1:lenBitStream) = BitStreamOne;

elseif k==2
BitStream(1:2:lenBitStream) = BitStreamOne;
BitStream(2:2:lenBitStream) = BitStreamTwo;

elseif k==4
BitStream(1:4:lenBitStream) = BitStreamOne;
BitStream(2:4:lenBitStream) = (2.0/sqrt(10.0)) - abs(BitStreamOne);
BitStream(3:4:lenBitStream) = BitStreamTwo;
BitStream(4:4:lenBitStream) = (2.0/sqrt(10.0)) - abs(BitStreamTwo);

end
```

*******Υλοποίηση QPSK Decoder*******

```
function [BitStreamOneRx, BitStreamTwoRx] = QPSKDecoder(I_SymbolsRx,
Q_SymbolsRx)

BitStreamOneRx = I_SymbolsRx > 0;
BitStreamTwoRx = Q_SymbolsRx > 0;
```

*******Δημιουργία κυματομορφών*******

```
function [I_SymbolsRx,Q_SymbolsRx] =  
ReceiveFilter(I_WaveformRx,Q_WaveformRx,hReceiveFilter,...  
    numSamplesPerSymbol,Delay)
```

```
% Κάνουμε το matched filtering πρώτα:  
I_FilterOutput = conv(I_WaveformRx, hReceiveFilter);  
Q_FilterOutput = conv(Q_WaveformRx, hReceiveFilter);
```

```
%Εδώ υποθέτουμε ότι τα φίλτρα transmit και receive έχουν το ίδιο  
μήκος.
```



```
SymbolRange = Delay+1:length(I_FilterOutput)-Delay;  
I_SymbolsRx =  
WaveformToSymbol(I_FilterOutput(SymbolRange), numSamplesPerSymbol);  
Q_SymbolsRx =  
WaveformToSymbol(Q_FilterOutput(SymbolRange), numSamplesPerSymbol);
```

*******Θεωρητικός Υπολογισμός SER για την 16QAM με MGF

```
% Evaluates SER for MQAM with MGF  
clear;  
global g m p W;  
  
M=16;  
K=4/pi;  
q=1-(1/sqrt(M));  
g=3/(2*(M-1));  
  
m=[1];  
W=[1];  
L=length(m);  
A1=pi/2;  
B1=sin(A1)*sin(A1);
```

```
A2=pi/4;
B2=sin(A2)*sin(A2);

j=1;
for esno=0:30
    p=10^(esno/10);    % target Es/No

    sum1=(4/pi)*q*quadl(@mgf,0,pi/2,10^(-6));
    sum2(j)=(4/pi)*(q^2)*quadl(@mgf,0,pi/4,10^(-6));
    ser_theory(j)=sum1-sum2(j);

    xx(j)=esno;
    j=j+1;
end

figure(1)
semilogy(xx, ser_theory, '-o')
axis([0 30 10^(-4) 10^0])
hold on
```

*******Μετατροπή συμβόλων σε κυματομορφή

```
function Waveform = SymbolToWaveform(SymbolsTx, numSamplesPerSymbol)

len_Waveform = length(SymbolsTx)*numSamplesPerSymbol;
Waveform = zeros(1, len_Waveform);
Waveform(1:numSamplesPerSymbol:len_Waveform) = SymbolsTx;
```

*******Μετατροπή κυματομορφής σε σύμβολα

```
function SymbolStream = WaveformToSymbol(Waveform,  
numSamplesPerSymbol)
```

```
SymbolStream = Waveform(1:numSamplesPerSymbol:length(Waveform));
```

*******Εφαρμογή τεχνικής CHANNEL ESTIMATION

```
function [rx_i_ce, rx_q_ce] = chan_est(rx_ch_i, rx_ch_q)

global pilotlen M

i_ce = 0;
q_ce = 0;

if M==2
    for i=1:pilotlen
        i_ce = i_ce + rx_ch_i(i);
        q_ce = q_ce + rx_ch_q(i);
    end
elseif M==4
    for i=1:pilotlen
        i_ce = i_ce + (rx_ch_i(i) + rx_ch_q(i))/2.0;
        q_ce = q_ce + (rx_ch_q(i) - rx_ch_i(i))/2.0;
    end
elseif M==16
    for i=1:pilotlen
        i_ce = i_ce + (rx_ch_i(i) + rx_ch_q(i))/6.0;
        q_ce = q_ce + (rx_ch_q(i) - rx_ch_i(i))/6.0;
```

```
    end
end

% CHANNEL ESTIMATION
i_ce = i_ce/pilotlen;
q_ce = q_ce/pilotlen;

RxData = rx_ch_i + j*rx_ch_q;
IQ_ch = (i_ce + j* q_ce);

% Correct channel effect
RX_symbols = RxData * conj(IQ_ch)/sqrt(i_ce.^2 + q_ce.^2);

RX = RX_symbols(pilotlen+1:end);
rx_i_ce = real(RX);
rx_q_ce = imag(RX);
```

*****Αημιουπγία Rayleigh fading (flat/normal)*****

```
% fade.m
%
% Generate Rayleigh fading
%
% Programmed by H.Harada

function
[iout,qout,ramp,rcos,rsin]=fade(idata,qdata,nsamp,tstp,fd,no,counter,
flat)

%***** variables *****
% idata : input Ich data
% qdata : input Qch data
% iout  : output Ich data
% qout  : output Qch data
% ramp  : Amplitude contaminated by fading
% rcos  : Cosine value contaminated by fading
% rsin  : Cosine value contaminated by fading
% nsamp : Number of samples to be simulated
% tstp  : Minimum time resolution
% fd    : maximum doppler frequency
% no    : number of waves in order to generate fading
% counter : fading counter
% flat  : flat fading or not
```

```
% (1->flat (only amplitude is fluctuated),0->nomal(phase and
amplitude are fluctutated)
%*****

if fd ~= 0.0
    ac0 = sqrt(1.0 ./ (2.0.*(no + 1)));    % power normalized
constant(ich)
    as0 = sqrt(1.0 ./ (2.0.*no));        % power normalized
constant(qch)
    ic0 = counter;                       % fading counter

    pai = 3.14159265;
    wm = 2.0.*pai.*fd;
    n = 4.*no + 2;
    ts = tstp;
    wmts = wm.*ts;
    paino = pai./no;

    xc=zeros(1,nsamp);
    xs=zeros(1,nsamp);
    ic=[1:nsamp]+ic0;

    for nn = 1: no
        cwn = cos( cos(2.0.*pai.*nn./n).*ic.*wmts );
        xc = xc + cos(paino.*nn).*cwn;
        xs = xs + sin(paino.*nn).*cwn;
    end

    cwmt = sqrt(2.0).*cos(ic.*wmts);
    xc = (2.0.*xc + cwmt).*ac0;
    xs = 2.0.*xs.*as0;

    ramp=sqrt(xc.^2+xs.^2);
    rcos=xc./ramp;
    rsin=xs./ramp;

    if flat ==1
        iout = sqrt(xc.^2+xs.^2).*idata(1:nsamp);    % output signal(ich)
        qout = sqrt(xc.^2+xs.^2).*qdata(1:nsamp);    % output signal(qch)
    else
        iout = xc.*idata(1:nsamp) - xs.*qdata(1:nsamp);    % output
signal(ich)
        qout = xs.*idata(1:nsamp) + xc.*qdata(1:nsamp);    % output
signal(qch)
    end
end

else
    iout=idata;
    qout=qdata;
end

% *****end of
file*****
```

*******Υλοποίηση Moment Generation
Function*******

```
function y = mgf(phi)

global g m p W;

tmp = sin(phi).*sin(phi);
y1=1;
for i=1:length(m)
y1 = y1.*(1+(g.*p.*W(i)./(m(i).*tmp))).^(-m(i)));
end
y=y1;
```

*******Διαδικασία Puncture *******

```
function y = puncture(msg,pattern)

if (length(pattern)==1)
    y = msg;

elseif (length(pattern)==18)
    % Puncture by the pattern '111001111001111001'
    for jj=1:length(msg)/18
        y((jj-1)*12+1:(jj-1)*12+3)=msg((jj-1)*18+1:(jj-1)*18+3);
        y((jj-1)*12+4:(jj-1)*12+7)=msg((jj-1)*18+6:(jj-1)*18+9);
        y((jj-1)*12+8:(jj-1)*12+11)=msg((jj-1)*18+12:(jj-1)*18+15);
        y((jj-1)*12+12)=msg((jj-1)*18+18);
    end
end
```


*******Υλοποίηση QPSK Encoder*******

```
function  
[I_SymbolsTx,Q_SymbolsTx]=QPSKEncoder(BitStreamOne,BitStreamTwo)  
I_SymbolsTx = 2*BitStreamOne-1;  
Q_SymbolsTx = 2*BitStreamTwo-1;
```

*******Θεωρητικός Υπολογισμός SER για BPS/QPSK με MGF

```
% Evaluate SER with mgf method
clear;
global g m p W;

M=4;
A=pi-pi/M;

m=[1]
W=[1];
L=length(m);
g=sin(pi/M)*sin(pi/M);

j=1;
for esno=0:30;
    p=10^(esno/10);      % target Es/No

    ser_theory(j)=(1/pi)*quadl(@mgf,0,A,10^(-10));
    xx(j)=esno;
    j=j+1;
end
figure(1)
semilogy(xx, ser_theory, '-o')
axis([0 30 10^(-4) 10^0])
hold on
```

*******Μετατροπή Σειριακού Stream σε
Παράλληλο*******

```
function [BitStreamOne, BitStreamTwo] = SerialToParallel(BitStream)
BitStreamOne = BitStream(1:2:length(BitStream));
BitStreamTwo = BitStream(2:2:length(BitStream));
```

*******Δημιουργία κυματομορφών μετάδοσης *******

```
function [I_WaveformTx, Q_WaveformTx] =  
TransmitFilter(I_SymbolsTx, Q_SymbolsTx, hTransmitFilter, ...  
    numSamplesPerSymbol)  
  
% Το πρώτο βήμα είναι να μετατρέψουμε το εισερχόμενο symbol stream  
  
I_Waveform = SymbolToWaveform(I_SymbolsTx, numSamplesPerSymbol);  
Q_Waveform = SymbolToWaveform(Q_SymbolsTx, numSamplesPerSymbol);  
  
% Τα επόμενο βήμα είναι να φιλτράρουμε το σήμα ώστε να παραχθούν οι  
κυματομορφές μετάδοσης.  
  
I_WaveformTx = conv(I_Waveform, hTransmitFilter);  
Q_WaveformTx = conv(Q_Waveform, hTransmitFilter);
```

*******Διαδικασία Depuncture*******

```
function y = depuncture(msg, pattern)

if (length(pattern)==1)
    y = msg;

elseif (length(pattern)==18)
    % DEPuncture by the pattern '111001111001111001'
    y=zeros(1,length(msg)*18/12);
    for jj=1:length(msg)/12
        y((jj-1)*18+1:(jj-1)*18+3)=msg((jj-1)*12+1:(jj-1)*12+3);
        y((jj-1)*18+6:(jj-1)*18+9)=msg((jj-1)*12+4:(jj-1)*12+7);
        y((jj-1)*18+12:(jj-1)*18+15)=msg((jj-1)*12+8:(jj-1)*12+11);
        y((jj-1)*18+18)=msg((jj-1)*12+12);
    end
end
```

ΒΙΒΛΙΟΓΡΑΦΙΑ

- [1] Wireless Communications
A.Goldsmith
- [2] Simulation and Software radio for Mobile Communications –
Hirochi Harada , Ramjee Prasad
- [3] Digital Communications – Fourth Edition
John Proakis
- [4] Digital Signal Processing – Principles, Algorithms and Applications
John Proakis , Dimitris G. Manolakis /1996
- [4] Error Control Coding – Second Edition
Shu Lin . Daniel J.Costello, Jr/
- [5] Digital Communications , Fundamentals & Applications
Bernard Sklar