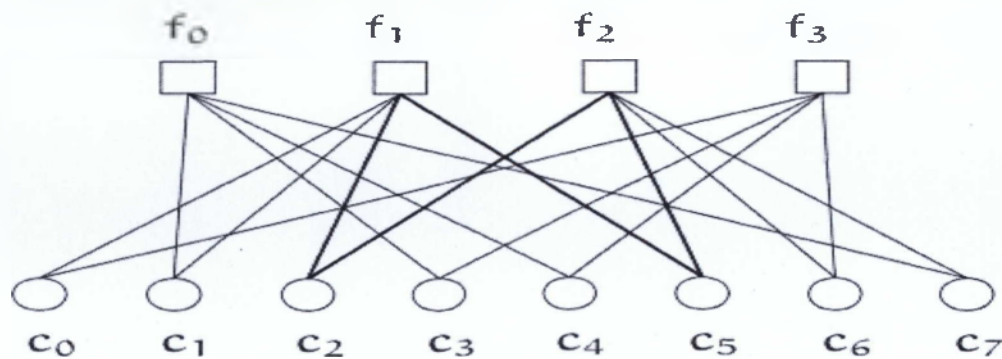


ΤΕΙ ΠΕΛΟΠΟΝΝΗΣΟΥ

ΣΧΟΛΗ ΤΕΧΝΟΛΟΓΙΚΩΝ ΕΦΑΡΜΟΓΩΝ ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΠΛΗΡΟΦΟΡΙΚΗΣ Τ.Ε.

«Κωδικοποίηση Καναλιού (Low Density Parity Check Codes - LDPC) για προσομοιωμένα κανάλια κινητών και δορυφορικών επικοινωνιών»



ΠΑΝΟΥ ΣΩΤΗΡΙΟΣ Α.Μ 2007096

ΜΑΥΡΟΕΙΔΟΓΓΟΝΑ ΜΑΡΘΑ Α.Μ 2008096

Επιβλέπων: Δρ ΒΑΣΙΛΕΙΟΣ ΜΠΟΖΑΝΤΖΗΣ

ΣΠΑΡΤΗ ΝΟΕΜΒΡΙΟΣ 2013

ΠΙΝΑΚΑΣ ΠΕΡΙΕΧΟΜΕΝΩΝ

ΚΕΦΑΛΑΙΟ 1^ο : Εισαγωγή	5
1.1 Γενική Περιγραφή.....	5
1.2 Κατηγορίες Μεθόδων Κωδικοποίησης Καναλιού	6
1.3 Οργάνωση της εργασίας.....	7
ΚΕΦΑΛΑΙΟ 2^ο : Τεχνικές Κωδικοποίησης Καναλιού	9
2.1 Γενικά.....	9
2.1.1 Τρόποι διασύνδεσης τερματικών	10
2.2 Αυτόματη αίτηση επανάληψης (ARQ)	11
2.2.1 Παύση και Αναμονή.....	12
2.2.2 Οπισθοδρόμησης κατά N	12
2.3 Κωδικοποίηση Δομημένων Ακολουθιών	13
2.3.1 Κώδικες Ελέγχου Ισοτιμίας	14
2.4 Γραμμικοί Συμπαγείς Κώδικες.....	17
2.4.1 Περιγραφή μέσω γεννήτορα πίνακα G	18
2.4.2 Συστηματικοί Συμπαγείς Κώδικες	20
2.4.3 Περιγραφή μέσω Πίνακα Ελέγχου Ισοτιμίας H.....	22
2.4.4 Έλεγχος Συνδρόμου	24
2.4.5 Διόρθωση σφαλμάτων.....	26
2.5 Ικανότητα Εντοπισμού και Διόρθωσης Λαθών	27
2.5.1 Βάρος και Απόσταση Δυαδικών Διανυσμάτων	27
2.5.2 Ελάχιστη Απόσταση ενός Γραμμικού Κώδικα	28
2.5.3 Εντοπισμός και Διόρθωση Λαθών	29
2.6 Αξιολόγηση Συστημάτων Κωδικοποίησης	31
2.6.1 Χωρητικότητα Διαύλου – Θεώρημα Shannon	31
2.6.2 Κέρδος Κωδικοποίησης	32
2.6.3 Διακριτό Μοντέλο Διαύλου	33
2.6.4 Πιθανότητα λάθους στους γραμμικούς συμπαγείς κώδικες.....	33

2.6.5	Ισοζύγιο επίδοσης-εύρους ζώνης.....	36
2.6.6	Ισοζύγιο ισχύος-εύρους ζώνης.....	36
2.6.7	Ισοζύγιο Ρυθμού Μετάδοσης -εύρους ζώνης.....	36

ΚΕΦΑΛΑΙΟ 3^ο : Προσομοίωση Τεχνικών Κωδικοποίησης για την Εκτίμηση της Επίδοσης τους **38**

3.1	Γενικά περί προσομοίωσης.....	38
◦ 3.2	Μοντελοποίηση.....	41
3.3	Ντετερμινιστικές και Στοχαστικές Προσομοιώσεις.....	42
3.4	Επιλογή της Πλατφόρμας για την Προσομοίωση μας.....	44
3.5	Μοντελοποίηση ενός απλού πομποδέκτη για την εκτίμηση της επίδοσης του.....	45
3.5.1	Πηγή Ψηφιακής Πληροφορίας.....	46
3.5.2	Κωδικοποιητής.....	47
3.5.3	Διαμορφωτής.....	48
3.5.4	Δίαυλος.....	50
3.5.5	Αποδιαμορφωτής.....	57
3.5.6	Αποκωδικοποιητής.....	61
3.5.7	Εκτίμηση Λαθών.....	61

ΚΕΦΑΛΑΙΟ 4^ο : Τεχνικές Κωδικοποίησης LDPC(Low Density Parity Check) **63**

4.1	Γενικά.....	63
4.2	Κωδικοποίηση LDPC.....	64
4.2.1	Περιγραφή LDPC κωδίκων.....	64
4.2.2	Κανονικοί (regural) LDPC κώδικες.....	65
4.2.3	Μη-Κανονικοί (irregural) LDPC κώδικες.....	66
4.3	Αποκωδικοποιητής LDPC.....	67
4.3.1	Ο γράφος Tanner.....	67
4.3.2	Ο αλγόριθμος BP.....	69
4.3.3	Ένα παράδειγμα εκτέλεσης BP αλγορίθμου.....	73

ΚΕΦΑΛΑΙΟ 5^ο : Αποτελέσματα της Προσομοίωσης μας	80
5.1 Μελέτη πάνω στα χαρακτηριστικά του LDPC κώδικα.....	80
5.1.1 Επίδοση του κώδικα για εύρος SNR από 0-4 dB.....	80
5.1.2 Κέρδος από τη χρήση κωδικοποίησης	82
5.1.3 Μελέτη επίδοσης του κώδικα για διάφορες τιμές του μέγιστου αριθμού επαναλήψεων του επαναληπτικού αλγορίθμου αποκωδικοποίησης.....	84
5.2 Επίδοση του LDPC (256, 128) Κώδικα σε σχέση με διάφορους Τύπου Διαύλου και διάφορα Σχήματα Διαμόρφωσης.....	87
5.2.1 Διαμόρφωση 4 – QAM και διάυλος τύπου Rayleigh.....	87
5.2.2 Διαμόρφωση 4 – QAM και διάυλος τύπου Rice.....	89
5.2.3 Διαμόρφωση 4 – QAM και διάυλος τύπου AWGN.....	90
5.2.4 Σύγκριση επίδοσης για διαύλους τύπου AWGN, Rayleigh, Rice.....	90
5.3 Συζήτηση αποτελεσμάτων	93
Παράρτημα – Κώδικας MATLAB	95
Βιβλιογραφία	101

Περίληψη

Η συγκεκριμένη εργασία, σαν στόχο έχει την μελέτη των κωδίκων LDPC (Low Density Parity Check) και την δημιουργία ενός προσομοιωτή για την αποτίμηση της επίδοσής του. Αρχικά, θα πραγματοποιηθεί μια παρουσίαση των μεθόδων κωδικοποίησης καναλιού, καθώς επίσης και των σταδίων της προσομοίωσης. Στη συνέχεια, εξηγούμε τη διαδικασία μοντελοποίησης του τηλεπικοινωνιακού συστήματος το οποίο θα προσομοιωθεί περιγράφοντας λεπτομερώς το στάδιο κωδικοποίησης και αποκωδικοποίησης LDPC. Παράλληλα επισημαίνονται ιδιαίτερα οι τεχνικές διαμόρφωσης και ο τύπος τηλεπικοινωνιακού διαύλου. Στο κεφάλαιο 5 πετυχαίνουμε τον σκοπό της πτυχιακής μας εργασίας, στο οποίο παρουσιάζονται τα αποτελέσματα της προσομοίωσής μας. Σύμφωνα με τα αποτελέσματα που προέκυψαν από τους LDPC κώδικες καταλήγουμε σε ορισμένα επωφελή συμπεράσματα καθώς γίνεται και μελέτη της επίδοσής τους με βάση τις διάφορες τεχνικές διαμόρφωσης και τους διάφορους τύπους διαύλου. Τέλος, συνοψίζεται η συμβολή αυτής εργασίας και είναι άλλωστε κοινά αποδεκτό ότι LDPC κώδικες συνιστούν το παρόν και το μέλλον των τηλεπικοινωνιών.

Λέξεις - κλειδιά: LDPC, Κινητές Επικοινωνίες, Δορυφορικές επικοινωνίες, Κώδικας καναλιού, BER, Προσομοίωση, Matlab.

1 Εισαγωγή

1.1 Γενική Περιγραφή

Ο όρος κωδικοποίηση καναλιού αναφέρεται στις διαδικασίες που γίνονται τόσο στον πομπό, όσο και στον δέκτη, σαν αποκωδικοποίηση, ενός τηλεπικοινωνιακού συστήματος. Οι διαδικασίες αυτές, απαιτούν τη χρήση επιπλέον bits, γνωστά και ως bits ισουμίας και χρησιμοποιούν επιπλέον εύρος ζώνης συχνοτήτων. Σκοπός της είναι η βέλτιστη αντιμετώπιση διαφόρων επιβαρύνσεων που εισάγει ο τηλεπικοινωνιακός διάυλος, όπως είναι ο θόρυβος, οι διαλείψεις και οι παρεμβολές.

Η κωδικοποίηση καναλιού συναντάται αποκλειστικά στα ψηφιακά τηλεπικοινωνιακά συστήματα. Αυτός είναι ένας από τους λόγους που τα ψηφιακά συστήματα υπερτερούν συγκριτικά με τα αναλογικά συστήματα, διότι έχουν τη δυνατότητα να παρέχουν στο τηλεπικοινωνιακό σύστημα αυξημένη ποιότητα. Εάν συσχετίσουμε κάποιες άλλες τεχνικές βελτίωσης της ποιότητας του συστήματος με την κωδικοποίηση καναλιού, όπως για παράδειγμα η χρήση πομποδεκτών μεγαλύτερης ισχύος, παρατηρείται ότι η κωδικοποίηση είναι από τις πιο αποτελεσματικές λύσεις, ειδικά χάρη στη πρόοδο του τομέα των ολοκληρωμένων κυκλωμάτων μεγάλης κλίμακας και των τεχνικών επεξεργασίας ψηφιακών συστημάτων.

Στα επικοινωνιακά συστήματα νέας γενιάς, όπως για παράδειγμα είναι τα δορυφορικά δίκτυα υπολογιστών, στα οποία απαιτείται ένα αρκετά μικρό ποσοστό σφαλμάτων, θεωρείται αναγκαία η κωδικοποίηση καναλιού. Διερεύνηση πραγματοποιείται για το ποια τεχνική κωδικοποίησης θα χρησιμοποιηθεί, εφόσον λάβουμε υπ' όψιν χαρακτηριστικά και τις απαιτήσεις του συστήματος.

Παρακάτω, δίνεται το σχήμα για τη περιγραφή της διαδικασίας της κωδικοποίησης και της αποκωδικοποίησης:

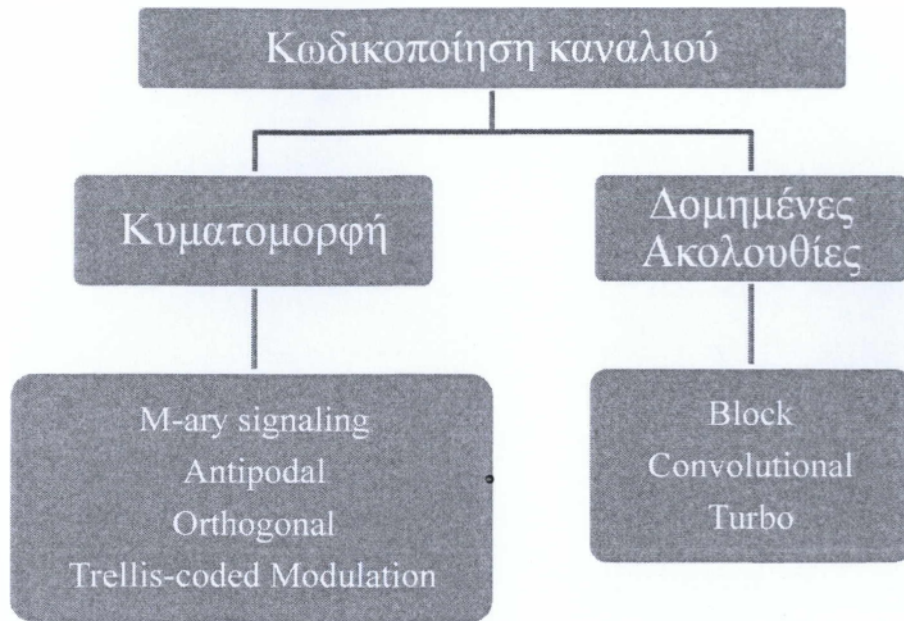


Σχήμα 1.1 Η διαδικασία κωδικοποίησης/αποκωδικοποίησης

Στο συγκεκριμένο σημείο να τονίσουμε πως από εδώ και έπειτα, όταν αναφέρουμε τον όρο κωδικοποίηση, θα αναφερόμαστε αποκλειστικά στην κωδικοποίηση καναλιού και όχι σε άλλου είδους κωδικοποιήσεις, όπως είναι η κωδικοποίηση πηγής.

1.2 Κατηγορίες Μεθόδων Κωδικοποίησης Καναλιού.

Οι κατηγορίες μεθόδων κωδικοποίησης καναλιού κατατάσσονται σε δύο κατηγορίες. Πρώτη κατηγορία είναι οι κωδικοποιήσεις κυματομορφών όπου από τις υπάρχουσες κυματομορφές δημιουργούνται νέες κυματομορφές με σκοπό η εκτίμηση του σήματος στο δέκτη να έχει λιγότερα σφάλματα. Δεύτερη κατηγορία είναι οι κωδικοποιήσεις δομημένων ακολουθιών, όπου γίνεται προσθήκη με δομημένο τρόπο στα αρχικά ψηφία επιπλέον ψηφία από τα οποία θα λάβουμε το κωδικοποιημένο μήνυμα που θα μεταδοθεί από το κανάλι. Αυτά τα επιπλέον ψηφία βοηθούν στην ανίχνευση ή και στη διόρθωση σφαλμάτων που έχουν προστεθεί από το κανάλι στο στάδιο της αποκωδικοποίησης στο δέκτη. Και στις δύο κατηγορίες κοινό στάδιο είναι ότι κατά την κωδικοποίηση παράγονται κωδικοποιημένα σήματα με βελτιωμένα χαρακτηριστικά απόστασης ανάμεσά τους σε σχέση με τα μη-κωδικοποιημένα σήματα. Παρακάτω στο σχήμα 1.2 παρουσιάζονται οι κατηγορίες μεθόδων κωδικοποίησης.



Σχήμα 1.2 Κατάταξη μεθόδων κωδικοποίησης

Σε αυτή την πτυχιακή θα επικεντρωθούμε στην μελέτη κωδικοποίησης με δομημένες ακολουθίες. Επίσης οι LDPC κώδικες που είναι το βασικό στοιχείο της μελέτης μας εμπίπτουν στην υποκατηγορία συμπαγών κωδίκων (βλ. Σχήμα 1.2).

1.3 Οργάνωση της εργασίας

Στην εργασία θα ακολουθηθεί η εξής δομή:

- **Κεφάλαιο 2^ο:** Θα αναφερθούν τα κύρια χαρακτηριστικά των μεθόδων κωδικοποίησης του καναλιού, καθώς επίσης και στη μεθοδική ανάλυση των γραμμικών συμπαγών κωδίκων, αλλά και γενικά στη διαδικασία εκτίμησης των σχημάτων κωδικοποίησης.
- **Κεφάλαιο 3^ο:** Θεωρητική ανάλυση της διαδικασίας της προσομοίωσης. Θα παρουσιαστεί αναλυτικά ο τρόπος με τον οποίο θα γίνει η μοντελοποίηση του τηλεπικοινωνιακού συστήματος που θα προσομοιώσουμε, όπου σαν στόχο έχουμε τη μελέτη της απόδοσης των κωδίκων LDPC.
- **Κεφάλαιο 4^ο:** Λεπτομερής παρουσίαση των κωδίκων LDPC και ανάλυση της πορείας της κωδικοποίησης/αποκωδικοποίησής τους.
- **Κεφάλαιο 5^ο:** Θα πραγματοποιηθεί η παρουσίαση και ο σχολιασμός των αποτελεσμάτων των προσομοιώσεων. Θα δοθεί επιπλέον προσοχή στην εξαγωγή των πορισμάτων σε σχέση με την απόδοση του κώδικα LDPC, σε

συνάρτηση με κάποιους παράγοντες εσωτερικών χαρακτηριστικών του κώδικα καθώς επίσης και με εξωτερικούς παράγοντες, όπως είναι το σχήμα διαμόρφωσης και ο τύπος διαύλου.

2 Τεχνικές Κωδικοποίησης Καναλιού

2.1 Γενικά

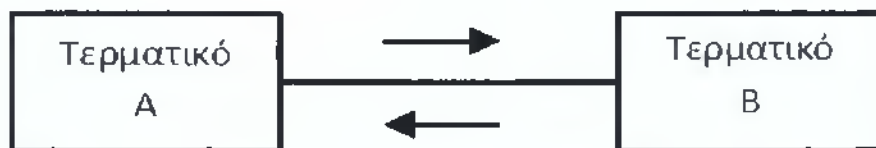
Στο κεφάλαιο αυτό, θα αναφέρουμε τις κύριες ιδιότητες των τεχνικών κωδικοποίησης ενός καναλιού. Στο προηγούμενο κεφάλαιο αναφέραμε ότι η ανάλυσή μας θα εστιάσει στην κωδικοποίηση με δομημένο πλεονασμό. Έχει ήδη ειπωθεί ότι η κωδικοποίηση ενός καναλιού έχει σαν στόχο την καλύτερη διερεύνηση των σφαλμάτων που εισάγονται από τον δίαυλο πάνω στο μεταδιδόμενο σήμα. Σύμφωνα με αυτή τη βασική αρχή, έχουμε δύο κατηγορίες που αφορούν τον τρόπο ανίχνευσης και αντιμετώπισης των σφαλμάτων. Η πρώτη κατηγορία περιλαμβάνει τη διαδικασία κατά την οποία γίνεται η ανίχνευση των λαθών και η επαναμετάδοση (error detection and retransmission). Πιο συγκεκριμένα, έχουμε προσθήκη κάποιων bits ισοτιμίας (parity bits) στα αρχικά bits πληροφορίας, τα οποία bits ισοτιμίας βοηθούν στην στον εντοπισμό πιθανών λαθών. Στην περίπτωση που εντοπισθεί σφάλμα ή και σφάλματα, ο δέκτης στέλνει ξανά στον πομπό τα δεδομένα που έλαβε και περιείχαν λάθος αιτώντας την επαναμετάδοσή τους ενώ δεν προσπαθεί εκείνος να τα διορθώσει. Καθίσταται σαφές από την παραπάνω περιγραφή ότι η συγκεκριμένη μέθοδος «error detection and retransmission», γνωστή και ως ARQ όπως θα τη συναντήσουμε και παρακάτω, απαιτεί αμφίδρομο κανάλι. Η δεύτερη κατηγορία ελέγχου σφαλμάτων που υφίσταται είναι η εμπρόσθια διόρθωση λαθών (forward error correction – FEC). Στην περίπτωση αυτή τα bits ισοτιμίας, δηλαδή τα επιπλέον bits, είναι σχεδιασμένα να διορθώνουν τα πιθανά σφάλματα εκτός από την ανίχνευσή τους. Πρέπει να επισημάνουμε ότι δεν μπορούν να διορθωθούν όλα τα σφάλματα με έναν κώδικα, διότι στην διορθωτική του ικανότητα, όπως θα δούμε παρακάτω, υπάρχουν κάποιοι περιορισμοί. Στα συστήματα που χρησιμοποιούν forward error correction παρατηρούμε ότι δεν αποστέλλονται πληροφορίες από τον δέκτη στον πομπό, επομένως δεν είναι αναγκαία η ύπαρξη ενός καναλιού ανάδρασης, αλλά επαρκεί ένα μονόδρομο κανάλι. Καθότι αρκετές φορές η μεθοδολογία που θα χρησιμοποιηθεί έγκειται στο αν το κανάλι δίνει τη δυνατότητα αμφίδρομης επικοινωνίας ή όχι, έτσι εκτιμάται ότι είναι σκόπιμη η εκτενέστερη παρουσίαση των διαφόρων τύπων καναλιού που σχετίζονται με το συγκεκριμένο θέμα.

2.1.1 Τρόποι διασύνδεσης τερματικών

Εν συνεχεία της προηγούμενης ανάλυσης, θα παρουσιάσουμε τους βασικούς τρόπους διασύνδεσης των τερματικών σταθμών. Όπως διακρίνεται στο Σχήμα 2.1, υπάρχουν τρία είδη καναλιών σε σχέση με την ένωση των δύο τερματικών. Έχουμε το μονόδρομο κανάλι (simplex), το ημιαμφίδρομο (half duplex) και το πλήρως αμφίδρομο (full duplex). Στην περίπτωση του μονόδρομου καναλιού (Σχήμα 2.1a) τα δεδομένα στέλνονται αποκλειστικά από το τερματικό Α προς το Β και ποτέ προς την αντίθετη κατεύθυνση. Στο ημιαμφίδρομο κανάλι (Σχήμα 2.1b) έχουμε δεδομένα τα οποία αποστέλλονται και προς τις δύο κατευθύνσεις, αλλά όχι ταυτόχρονα, ενώ στο πλήρως αμφίδρομο κανάλι έχουμε ταυτόχρονη εκπομπή δεδομένων



α) Μονόδρομο κανάλι



β) Ημι-αμφίδρομο κανάλι



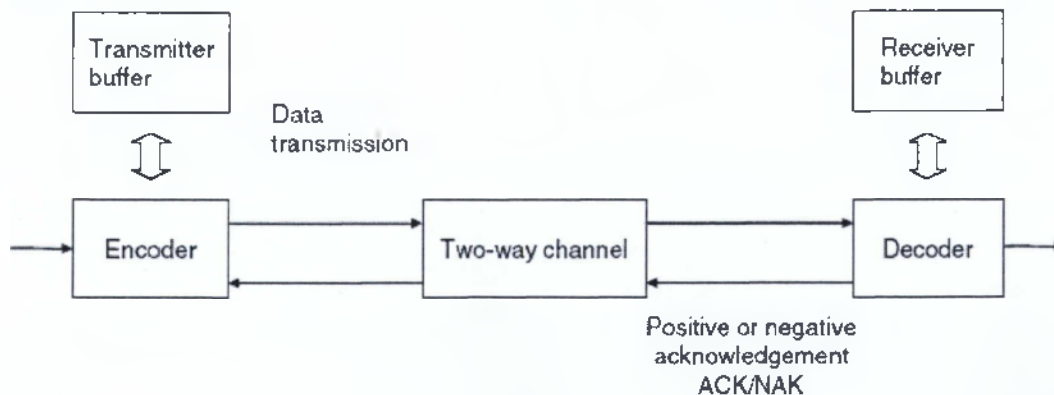
γ) Πλήρως αμφίδρομο κανάλι

Σχήμα 2.1 Τύποι συνδετικότητας.

2.2 Αυτόματη αίτηση επανάληψης (ARQ)

Όταν η διαδικασία ελέγχου σφαλμάτων αποτελείται μόνο από τον εντοπισμό αυτών και όχι και από την διόρθωσή τους, πρέπει ο δέκτης να στείλει στον πομπό μια γνωστοποίηση για το λάθος που εντοπίστηκε και ειδοποίηση ότι αιτείται επαναμετάδοση. Τέτοιου είδους διαδικασίες είναι γνωστές ως μέθοδοι αυτόματης αίτησης επανάληψης (automatic repeat request) ή αυτόματης αίτησης επαναμετάδοσης (automatic retransmission query – ARQ). Στο Σχήμα 2.2 περιγράφεται η λειτουργία των συστημάτων ARQ.

Εν ολίγοις μπορούμε να αναφέρουμε ότι, αφού δημιουργηθεί ένα νέο κωδικοποιημένο διάνυσμα, πρώτα αποθηκεύεται σε ένα buffer και έπειτα μεταδίδεται μέσω του καναλιού. Το αρχικό διάνυσμα είναι πιθανό να αλλοιωθεί από τον θόρυβο που εισάγει το κανάλι. Επομένως, ο αποκωδικοποιητής του δέκτη πραγματοποιεί έλεγχο για να εξακριβώσει εάν το διάνυσμα που έλαβε είναι έγκυρο ή όχι. Αν είναι έγκυρο, θεωρείται από τον αποκωδικοποιητή ότι δεν υπάρχει κάποιο λάθος και στέλνεται στον πομπό μια θετική επιβεβαίωση (ACK). Σε περίπτωση όμως που το διάνυσμα δεν είναι έγκυρο, αυτό σημαίνει ότι κατά τη διάρκεια της εκπομπής εισήχθησαν σφάλματα και στέλνεται στον πομπό ένα σήμα αρνητικής επιβεβαίωσης. Τότε αυτόματα ξεκινάει μια διαδικασία επαναμετάδοσης του κωδικοποιημένου διανύσματος, το οποίο βρίσκεται όπως έχουμε ήδη αναφέρει αποθηκευμένο στο buffer του πομπού.

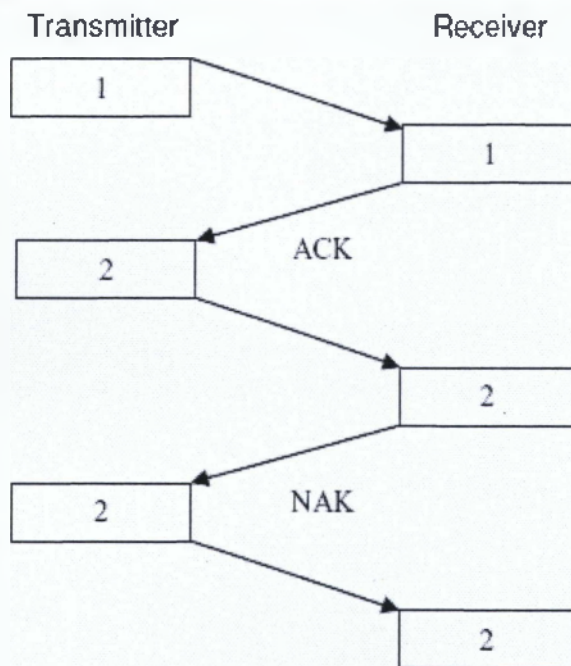


Σχήμα 2.2 Μπλοκ διάγραμμα ARQ συστήματος

Εν συνεχεία θα περιγράψουμε τα πιο διαδεδομένα και χρησιμοποιούμενα πρωτόκολλα τύπου ARQ.

2.2.1 Παύση και Αναμονή

Η ουσιώδης αρχή του σχήματος stop-and-wait είναι ότι για να μεταδώσει ο πομπός ένα νέο πακέτο δεδομένων πρέπει πρώτα να λάβει κάποια απάντηση για το προηγούμενο, είτε θετική αλλά είτε και αρνητική. Βέβαια, αν υποθέσουμε ότι ελήφθη NAK πραγματοποιείται επαναμετάδοση, διαφορετικά η λήψη ενός ACK οδηγεί στην μετάδοση του επόμενου πακέτου που έπεται. Στο Σχήμα 2.3 φαίνονται καθαρά οι δύο ενδεχόμενες περιπτώσεις τις οποίες προαναφέραμε. Στο πρωτόκολλο αυτό, πραγματοποιείται αποθήκευση ενός μόνο πακέτου δεδομένων κάθε φορά στον buffer. Αυτό σημαίνει ότι ο αναλώσιμος χώρος αποθήκευσης είναι ο ελάχιστος δυνατός. Από την άλλη μεριά όμως, ο χρόνος αναμονής του πομπού D είναι πιθανό να είναι αισθητά μεγάλος. Σε οποιαδήποτε περίπτωση, όσον αφορά τον χρόνο αναμονής D ισχύει $D \geq 2T_d$, όπου T_d είναι η καθυστέρηση μετάδοσης από άκρη σε άκρη.

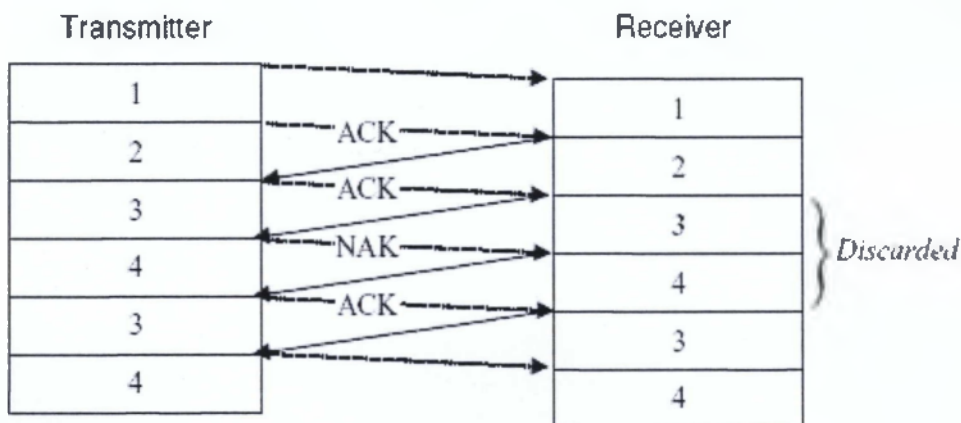


Σχήμα 2.3 Stop and wait ARQ πρωτόκολλο

2.2.2 Οπισθοδρόμησης Κατά N

Στο πρωτόκολλο οπισθοδρόμησης κατά N (Go-back N) το οποίο είναι τύπου ARQ υπάρχει διαρκής μετάδοση πακέτων από τον πομπό, εν αντιθέσει με το πρωτόκολλο παύσης και αναμονής (Stop – and- Wait) που συναντήσαμε πριν. Στην περίπτωση που ληφθεί NAK, ο πομπός γυρνάει πίσω στο αντίστοιχο πακέτο και

αρχίζει να επανεκπέμπει από εκείνο το σημείο. Για να γίνει κάτι τέτοιο είναι αναγκαίο να υπάρχει η δυνατότητα αποθήκευσης στον buffer του πομπού N πακέτων, όπου το N ορίζεται από την καθυστέρηση πλήρους διαδρομής (round-trip delay) του συστήματος. Τώρα, όσον αφορά τον δέκτη, παραβλέπονται όλα τα υπόλοιπα $N-1$ πακέτα που λαμβάνονται μετά το λανθασμένο πακέτο, ακόμα και αν αυτά δεν περιέχουν σφάλματα, με σκοπό να συνεχίσει να υπάρχει η σωστή διαδοχή των πακέτων. Επομένως, ο αποθηκευτικός χώρος στον buffer του δέκτη είναι αρκετός εάν είναι ίσος με ένα πακέτο. Στο Σχήμα 2.4 περιγράφεται η διαδικασία για $N = 2$.



Σχήμα 2.4 Παράδειγμα Go-Back N πρωτοκόλλου ($N=2$)

2.3 Κωδικοποίηση Δομημένων Ακολουθιών

Από εδώ και έπειτα θα μας απασχολήσουν οι κώδικες ελέγχου ισοτιμίας. Αυτοί οι κώδικες αναφέρονται ως κώδικες δομημένων ακολουθιών, διότι κατά τη διαδικασία της κωδικοποίησης προστίθενται με συγκεκριμένο τρόπο επιπρόσθετα δεδομένα στα αρχικά, για να γίνει εφικτή η ανίχνευση ή η αποκατάσταση σφαλμάτων. Είδαμε, στο Σχήμα 1.3 ότι οι κώδικες με δομημένες ακολουθίες διαχωρίζονται σε τρεις κατηγορίες. Αυτές οι κατηγορίες είναι οι εξής: οι συμπαγείς κώδικες (block codes), οι συνελκτικοί (convolutional) και οι turbo κώδικες. Εμείς θα επιμείνουμε περισσότερο στην περιγραφή των συμπαγών κωδίκων, εφόσον είναι το βασικό θέμα μελέτης μας, που είναι οι κώδικες LDPC, οι οποίοι ανήκουν σε αυτή τη κατηγορία.

2.3.1 Κώδικες Ελέγχου Ισοτιμίας

Πριν συνεχίσουμε με επιπλέον διερεύνηση, είναι απαραίτητο να εισάγουμε την ερμηνεία του ρυθμού κωδικοποίησης, η οποία είναι απαραίτητη για κάθε σχήμα κωδικοποίησης δομημένων ακολουθιών. Συνεπώς, κατά τη διάρκεια της κωδικοποίησης, ο κωδικοποιητής δέχεται σαν είσοδο k bits δεδομένων. Στην έξοδο δίνει n bits, για τα οποία ισχύει $n > k$. Τα n bits τα ορίζουμε ως κωδικοποιημένα bits ή σύμβολα. Τα $(n-k)$ bits που προσάρτησε ο κωδικοποιητής είναι τα επιπλέον bits τα οποία και δεν αποτελούνται από νέα ποσότητα πληροφορίας, όμως συνυπολογίζονται για την ανίχνευση ή την αποκατάσταση των σφαλμάτων. Ένας κώδικας αυτού του είδους αναπαρίσταται ως (n,k) κώδικας. Ο ρυθμός κωδικοποίησης ορίζεται ως ο λόγος των bits της πληροφορίας προς τα συνολικά bits, δηλαδή διατυπώνεται από τον λόγο k/n .

2.3.1.1 Κώδικας Μονού Ελέγχου Ισοτιμίας

Ο κώδικας ελέγχου ισοτιμίας (parity-check code) για να παράγει bit ισοτιμίας (parity bits) προκειμένου να ανιχνεύσει και να διορθώσει σφάλματα, χρησιμοποιεί γραμμικά άθροισμα των bits πληροφορίας. Στην περίπτωση ενός κώδικα μονού ελέγχου ισοτιμίας, πρέπει να προστεθεί μόνο ένα bit ισοτιμίας σε κάθε πακέτο που περιλαμβάνει bits. Στο bit ισοτιμίας μπορεί να οριστεί η τιμή 1 ή 0 ούτως ώστε το άθροισμα modulo-2 να είναι περιττό ή άρτιο, σύμφωνα με τον σχεδιασμό του πίνακα. Στην περίπτωση που βγει άρτιο το άθροισμα έχουμε κώδικες άρτιας ισοτιμίας, ενώ αντίστοιχα έχουμε κώδικες περιττής ισοτιμίας. Η άθροιση modulo-2 ορίζεται ως εξής:

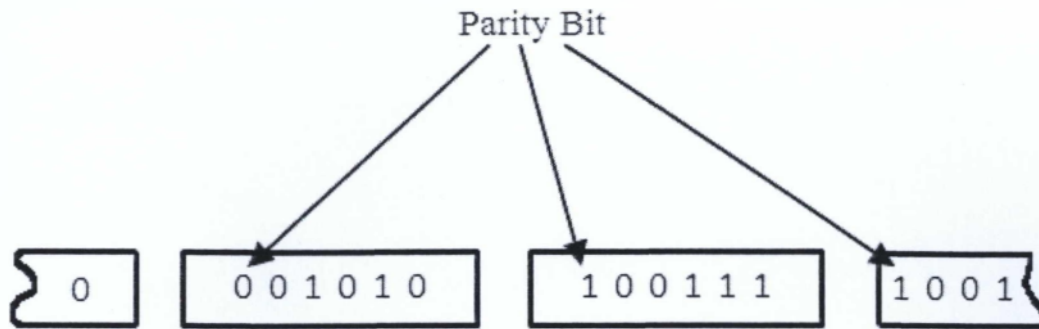
$$0 \oplus 0 = 0$$

$$0 \oplus 1 = 1$$

$$1 \oplus 0 = 1$$

$$1 \oplus 1 = 0$$

Παρακάτω, στο Σχήμα 2.5 διακρίνεται ο τρόπος με τον οποίο το bit ισοτιμίας προστίθεται μέσα σε κάθε μπλοκ, προκειμένου να διατηρηθεί άρτια ισοτιμία.



Σχήμα 2.5 Περίπτωση κώδικα άρτιας ισοτιμίας

Τώρα, από την μεριά του δέκτη, ο αποκωδικοποιητής θα ελέγξει εάν το σύνολο των bits για κάθε κωδική λέξη που έλαβε, καλύπτει τις προϋποθέσεις του συστήματος για άρτια ισοτιμία. Εάν για μια οποιαδήποτε κωδική λέξη το άθροισμά της βγει 0 τότε είναι μια λέξη η οποία δεν περιέχει λάθη, αν όμως βγει 1, τότε η λέξη είναι λανθασμένη. Από τα παραπάνω συμπεραίνεται ότι αυτού του είδους ο κώδικα δεν μπορεί να αποκαταστήσει τα σφάλματα, επειδή δεν υπάρχουν επιπλέον πληροφορίες για το που ακριβώς βρίσκονται. Μάλιστα η ύπαρξη μονού αριθμού σφαλμάτων για κάθε κωδική λέξη είναι το μοναδικό που μπορεί να εντοπιστεί. Αν όμως προκύψει ζυγός αριθμός σφαλμάτων, υπάρχει η πιθανότητα το σφάλμα να μην είναι εντοπισμένο. Για παράδειγμα, αν η κωδική λέξη 100100 καταλήξει στον αποκωδικοποιητή ως 010001, το σφάλμα δεν εντοπίζεται διότι ισχύει η συνθήκη της άρτιας ισοτιμίας. Όσον αφορά την μετάδοση των bits, θεωρούμε ότι τα bits μεταδίδονται με τυχαίο τρόπο και συμβολίζουμε με p την πιθανότητα ένα bit να ληφθεί εσφαλμένο. Για την πιθανότητα να ληφθούν από ένα πακέτο των n bits, j εσφαλμένα bits, ισχύει η κάτωθι περίπτωση:

$$P(j, n) = \binom{n}{j} p^j (1-p)^{n-j} \quad (2.1)$$

όπου

$$\binom{n}{j} = \frac{n!}{j!(n-j)!}$$

Από την σχέση (2.1) προκύπτει ότι για κώδικες μονού ελέγχου ισοτιμίας, η πιθανότητα μη εντοπισμού σφάλματος δίνεται από τη σχέση

$$P_{nd} = \sum_{j=1}^{(n-1)/2} \binom{n}{2j} p^{2j} (1-p)^{n-2j} \quad (2.2)$$

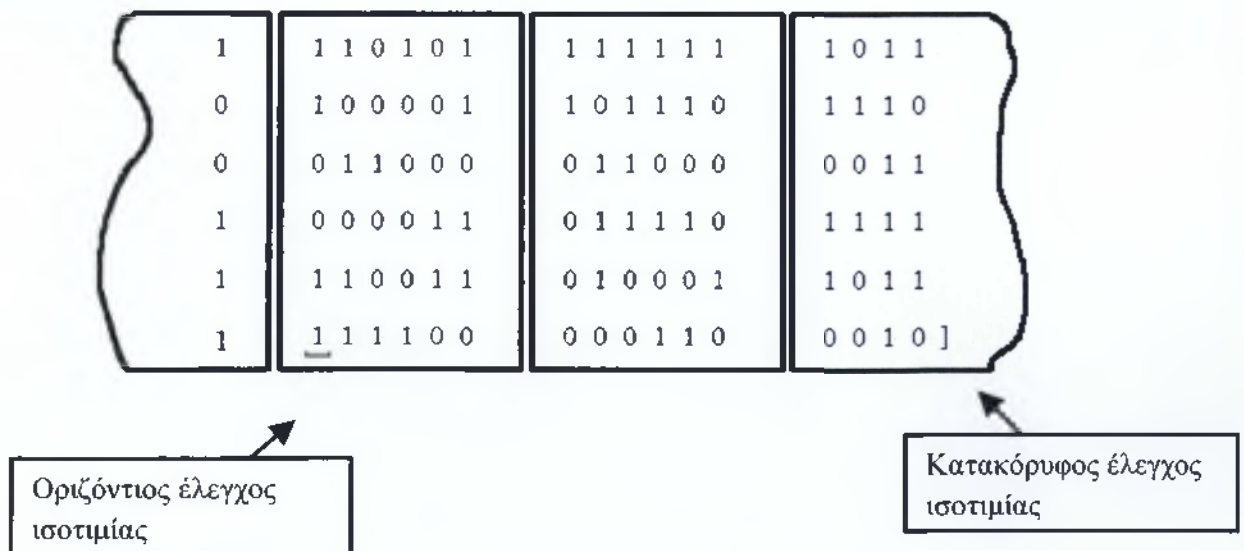
για n περιττό και

$$P_{nd} = \sum_{j=1}^{n/2} \binom{n}{2j} p^{2j} (1-p)^{n-2j} \quad (2.3)$$

για n άρτιο.

2.3.1.2 Ορθογώνιοι κώδικες

Η δισδιάστατη παραλλαγή των κωδίκων που περιγράφηκε είναι οι ορθογώνιοι κώδικες. Για να δημιουργηθούν αυτοί οι κώδικες απαιτούνται ορισμένα βήματα. Στην αρχή, τα bits πληροφορίας διαμορφώνονται σε ορθογώνια με διαστάσεις M x N. Έπειτα, ασκείται έλεγχος ισοτιμίας σε κάθε στήλη και γραμμή ξεχωριστά του πίνακα M x N που δημιουργήθηκε, κατασκευάζοντας με αυτόν τον τρόπο ένα κωδικοποιημένο πίνακα διαστάσεων (M+1) x (N+1) και κάνοντας το ρυθμό κώδικα να ισούται με MN / ((M+1) x (N+1)). Παρακάτω, έχουμε την διαδικασία με την οποία δημιουργείται ένας τέτοιος κώδικας, για M = N = 5, όπου έχουμε άρτια ισοτιμία.



Σχήμα 2.6 Παράδειγμα ορθογώνιου κώδικα ελέγχου ισοτιμίας

Στο παράδειγμα μας, σε περίπτωση ενός μόνο σφάλματος σε ένα από τα 36 bits του κωδικοποιημένου πίνακα θα προσκληθεί αποτυχία ισοτιμίας σε μία μόνο γραμμή και σε μία μόνο στήλη του πίνακα. Αποτυχία όμως μόνο στη γραμμή i και μόνο στη στήλη j σημαίνει ότι το bit με σφάλμα βρίσκεται στη θέση (i, j) . Αυτός ο ακριβής εντοπισμός της θέσης του μοναδικού σφάλματος σημαίνει και διορθωτική ικανότητα του κώδικα και είναι αυτό που τον ξεχωρίζει από τους μέχρι τώρα αναλυθέντες κώδικες. Όπως βλέπουμε όμως, η διορθωτική ικανότητα τέτοιων κωδικών περιορίζεται μόνο στην περίπτωση ενός σφάλματος ανά κωδική λέξη. Δεν παύει όμως αυτός ο κώδικας να αποτελεί την εισαγωγή μας στην κύρια κατηγορία που θα μελετηθούν: τους κώδικες διόρθωσης λαθών.

2.4 Γραμμικοί Συμπαγείς Κώδικες

Τα δεδομένα τα οποία περιλαμβάνουν οι γραμμικοί συμπαγείς κώδικες, οργανώνονται σε διανύσματα με σταθερό μήκος k bits. Με \mathbf{m} συμβολίζουμε το διάνυσμα που μεταφέρει κάθε φορά τα δεδομένα και ονομάζεται διάνυσμα μηνύματος ή διάνυσμα πληροφορίας. Επομένως, για $\mathbf{m}=(\mathbf{m}_0, \dots, \mathbf{m}_{k-1})$ θεωρούμε πως τα στοιχεία του \mathbf{m} είναι 1 ή 0, όπου και ανήκει στο δυαδικό σύστημα. Σ'αυτή τη περίπτωση λοιπόν, το διάνυσμα k μπορεί να πάρει 2^k διαφορετικές μορφές. Η διαμόρφωση του \mathbf{m} σε ένα καινούριο διάνυσμα $\mathbf{c}=(\mathbf{c}_0, \dots, \mathbf{c}_{n-1})$ με μήκος $n > k$, προέρχεται από τη διεργασία της κωδικοποίησης, επομένως ο γραμμικός συμπαγής κώδικας αναπαρίσταται ως (n, k) .

Όποτε οι τιμές των k, n είναι σχετικά χαμηλές, η προαναφερθείσα διαμόρφωση μπορεί να υλοποιηθεί μέσα από έναν ενδεικνυόμενο πίνακα, ο οποίος θα αντιστοιχίζει το καθένα από τα 2^k διανύσματα πληροφορίας ξεχωριστά σε μια κωδική λέξη μήκους n . Αν όμως τα k και n είναι μεγάλα, είναι απρόσιτες οι διαστάσεις ενός τέτοιου πίνακα, άρα για την διαδικασία της κωδικοποίησης είναι αναγκαία η δημιουργία ενός γεννήτορα μηχανισμού. Γνωρίζοντας αυτή τη δυσκολία, η παρουσία της γραμμικότητας στον συγκεκριμένο μηχανισμό, χρησιμεύει στην απλούστευση της διαδικασίας.

Ορισμός 2.1: Ένας συμπαγής κώδικας με μήκος n και 2^k πιθανά μηνύματα δεδομένων, λέγεται γραμμικός συμπαγής κώδικας (n, k) εφόσον οι 2^k κωδικές λέξεις

του σχηματίζουν ένα διανυσματικό υπόχωρο, διάστασης k , του διανυσματικού χώρου V_n που περιλαμβάνει όλα τα διανύσματα μήκους n , με στοιχεία στο δυαδικό χώρο.

Ως επακόλουθο του ανωτέρω ορισμού, ένας γραμμικός συμπαγής κώδικας, περιγράφεται από το ότι σε κάθε περίπτωση μία από τις κωδικές λέξεις είναι μηδενικό διάνυσμα, καθώς επίσης το άθροισμα δύο τυχαίων κωδικών λέξεων είναι και αυτό κωδική λέξη.

2.4.1 Περιγραφή μέσω γεννήτορα πίνακα G

Εφόσον ένας γραμμικός συμπαγής κώδικας (n, k) συνιστά ένα διανυσματικό υπόχωρο του V_n , τότε θα υπάρχουν k γραμμικά ανεξάρτητα διανύσματα, ούτως ώστε οποιαδήποτε ενδεχόμενη κωδική λέξη να απορρέει σαν γραμμική συσχέτιση αυτών:

$$c = m_0 \bullet g_0 \oplus m_1 \bullet g_1 \oplus \dots \oplus m_{k-1} \bullet g_{k-1} \quad (2.4)$$

Τα k διανύσματα g_0, \dots, g_{k-1} , συνιστούν τον θεμέλιο λίθο του υποχώρου και σχετικά με την κωδικοποίηση κάθε ένα από αυτά είναι κωδική λέξη. Σε μορφή πίνακα μπορούν να συνταχθούν τα γραμμικώς ανεξάρτητα διανύσματα, η οποία είναι η εξής:

$$G = \begin{bmatrix} g_0 \\ g_1 \\ \vdots \\ g_{k-1} \end{bmatrix} = \begin{bmatrix} g_{00} & g_{01} & \dots & g_{0,n-1} \\ g_{10} & g_{11} & \dots & g_{1,n-1} \\ \vdots & \vdots & \dots & \vdots \\ g_{k-1,0} & g_{k-1,1} & \dots & g_{k-1,n-1} \end{bmatrix}$$

Τον πίνακα G τον ονομάζουμε γεννήτορα πίνακα, διότι είναι η κύρια λειτουργία για την δημιουργία κωδικών λέξεων. Ως επακόλουθο των ανωτέρω, ο πίνακας G είναι ο γραμμικός συνδυασμός των γραμμών, ο οποίος δημιουργεί την κάθε λέξη που συνάδει με τον κώδικα. Αυτός ο πίνακας έχει n στήλες και k γραμμές. Για ένα συγκεκριμένο διάνυσμα δεδομένων, το $m = (m_0, \dots, m_{k-1})$, η ανάλογη κωδική λέξη λαμβάνεται μέσω του πολλαπλασιασμού των πινάκων όπως θα δούμε και παρακάτω :

$$c = m \circ G = (m_0, m_1, \dots, m_{k-1}) \circ \begin{bmatrix} g_{0,0} & g_{0,1} & \dots & g_{0,p-1} \\ g_{1,0} & g_{1,1} & \dots & g_{1,p-1} \\ \vdots & \vdots & \ddots & \vdots \\ g_{k-1,0} & g_{k-1,1} & \dots & g_{k-1,p-1} \end{bmatrix} = (m_0, m_1, \dots, m_{k-1}) \circ \begin{bmatrix} g_0 \\ g_1 \\ \vdots \\ g_{k-1} \end{bmatrix} = m_0 \bullet g_0 \oplus m_1 \bullet g_1 \oplus \dots \oplus m_{k-1} \bullet g_{k-1}$$

Στην διαδικασία που προηγήθηκε, το σύμβολο \circ ορίζει εσωτερικό γινόμενο μεταξύ διανυσμάτων και το σύμβολο \bullet ορίζει τον πολλαπλασιασμό ενός μονόμετρου μεγέθους με ένα διάνυσμα.

Παράδειγμα 2.1: Έστω ότι έχουμε τον κάτωθι γεννήτορα πίνακα 3 x 5,

$$G = \begin{bmatrix} 0 & 1 & 1 & 0 & 1 \\ 1 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 1 \end{bmatrix}$$

Επομένως, αν στην είσοδο του κωδικοποιητή έχουμε το μήνυμα $m=(0 \ 1 \ 0)$ στην έξοδο θα πάρουμε την αντίστοιχη κωδική λέξη c ως εξής:

$$c = [0 \ 1 \ 0] \circ \begin{bmatrix} 0 & 1 & 1 & 0 & 1 \\ 1 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 1 \end{bmatrix} = [0 \cdot 0 \oplus 1 \cdot 1 \oplus 0 \cdot 0 \ 0 \cdot 1 \oplus 1 \cdot 0 \oplus 0 \cdot 1 \ 0 \cdot 1 \oplus 1 \cdot 0 \oplus 0 \cdot 0 \ 0 \cdot 0 \oplus 1 \cdot 1 \oplus 0 \cdot 0 \ 0 \cdot 1 \oplus 1 \cdot 0 \oplus 0 \cdot 1]$$

$$= [1 \ 0 \ 0 \ 1 \ 0]$$

Με τον ίδιο τρόπο προκύπτει ο παρακάτω πίνακας

Πίνακας 2.1 Κωδικές λέξεις του (5, 3) κώδικα του παραδείγματος 2.1

Μήνυμα Πληροφορίας	Κωδική Λέξη
000	00000
001	01001
010	10010
011	11011
100	01101
101	00100
110	11111
111	10110

2.4.2 Συστηματικοί Συμπαγείς Κώδικες

Συστηματικός συμπαγής κώδικας είναι ο γραμμικός συμπαγής κώδικας του οποίου τα πρώτα k ψηφία ταυτίζονται με την ακωδικοποίητη πληροφορία, ενώ τα ακόλουθα $(n-k)$ ψηφία προστίθενται από τον αποκωδικοποιητή. Η διάρθρωση αυτή διακρίνεται στο Σχήμα 2.7



Σχήμα 2.7 Συστηματική μορφή κωδικής λέξης ενός συμπαγούς κώδικα

Ο αποκωδικοποιητής εισάγει ψηφία (bits), τα οποία αποκαλούνται ψηφία ελέγχου ισοτιμίας.

Στην βιβλιογραφία βλέπουμε και τον παρεμφερή ορισμό του συστηματικού κώδικα σε σχέση με αυτόν που αναφέρθηκε παραπάνω. Συγκεκριμένα, συστηματικός κώδικας θεωρείται ο κώδικας του οποίου τα πρώτα k bits ταυτίζονται με την ακωδικοποίητη πληροφορία σε αντίθεση με τα επόμενα $(n-k)$ ψηφία τα οποία προστίθενται από τον αποκωδικοποιητή. Μια τέτοια περίπτωση συστηματικού κώδικα είναι το παράδειγμα 2.1. Από εδώ και στο εξής, όπου συναντάται ο ορισμός του συστηματικού, θα αναφερόμαστε στη διάρθρωση που έχει το Σχήμα 2.7.

Ένας γεννήτορας πίνακας με την παρακάτω μορφή, ορίζει μονοσήμαντα έναν συστηματικό γραμμικό συμπαγή κώδικα.

$$G = \begin{bmatrix} g_0 \\ g_1 \\ \vdots \\ g_{k-1} \end{bmatrix} = \begin{bmatrix} p_{00} & p_{01} & \dots & p_{0,n-k-1} & 1 & 0 & 0 & \dots & 0 \\ p_{10} & p_{11} & \dots & p_{1,n-k-1} & 0 & 1 & 0 & \dots & 0 \\ \vdots & \vdots & & \vdots & \vdots & \vdots & \vdots & & \vdots \\ p_{k-1,0} & p_{k-1,1} & \dots & p_{k-1,n-k-1} & 0 & 0 & 0 & \dots & 0 \end{bmatrix}$$

Υποπίνακα P $k \times (n-k)$

Υποπίνακα I $k \times k$

ο οποίος μπορεί να γραφτεί περιληπτικά και σαν

$$G = [P \ I_k] \quad (2.5)$$

το οποίο I_k , ορίζει έναν μοναδιαίο πίνακα $k \times k$.

Η δομή που έχουν οι συστηματικοί κώδικες, μας επιτρέπουν να συμπεράνουμε λεπτομερείς εκφράσεις για την αλληλουχία των bits ελέγχου ισοτιμίας (parity check

bits) με τα bits ακωδικοποίητης πληροφορίας (message bits). Στην περίπτωση που το ακωδικοποίητο διάνυσμα πληροφορίας είναι $\mathbf{m}=(\mathbf{m}_0, \mathbf{m}_1, \dots, \mathbf{m}_{k-1})$ και το ανάλογο κωδικοποιημένο είναι $\mathbf{c}=(\mathbf{c}_0, \mathbf{c}_1, \dots, \mathbf{c}_{n-1})$ ισχύει:

$$c_{n-k+i} = m_i, 0 \leq i \leq k-1$$

$$c_j = m_0 \cdot p_{0j} + m_1 \cdot p_{1j} + \dots + m_{k-1} \cdot p_{k-1,j}, 0 \leq j \leq n-k \quad (2.6)$$

Οι εξισώσεις ελέγχου ισοτιμίας ορίζονται από τις εξισώσεις (n-k) που βρίσκονται στο δεύτερο μέρος της (2.6).

Παράδειγμα 2.2 Έστω ότι έχουμε τον παρακάτω γεννήτορα πίνακα του γραμμικού συστηματικού συμπαγούς κώδικα (8, 4).

$$G = \begin{bmatrix} 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 1 & 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 1 & 0 & 0 & 1 & 0 \\ 1 & 1 & 1 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

Ο πίνακας G έχει τη μορφή $G = [P \ I_k]$ με $k=4$. Η κάθε κωδική λέξη προκύπτει από τη βασική σχέση

$$c = m \circ G \quad (2.7)$$

Στο παράδειγμα μας,

$$c = m \circ G = (m_0, m_1, m_2, m_3) \circ \begin{bmatrix} 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 1 & 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 1 & 0 & 0 & 1 & 0 \\ 1 & 1 & 1 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

Επομένως οι εξισώσεις ελέγχου ισοτιμίας είναι:

$$c_0 = m_1 \oplus m_2 \oplus m_3$$

$$c_1 = m_0 \oplus m_1 \oplus m_3$$

$$c_2 = m_0 \oplus m_2 \oplus m_3$$

$$c_3 = m_0 \oplus m_1 \oplus m_2$$

ενώ για τα υπόλοιπα bits της κωδικής λέξης ισχύει:

$$\begin{aligned}
c_4 &= m_0 \\
c_5 &= m_1 \\
c_6 &= m_2 \\
c_7 &= m_3
\end{aligned}$$

2.4.3 Περιγραφή μέσω Πίνακα Ελέγχου Ισοτιμίας H

Μία διαφορετική αναπαράσταση των γραμμικών συμπαγών κωδίκων μπορεί να γίνει μέσω του πίνακα ελέγχου ισοτιμίας H . Από τον κώδικα (n, k) , προκύπτει ότι ο πίνακας έχει διαστάσεις $(n-k) \times n$. Η βασική ιδιότητα του H είναι ότι δημιουργείται από τον γεννήτορα πίνακα του κώδικα G , βάσει της κάτωθι σχέσης:

$$G \circ H^T = 0 \quad (2.8)$$

όπου με H^T ορίζεται ο ανάστροφος πίνακας του H .

Από την (2.8) απορρέει η συστηματική μορφή του H με δεδομένο τον συστηματικό πίνακα G (2.5).

$$H = \left[\begin{array}{cccc|cccc}
1 & 0 & \cdots & 0 & p_{00} & p_{10} & \cdots & p_{k-1,0} \\
0 & 1 & \cdots & 0 & p_{01} & p_{11} & \cdots & p_{k-1,1} \\
\vdots & \vdots & & \vdots & \vdots & \vdots & & \vdots \\
0 & 0 & \cdots & 1 & p_{0,n-k-1} & p_{1,n-k-1} & \cdots & p_{k-1,n-k-1}
\end{array} \right] = [I_{n-k} \quad P^T]$$

Υποπίνακα $I(n-k) \quad (n-k)$
Υποπίνακα $P^T(n-k) \times k$

Σχήμα 2.8 Γενική συστηματική μορφή του πίνακα ελέγχου ισοτιμίας

Επιπλέον μας δίνεται η δυνατότητα να δημιουργήσουμε την ακόλουθη σχέση, η οποία ισχύει για κάθε κωδική λέξη c .

$$c \circ H^T = m \circ G \circ H^T = 0 \quad (2.9)$$

Η παραπάνω εξίσωση (2.9), είναι σχέση ελέγχου αξιοπιστίας μιας κωδικής λέξης. Επομένως, αν ο αποκωδικοποιητής λάβει την κωδική λέξη r , πραγματοποιείται έλεγχος για το αν ισχύει η παρακάτω σχέση:

$$r \circ H^T = 0$$

Εάν ισχύει η παραπάνω σχέση, το διάνυσμα \mathbf{r} θεωρείται ως έγκυρη κωδική λέξη, διαφορετικά είναι μη-έγκυρη (πιθανόν να διαστρεβλώθηκε κατά τη μετάδοσή του).

Παράδειγμα 2.3 Καθορισμός πίνακα ελέγχου ισοτιμίας δεδομένου του γεννήτορα πίνακα. Έστω λοιπόν ότι έχουμε τον συστηματικό γεννήτορα πίνακα 4×8 , ίδιο με το προηγούμενο παράδειγμα, ως ακολούθως:

$$G = \begin{bmatrix} 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 1 & 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 1 & 0 & 0 & 1 & 0 \\ 1 & 1 & 1 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

Ανατρέχοντας στην μορφή (2.5) ο G συντίθεται από τον υποπίνακα

$$I = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

και από το μοναδιαίο υποπίνακα

$$I = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Από τους παρακάτω υποπίνακες και από το σχήμα 2.4 προκύπτει ο \mathbf{H}

$$H = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 1 & 1 & 1 \\ 0 & 1 & 0 & 0 & 1 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 1 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 \end{bmatrix}$$

Επιβεβαιώνεται ότι οι διαστάσεις του πίνακα ελέγχου ισοτιμίας είναι $(n-k) \times n = (8-4) \times 8 = 4 \times 8$.

2.4.4 Έλεγχος Συνδρόμου

Εάν έχουμε έναν κώδικα (n,k) , η λέξη η οποία θα μεταδοθεί έχει τη μορφή $c=(c_0, c_1, \dots, c_{n-1})$. Επίσης, θεωρούμε πως το διάνυσμα το οποίο αναπαριστά τη λέξη που ελήφθη ορίζεται με r και ισχύει $r=(r_0, r_1, \dots, r_{n-1})$. Πιθανά λάθη τα οποία προέκυψαν κατά την εκπομπή της κωδικής λέξης, είναι δυνατό να μοντελοποιηθούν μέσω του διανύσματος λαθών e ως εξής :

$$e = r \oplus c \quad (2.10)$$

Στο διάνυσμα e , περιέχονται μη μηδενικά στοιχεία, αποκλειστικά στις θέσεις i του διανύσματος που ελήφθη και έχει υποστεί κάποια αλλοίωση ($r_i \neq c_i$).

Από την σχέση (2.10) ορίζεται ισοδύναμα

$$c = r \oplus e \text{ και } r = c \oplus e \quad (2.11)$$

Ο αποκωδικοποιητής μπορεί να χρησιμοποιήσει τη σχέση $c = r \oplus e$ για να επανορθώσει το διάνυσμα r το οποίο περιλαμβάνει τα λάθη που αναλογούν στο διάνυσμα λάθους e . Όπως μπορούμε εύκολα να καταλάβουμε, είναι αρκετά δύσκολος ο ορισμός του e .

Προς αυτή την κατεύθυνση, ορίζεται το r ως εξής:

$$S = r \circ H^T \quad (2.12)$$

Από την (2.9) συμπεραίνεται ότι σε περίπτωση που το διάνυσμα r είναι μία κωδική λέξη, τότε το μηδενικό διάνυσμα είναι το σύνδρομο S . Στην αντίθετη περίπτωση, αν το διάνυσμα που λήφθηκε περιέχει λάθη, τότε θα περιέχονται μερικά μη-μηδενικά στοιχεία στο σύνδρομο. Σε τέτοια περίπτωση, ο αποκωδικοποιητής έχει δύο επιλογές. Η πρώτη είναι να συνεχίσει με την αυτόματη διόρθωση λαθών (σύστημα FEC) ή να απαιτήσει την επαναμετάδοση του μηνύματος (σύστημα ARQ).

Εφόσον έχουμε $r = c \oplus e$, η (2.12) μπορεί να διαμορφωθεί ως εξής:

$$S = r \circ H^T = (c \oplus e) \circ H^T = c \circ H^T \oplus e \circ H^T \quad (2.13)$$

Μέσω της (2.9), η (2.13) γίνεται:

$$S = e \circ H^T \quad (2.14)$$

Η προηγούμενη ανάλυση επισήμανε ότι το σύνδρομο που ελήφθη με σφάλμα, δεν διαφέρει από το σύνδρομο του διανύσματος σφάλματός του. Οι γραμμικοί συμπαγείς κώδικες έχουν μία βασική ιδιότητα, απαραίτητη για την διεξαγωγή της κωδικοποίησης. Αυτή είναι ότι η αναλογία είναι ένα προς ένα μεταξύ των διανυσμάτων σφάλματος που μπορούν να διορθωθούν και των συνδρόμων.

Ο πίνακας ελέγχου ισοτιμίας H έχει δύο κύριες ιδιότητες:

- Στον πίνακα ελέγχου ισοτιμίας, δεν πρέπει να περιέχεται ουδεμία στήλη που να περιέχει αποκλειστικά μηδενικά στοιχεία, διότι σε αυτό το ενδεχόμενο υπάρχει η περίπτωση πιθανού λάθους στην ανάλογη θέση της λέξης που περιλαμβάνεται γιατί θα υπήρχε επίδραση στο σύνδρομο. Αυτή η ιδιότητα στηρίζεται στην σχέση (2.14).
- Οι στήλες που περιέχονται στον πίνακα ελέγχου ισοτιμίας επιβάλλεται να είναι μοναδικές. Εάν υπάρχουν δύο όμοιες στήλες, θα υπάρχουν σφάλματα στις δύο ανάλογες θέσεις της κωδικής λέξης, τα οποία είναι μη-μηδενικά.

Παράδειγμα 2.4 Έστω ότι έχουμε τον γραμμικό συμπαγή κώδικα (3, 6) με πίνακα ελέγχου ισοτιμίας

$$H = \begin{bmatrix} 1 & 0 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 & 1 \end{bmatrix}$$

Αν μεταδόθηκε η κωδικοποιημένη λέξη $c = (1\ 0\ 1\ 1\ 1\ 0)$ και ελήφθη η λέξη $r = (0\ 0\ 1\ 1\ 1\ 0)$, τότε το σύνδρομο της ληφθείσας λέξης ορίζεται ως:

$$S = r \circ H^T = [0\ 0\ 1\ 1\ 1\ 0] \circ \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 1 & 0 \\ 0 & 1 & 1 \\ 1 & 0 & 1 \end{bmatrix} = [1\ 1 \oplus 1\ 1 \oplus 1] = [1\ 0\ 0]$$

Επιπλέον,

$$e = r \oplus c = [1\ 0\ 0\ 0\ 0\ 0]$$

Τέλος, επαληθεύουμε ότι

$$S = e \circ H^T = [1 \ 0 \ 0 \ 0 \ 0 \ 0] \cdot \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 1 & 0 \\ 0 & 1 & 1 \\ 1 & 0 & 1 \end{bmatrix} = [1 \ 0 \ 0]$$

Δηλαδή ότι σύνδρομο του r ταυτίζεται με το σύνδρομο του e .

2.4.5 Διόρθωση σφαλμάτων

Όπως ήδη αναφέρθηκε, οι γραμμικοί συμπαγείς κώδικες έχουν μία βασική ιδιότητα, η οποία είναι απαραίτητη για την διεξαγωγή της κωδικοποίησης. Αυτή είναι ότι η αναλογία είναι ένα προς ένα μεταξύ των διανυσμάτων σφάλματος που μπορούν να διορθωθούν και των συνδρόμων. Η πρώτη ασχολία όσον αφορά τη διόρθωση των σφαλμάτων, είναι η δημιουργία ενός πίνακα σύγκρισης των ενδεχόμενων συνδρόμων με τα ανάλογα διανύσματα των οποίων ο κώδικας επιδέχεται διόρθωση.

Όταν ο αριθμός των λαθών έγκειται εντός των δυνατοτήτων αποκατάστασης του κώδικα, εφαρμόζεται η εξής λογαριθμική διαδικασία για την διόρθωσή τους.

1. Υπολογίζεται το σύνδρομο της ληφθείσας κωδικοποιημένης λέξης ως $S = e \circ H^T$.
2. Από τον πίνακα αντιστοίχισης συνδρόμου και διανύσματος λαθών προκύπτει η εκτίμηση του συστήματος για το διάνυσμα λαθών \hat{e} .
3. Από τη σχέση $\hat{c} = r \oplus \hat{e}$, ο αποκωδικοποιητής εξάγει την εκτίμηση του συστήματος για τη μεταδοθείσα κωδική λέξη.

Παράδειγμα 2.5 Έστω ότι έχουμε έναν κώδικα, όμοιο με το προηγούμενο παράδειγμα 2.4. Τότε, από τη σχέση $S = e \circ H^T$ κατασκευάζεται ο πίνακας 2.2 που αντιστοιχεί τα πιθανά σύνδρομα με τα αντίστοιχα διανύσματα λαθών.

Πίνακας 2.2 Αντιστοίχιση διανύσματος λαθών και συνδρόμου για το παράδειγμα 2.5

Διάνυσμα λαθών	Σύνδρομο
000000	000
000001	101
000010	011
000100	110
001000	001
010000	010
100000	100
010001	111

Αν μεταδόθηκε η κωδικοποιημένη λέξη $c = (1\ 0\ 1\ 1\ 1\ 0)$ και ελήφθη η λέξη $r = (0\ 0\ 1\ 1\ 1\ 0)$, τότε το σύνδρομο της ληφθείσας λέξης προκύπτει από το προηγούμενο παράδειγμα $S = [1\ 0\ 0]$. Χρησιμοποιώντας τον πίνακα 2.2, προκύπτει η εκτίμηση για το αντίστοιχο διάνυσμα λαθών $\hat{e} = [1\ 0\ 0\ 0\ 0\ 0]$. Το διορθωμένο διάνυσμα βρίσκεται ως

$$\hat{c} = r \oplus \hat{e} = 001110 \oplus 100000 = 101110$$

Καθώς $\hat{c} = c$, ο αποκωδικοποιητής έκανε σωστή διόρθωση λαθών.

2.5 Ικανότητα Εντοπισμού και Διόρθωσης Λαθών

2.5.1 Βάρος και Απόσταση Δυαδικών Διανυσμάτων

Δυστυχώς, δεν γίνεται να διορθωθούν όλα τα σφάλματα. Παραδείγματος χάριν, εν συνεχεία των προαναφερθέντων για τους γραμμικούς συμπαγείς κώδικες, αν γίνει μετάδοση μιας κωδικής λέξης και καταλήξει στον αποκωδικοποιητή αλλοιωμένη σε σχέση με την αρχική κωδική λέξη η οποία αναλογεί σε κάποια άλλη κωδική λέξη, τότε η κωδική λέξη που ελήφθη θα θεωρηθεί σωστή αν το σύνδρομο είναι μηδενικό. Το συγκεκριμένο σφάλμα δεν είναι δυνατόν να διορθωθεί αλλά ούτε και να ανιχνευτεί

(καθοριστεί). Για να αναλύσουμε εις βάθος την διορθωτική ικανότητα ενός κώδικα θα χρειαστούμε τις έννοιες του βάρους και της απόστασης.

Ο αριθμός των ψηφίων που δεν περιέχουν το μηδέν σε μια κωδική λέξη U ονομάζεται βάρος w . Σε ένα δυαδικό διάνυσμα όπως αυτά που θα αναφερθούμε το βάρος του ισούται με τον συνολικό αριθμό των άσπων που περιέχονται στο διάνυσμα. Ο αριθμός των θέσεων σε δύο κωδικοποιημένες λέξεις στις οποίες οι λέξεις διαφέρουν ονομάζεται απόσταση d . Τις δύο παραπάνω έννοιες θα τις δούμε και ως απόσταση Hamming και βάρος Hamming. Για παράδειγμα έσω ότι έχουμε δύο κωδικές λέξεις U, V με

$$U = 1010$$

$$V = 1110$$

Τότε, $w(U)=2$, $w(V)=3$ και $d(U, V)=1$.

Από την ιδιότητα που έχει η modulo-2 άθροισης συμπεραίνουμε ότι η πρόσθεση δύο δυαδικών είναι η δημιουργία ενός νέου δυαδικού διανύσματος, στο οποίο οι άσποι είναι οι διαφορές ανάμεσα στα δύο διανύσματα και στα υπόλοιπα ψηφία μηδενικά. Για παράδειγμα,

$$U \oplus V = 0100$$

Παρατηρούμε ότι η απόσταση Hamming μεταξύ δύο κωδικών λέξεων ισούται με το βάρος Hamming της πρόσθεσης τους, δηλαδή $d(U, V)=w(U \oplus V)$. Επίσης, εικάζοντας από το βάρος Hamming μια κωδικής λέξης είναι ίσο με την απόσταση της λέξης από το μηδενικό διάνυσμα.

2.5.2 Ελάχιστη Απόσταση ενός Γραμμικού Κώδικα

Ελάχιστη απόσταση d_{\min} ενός κώδικα χαρακτηρίζεται η ελάχιστη απόσταση ανάμεσα στις κωδικές λέξεις, του κώδικα λαμβανομένων κατά δύο με όλους τους εφικτούς συνδυασμούς.

Όπως προαναφερθήκαμε, η πρόσθεση δύο κωδικών λέξεων είναι ομοίως μια κωδική λέξη. Δηλαδή αν U, V δύο κωδικές λέξεις τότε η πρόσθεση τους $U \oplus V = W$ πρέπει να αποτελεί μια κωδική λέξη. Προηγουμένως αναφέραμε ότι η απόσταση ανάμεσα σε δύο κωδικές λέξεις είναι ίσον με το βάρος του αθροίσματος τους:

$d(U, V) = w(U \oplus V) = d(W)$. Επομένως δεν είναι αναγκαίο να ελέγξουμε όλα τα ζεύγη κωδικών λέξεων για τον εντοπισμό της ελάχιστης απόστασης, παρά μόνο να ελέγξουμε το βάρος όλων των ψηφίων που δεν περιέχουν το μηδέν στις λέξεις του κώδικα και να υπολογίσουμε την ελάχιστη απόσταση ως το ελάχιστο βάρος αυτών των λέξεων. Το μηδενικό διάνυσμα αφαιρέθηκε από την διαδικασία διότι δεν μπορεί να προκύψει ως άθροισμα διαφορετικών διανυσμάτων. **Αντιθέτως**, προκύπτει ως άθροισμα μιας κωδικής λέξης με τον εαυτό της $U \oplus V = 0$. Η ελάχιστη απόσταση χαρακτηρίζεται ως η απόσταση ανάμεσα σε διαφορετικές λέξεις και όχι σαν την απόσταση μιας λέξης με τον εαυτό της που είναι μηδενική.

Η βασική έννοια της ελάχιστης απόστασης είναι η ελάχιστη ικανότητα και η δύναμη που έχει ο κώδικας. Επίσης η ελάχιστη απόσταση χαρακτηρίζει την δύναμη του κώδικα και όχι η μέγιστη απόσταση. (εδώ υπάρχει και ένα παράδειγμα με τον κρίκο της αλυσίδας που χαρακτηρίζει το ελάχιστο και το μέγιστο δεν έγραψα γιατί δεν βρήκα παρόμοιο παράδειγμα, αν βρείτε εσείς κάτι που να ταιριάζει συμπληρώστε το)

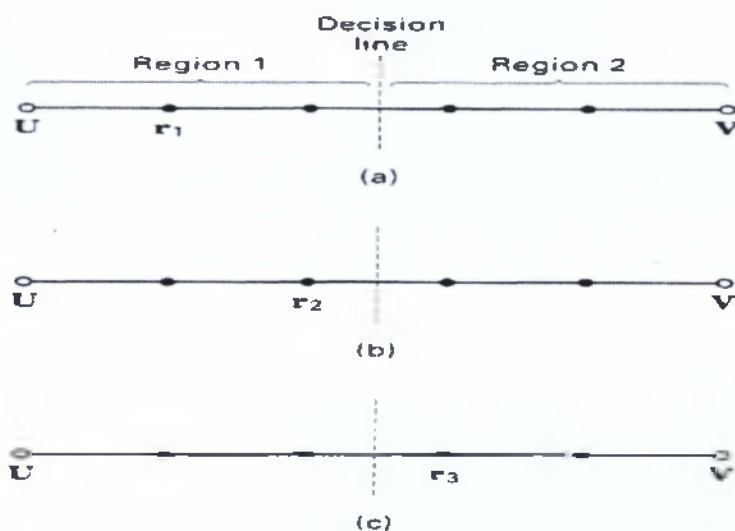
2.5.3 Εντοπισμός και Διόρθωση Λαθών

Με την παραλαβή της κωδικού διανύσματος r στον αποκωδικοποιητή, ο αποκωδικοποιητής πρέπει να κάνει τη βέλτιστη εκτίμηση για τη μεταδοθείσα κωδική λέξη. Η πιθανότητα εκπομπής μιας κωδικής λέξης U δεδομένης της λήψης του διανύσματος r είναι αντιστρόφως ανάλογη με την απόσταση ανάμεσα στα διανύσματα r, U . Άρα η βέλτιστη εκτίμηση βρίσκεται στην εύρεση της από τον αποκωδικοποιητή της κωδικής λέξης με την ελάχιστη απόσταση από το διάνυσμα r .

Αναλύοντας το σχήμα 2.5 βλέπουμε ότι η απόσταση μεταξύ των κωδικών λέξεων U, V απεικονίζεται ως απόσταση Hamming. Κάθε αλλοιωμένη κωδική λέξη απεικονίζεται με μια μαύρη τελεία. Στο σχήμα 2.5α απεικονίζεται η λήψη του διανύσματος r_1 απέχει 1 κατά Hamming από την κωδική λέξη U και 4 από την κωδική λέξη V . Όπως εξηγήσαμε προηγουμένως, ένας αποκωδικοποιητής βέλτιστης πιθανοτικής εκτίμησης θα διαλέξει το διάνυσμα U σε σύγκριση με το V καθώς απέχει λιγότερο από το r_1 . Αν στην περίπτωση που το r_1 προέκυψε ως παραμόρφωση του U κατά ένα bit, τότε ο αποκωδικοποιητής διόρθωσε σωστά το λάθος, αλλιώς αν το r_1 είναι αποτέλεσμα παραμόρφωσης του V κατά τέσσερα bits τότε έχουμε λάθος στη διαδικασία αποκωδικοποίησης. Ομοίως διπλό σφάλμα θα έχουμε στην μετάδοση του U όπου οδηγεί στο r_2 που απέχει δύο κατά Hamming από το U και 3 από το V , όπως

διακρίνεται στο σχήμα 2.5β. Ομοίως και εδώ η επιλογή του αποκωδικοποιητή θα είναι το διάνυσμα U . Επίσης στο σχήμα 2.5γ αναφέρουμε την περίπτωση τριπλού σφάλματος κατά την μετάδοση του U που οδηγεί στη λήψη του διανύσματος r_3 . Στην συγκεκριμένη περίπτωση και με την ίδια ακριβώς στρατηγική, ο αποκωδικοποιητής θα διαλέξει την κωδική λέξη V πραγματοποιώντας μια εσφαλμένη εκτίμηση. Από το σχήμα 2.5 συμπεραίνουμε ότι ο σκοπός είναι μόνο ο εντοπισμός και όχι η διόρθωση των λαθών με αποτέλεσμα λάθη με 1, 2, 3, ή 4 bits της μεταδοθείσας λέξης να είναι εντοπίσιμα. Αντιθέτως με 5 λανθασμένα bits της μεταδοθείσας κωδικής λέξης να καταλήξει στην λήψη της κωδικής λέξης V ενώ είχε αποσταλεί η κωδική λέξη U το οποίο σαν σφάλμα είναι μη εντοπίσιμο.

Γενικά από το σχήμα 2.5 συμπεραίνουμε ότι η ικανότητα εντοπισμού σφαλμάτων και η ικανότητα διόρθωσή τους σε ένα κώδικα αποτελούνται από την ελάχιστη απόσταση του κώδικα. Στο παράδειγμα μας, όπου το κριτήριο απόφασης για την για την επιλογή της κωδικής λέξης U είναι αν το r βρίσκεται στην περιοχή 1 (ομοίως για την V αν το r βρίσκεται στην περιοχή 2), βλέπουμε ότι ένας τέτοιος κώδικας με ελάχιστη απόσταση 5 μπορεί να διορθώσει μέχρι 2 λάθη.



Σχήμα 2.9 Σχετικά με τη διορθωτική ικανότητα (α) Λήψη r_1 , (β) Λήψη r_2 , (γ) Λήψη r_3

Σύμφωνα με τα παραπάνω έχουμε ότι ένας κώδικας με ελάχιστη απόσταση d_{\min} μπορεί αναμφισβήτητα να διορθώνει μέχρι και t λάθη ανά κωδική λέξη όπου το t ορίζεται από την παρακάτω σχέση:

$$t = \left\lfloor \frac{d_{min}-1}{2} \right\rfloor \quad (2.15)$$

Όπου με $\lfloor x \rfloor$ συμβολίζουμε το ακέραιο μέρος του x .

Έπειτα από την ανάλυση που προηγήθηκε συμπεραίνουμε ότι ένας κώδικας μπορεί αναμφισβήτητα να εντοπίσει το λάθος αν τα σφάλματα ανά κωδική λέξη είναι μικρότερα ή ίσα με από την τιμή e η οποία δίνεται από την παρακάτω σχέση:

$$e = d_{min} - 1 \quad (2.16)$$

2.6 Αξιολόγηση Συστημάτων Κωδικοποίησης

2.6.1 Χωρητικότητα Διαύλου – Θεώρημα Shannon

Το 1948 ο Shannon απέδειξε ότι μπορούν να μειωθούν τα σφάλματα που εισάγονται κατά την διάρκεια μετάδοσης σε διαύλους που διακρίνονται από λευκό προσθετικό θόρυβο με κωδικοποίηση της ψηφιακής πληροφορίας χωρίς να γίνει αύξηση του ρυθμού μετάδοσης πληροφορίας R_i , προϋποθέτοντας όμως ο ρυθμός R_i να μην υπερβαίνει την χωρητικότητα C του διαύλου. Η χωρητικότητα C δίνεται από την σχέση:

$$C = B \log_2 \left(1 + \frac{S}{N} \right) \quad (2.17)$$

όπου B το εύρος ζώνης συχνοτήτων του διαύλου και S/N ο σηματοθορυβικός λόγος (Signal-to-noise ratio, SNR) στην είσοδο του δέκτη. Για το σηματοθορυβικό λόγο ισχύει

$$\frac{S}{N} = \frac{E_b}{n_0} \times \frac{R_i}{B} \quad (2.18)$$

όπου E_b η ενέργεια ανά bit και n_0 η μονόπλευρη φασματική πυκνότητα θορύβου. Σύμφωνα με την (2.18), η σχέση (2.17) γράφεται ως

$$C = B \log_2 \left(1 + \frac{R_i \times E_b}{n_0 \times B} \right) \quad (2.19)$$

Ισοδύναμα,

$$C = R_i \frac{E_b}{n_0} \left(z \log_2 \left(1 + \frac{1}{z} \right) \right) \quad (2.20)$$

με

$$z = \left(\frac{E_b}{n_0}\right)^{-1} \times \left(\frac{B}{R_i}\right) \quad (2.21)$$

Καθώς

$$\lim_{z \rightarrow \infty} z \log_2 \left(1 + \frac{1}{z}\right) = \log_2 e \cong 1.443 \quad (2.22)$$

λόγω της (2.20) παίρνουμε

$$\lim_{B \rightarrow \infty} C = 1.443 R_i (E_b/n_0) \quad (2.23)$$

Επειδή ο ρυθμός μετάδοσης R_i δεν πρέπει να υπερβαίνει τη χωρητικότητα C

$$R_i \leq \lim_{B \rightarrow \infty} C = 1.443 R_i (E_b/n_0) \quad (2.24)$$

από όπου προκύπτει

$$(E_b/n_0) \geq 0.693 = -1.6 \text{ db} \quad (2.25)$$

Η παραπάνω σχέση θέτει ένα ελάχιστο όριο για την τιμή του λόγου (E_b/n_0) , πάνω από το οποίο μπορούμε να επιτύχουμε θεωρητικά όσο μικρή πιθανότητα λάθους θέλουμε πάντα με το κατάλληλο σχήμα κωδικοποίησης. Επίσης διατυπώνει ένα σημαντικό θεωρητικό όριο για τις τηλεπικοινωνίες.

2.6.2 Κέρδος Κωδικοποίησης

Το κέρδος κωδικοποίησης δίνεται από την σχέση:

$$G_c = \frac{(E_b/n_0)_u}{(E_b/n_0)_c} \quad (2.26)$$

όπου $(E_b/n_0)_u$ και $(E_b/n_0)_c$ είναι οι λόγοι ενέργειας bit προς πυκνότητα θορύβου χωρίς ή με κωδικοποίηση αντίστοιχα, έτσι ώστε το ποσοστό λαθών να είναι το ίδιο και στις δύο περιπτώσεις. Η παράμετρος που ορίστηκε μέσω της σχέσης (2.26) και που ονομάστηκε κέρδος κωδικοποίησης ουσιαστικά ποσοτικοποιεί το πλεονέκτημα της χρήσης κωδικοποίησης σε όρους επιτρεπτής μείωσης του σηματοθορυβικού λόγου.

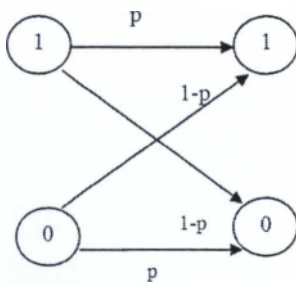
Σε για διαύλους που χαρακτηρίζονται από λευκό προσθετικό θόρυβο, το κέρδος κωδικοποίησης ορίζεται προσεγγιστικά ως

$$G_c = 10 \log_2 \left(\frac{k}{n} d_{min} \right) \quad (2.27)$$

Παρατηρούμαι ότι το κέρδος κωδικοποίησης αυξάνεται με την αύξηση της ελάχιστης απόστασης d_{min} σύμφωνα με την ανάλυση που έγινε στην ενότητα 2.4.

2.6.3 Διακριτό Μοντέλο Διαύλου

Στο παρακάτω σχήμα απεικονίζεται το διακριτό μοντέλο διαύλου με δυαδικό αλφάβητο. Εξαιτίας της συμμετρίας που υπάρχει η πιθανότητα μετάβασης έπειτα από τη μετάδοση, από 1 σε 0 είναι όμοια με την πιθανότητα μετάδοσης από 0 σε 1 και ορίζεται ως $(1 - p)$. Η πιθανότητα μη τροποποίησης της κατάστασης του μεταδιδόμενου ψηφίου ορίζεται με p . Σε συστήματα που δεν χρησιμοποιείται κωδικοποίηση η πιθανότητα $(1-p)$ εκφράζει τη μέση πιθανότητα λάθους ανά μεταδιδόμενο ψηφίο που θα συμβολίζεται P_b .



Σχήμα 2.10 Διακριτό μοντέλο διαύλου

2.6.4 Πιθανότητα λάθους στους γραμμικούς συμπαγείς κώδικες

Σε αυτήν την ενότητα θα συγκρίνουμε θεωρητικά την πιθανότητα λάθους χωρίς κωδικοποίηση και με κωδικοποίηση για διόρθωση λαθών σε ένα γραμμικό συμπαγή κώδικα (n, k) με την ικανότητα διόρθωσης t λαθών ανά κωδικοποιημένη λέξη. Για την αξιοπιστία της σύγκρισης εικάζουμε ότι ο ρυθμός μετάδοσης πληροφορίας R_i και η ισχύς εκπομπής C μένουν αμετάβλητες. Ορίζοντας με E_b την ενέργεια ψηφίου στην περίπτωση μη χρήσης κωδικοποίησης και με E_c την ενέργεια ψηφίου σε περίπτωση χρήσης κωδικοποίησης έχουμε:

$$E_i = \frac{C}{R_i} \quad (2.28)$$

και

$$E_c = \frac{C}{R_c} \quad (2.29)$$

όπου R_c είναι ο ρυθμός μετάδοσης ψηφίων σε περίπτωση χρήσης κωδικοποίησης. Σύμφωνα με τα πρωταρχικά χαρακτηριστικά του κώδικα, έχουμε ότι σε n εκπεμπόμενα ψηφία τα k αντιστοιχούν σε μετάδοση ψηφίων πληροφορίας. Από τη στιγμή που υποθέσαμε ότι ο ρυθμός μετάδοσης πληροφορίας μένει αμετάβλητος πρέπει να ισχύει

$$R_c = \frac{n}{k} R_i \quad (2.30)$$

Τότε,

$$E_c = \frac{C}{R_c} = \frac{C}{nR_i/k} = \frac{k}{n} E_b < E_b \quad (2.31)$$

Στην περίπτωση που δεν θα χρησιμοποιηθεί κωδικοποίηση λάθος ανά λέξη θα έχουμε όταν τουλάχιστον ένα ψηφίο της λέξης ληφθεί λανθασμένα. Έτσι, η πιθανότητα λάθους ανά λέξη προκύπτει ως συμπληρωματική πιθανότητα του ενδεχόμενου όλα τα μεταδιδόμενα ψηφία να ληφθούν δίχως σφάλμα. Έχοντας τον ίδιο συμβολισμό που εισήχθη στο διακριτό μοντέλο διαύλου παίρνουμε για την πιθανότητα λάθους ανά μη κωδικοποιημένη λέξη την παρακάτω σχέση:

$$P_e = 1 - p^k \quad (2.32)$$

Τώρα αν στην περίπτωση που γίνει κωδικοποίηση η πιθανότητα λάθους ανά κωδικοποιημένη λέξη P_{ec} φράσσεται από το όριο

$$P_{ec} \leq \sum_{i=t+1}^n \binom{n}{i} p_c^{n-i} (1 - p_c)^i \quad (2.33)$$

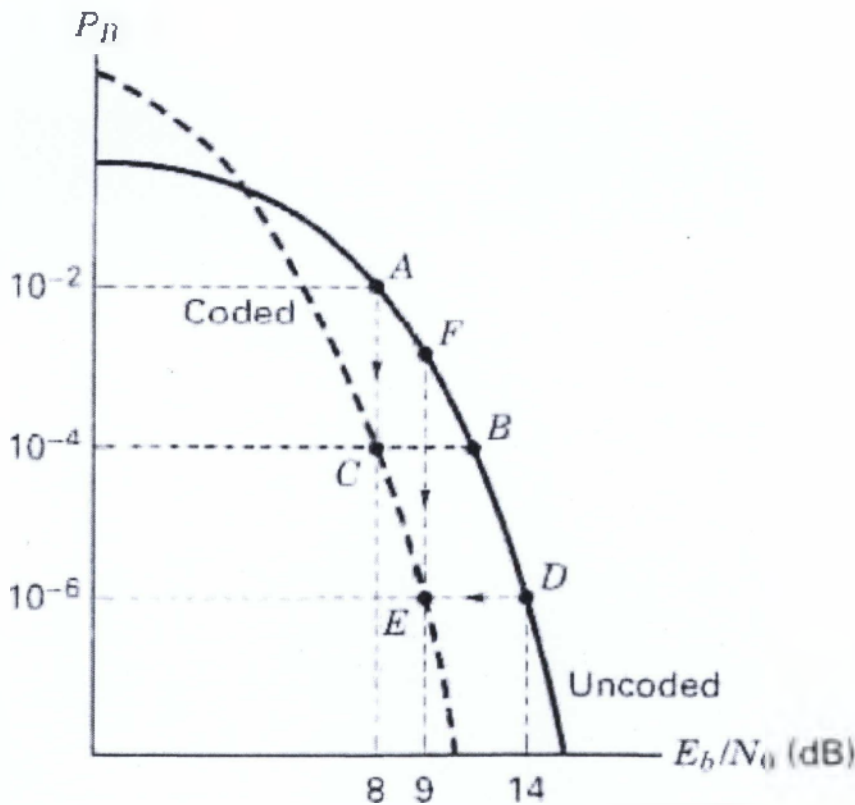
που αντιστοιχεί στην πιθανότητα να γίνουν περισσότερα από t σφάλματα στα n ψηφία της λέξης. Με $(1 - p_c)$ ορίζουμε την πιθανότητα εσφαλμένης μετάβασης ψηφίου όταν χρησιμοποιείται κωδικοποίηση.

Αν $n(1 - p_c) \ll 1$ προσεγγίζουμε το παραπάνω άθροισμα κρατώντας μόνο τον πρώτο όρο που είναι σημαντικά μεγαλύτερος και παίρνουμε

$$P_{ec} \leq \binom{n}{t+1} p_c^{n-t-1} \times (1 - p_c)^{t+1} \approx \binom{n}{t+1} (1 - p_c)^{t+1} \quad (2.34)$$

Για τα σύγχρονα τηλεπικοινωνιακά συστήματα η χρήση κωδικοποίησης για διόρθωση λαθών (FEC) είναι αχώριστο κομμάτι. Στον χώρο των ασύρματων επικοινωνιών είναι απαραίτητη η χρήση πρόσθιας διόρθωσης λαθών (FEC). Σε αυτό το σημείο θα διαχωρίσουμε τις δορυφορικές επικοινωνίες που φορτώνονται από τις ιδιαιτερότητες που εμφανίζει η διάδοση ηλεκτρομαγνητικών κυμάτων στην ατμόσφαιρα και τις κινητές επικοινωνίες που φορτώνονται από φαινόμενα πολυδιαδρομικής διάδοσης του σήματος.

Στην συγκεκριμένη ενότητα θα αναφερθούμε σε έναν γενικό σχολιασμό σχετικά στο ισοζύγιο κερδών-απωλειών χρησιμοποιώντας κωδικοποίηση για την διόρθωση λαθών. Η σύγκριση θα γίνει χωρίς την χρήση κάποιου σχήματος κωδικοποίησης. Η τυπική καμπύλη που συγκρίνει την επίδοση συστήματος με χρήση κωδικοποίησης ή χωρίς απεικονίζεται στο σχήμα 2.7. Η επίδοση του συστήματος εκφράζεται σε όρους πιθανότητας λάθους ανά μεταδιδόμενο ψηφίο (P_b).



Σχήμα 2.11 Τυπική σύγκριση επίδοσης χωρίς και με κωδικοποίηση

Από το σχήμα 2.7 συμπεραίνουμε ότι υπάρχουν κάποια συμπεράσματα κάνοντας χρήση κωδικοποίησης η οποία απεικονίζεται με την διακεκομμένη γραμμή όμως έχουν και το ανάλογο κόστος.

2.6.5 Ισοζύγιο Επίδοσης-Εύρους Ζώνης

Θεωρώντας τηλεπικοινωνιακό σύστημα το οποίο δεν χρησιμοποιεί κωδικοποίηση και το σημείο A ($E_b/N_0 = 8dB, P_b = 10^{-2}$) της καμπύλης στο σχήμα 2.7 ως το σημείο λειτουργίας του. Αν υποθέσουμε ότι απαιτείται βελτίωση της επίδοσης του συστήματος σε όρους πιθανότητας λάθους στην τιμή 10^{-4} . Για να έχουμε μια βελτίωση απόδοσης στο συγκεκριμένο σύστημα θα πρέπει το σημείο λειτουργίας να μετατοπιστεί στο σημείο B της καμπύλης. Όπως φαίνεται, αυτή η μετατόπιση προϋποθέτει αύξηση της τιμής του λόγου E_b/N_0 κάτι που δεν είναι πάντα εφικτό κυρίως λόγω της περιορισμένης ισχύος των πομποδεκτών. Σε αυτήν την περίπτωση η κωδικοποίηση μπορεί να αποτελέσει λύση επειδή από την κωδικοποιημένη καμπύλη φαίνεται ότι το σημείο C αποτελεί όλες τις προδιαγραφές επίδοσης χωρίς να απαιτεί αύξηση του σηματοθορυβικού λόγου. Το κόστος αυτής της ποιοτικής βελτίωσης είναι η αύξηση του χρησιμοποιημένου εύρους ζώνης συχνοτήτων. Αυτό γιατί η κωδικοποίηση χρησιμοποιεί πλεονάζοντα δεδομένα και επίσης για να διατηρείται σταθερός ο ρυθμός μετάδοσης πληροφορίας θα πρέπει ο ολικός ρυθμός μετάδοσης R_c να αυξηθεί όπως αναφέρεται από τη σχέση 2.30.

$$R_c = \frac{n}{k} R_i$$

Αυτή η αύξηση του ολικού ρυθμού μετάδοσης απαιτεί δέσμευση μεγαλύτερου εύρους ζώνης.

2.6.6 Ισοζύγιο Ισχύος-Εύρους Ζώνης

Στο σημείο λειτουργίας D ($E_b/N_0 = 14dB, P_b = 10^{-6}$) της καμπύλης 2.7 η επίδοση φαίνεται ικανοποιητική αλλά ο σηματοθορυβικός λόγος θα ήταν καλύτερα να μειωθεί. Κάνοντας χρήση κωδικοποίησης υπάρχει η δυνατότητα μετατόπισης του σημείου λειτουργίας D στο σημείο E χωρίς να γίνουν αλλαγές στα επίπεδα ποιότητας-

επίδοσης κατορθώνουμε να μειώσουμε τον απαιτούμενο λόγο E_b/N_0 δημιουργώντας έτσι κέρδος κωδικοποίησης όπως αυτό διατυπώθηκε μέσω της σχέσης (2.26). Το κόστος σε αυτή την περίπτωση είναι το ίδιο με προηγουμένως: κατανάλωση περισσότερου εύρους ζώνης.

Σημειώνουμε ότι η απαίτηση για σταθερό ρυθμό μετάδοσης πληροφορίας που προκαλεί και την αύξηση του εύρους ζώνης αφορά τηλεπικοινωνιακά συστήματα πραγματικού χρόνου (real-time) όπως είναι η τηλεφωνία. Σε συστήματα μη πραγματικού χρόνου το κόστος θα ήταν η χρονοκαθυστέρηση αντί του εύρους ζώνης.

2.6.7 Ισοζύγιο Ρυθμού Μετάδοσης-Εύρους Ζώνης

Η τιμή του σηματοθορυβικού λόγου στο δέκτη (SNR) ορίζεται από τη σχέση

$$\frac{S}{N} = \frac{E_b \times R}{N_0 W} \quad (2.35)$$

με S τη λαμβανόμενη ισχύ, R το ρυθμό μετάδοσης (bit/sec) και W το εύρος ζώνης του συστήματος.

Ισοδύναμα

$$\frac{E_b}{N_0} = \frac{S}{N_0} \left(\frac{1}{R} \right) \quad (2.36)$$

Αν θέλουμε να αυξήσουμε το ρυθμό μετάδοσης R στο σημείο λειτουργίας D χρησιμοποιώντας την παραπάνω σχέση θα έχουμε μείωση του λόγου E_b/N_0 και μετατόπιση του σημείου λειτουργίας στα αριστερά της καμπύλης, έστω στο σημείο F όπου έχουμε μείωση της επίδοσης. Έπειτα από χρήση κωδικοποίησης το σύστημα μπορεί να μεταφερθεί στο σημείο E στο οποίο κρατώντας σταθερή την ποιότητα υπηρεσίας κατορθώνουμε και την επιθυμητή αύξηση του ρυθμού μετάδοσης. Ομοίως το κόστος είναι το αυξημένο εύρος ζώνης.

3 Προσομοίωση Τεχνικών Κωδικοποίησης για την Εκτίμηση της Επίδοσής τους

3.1 Γενικά περί προσομοίωσης

Η διαδικασία της προσομοίωσης, καθίσταται όλο και πιο απαραίτητη στα μοντέρνα τηλεπικοινωνιακά συστήματα. Αυτή η ανάπτυξη οφείλεται σε διάφορες συνιστώσες. Αρχικά, η αυξανόμενη πολυπλοκότητα των επικοινωνιακών συστημάτων από αρχιτεκτονικής άποψης, κάνει αρκετά απαιτητική την ανεύρεση εκτεταμένων διατυπώσεων για την αποδοτικότητα τους. Επιπλέον, λεπτομερείς διατυπώσεις δύσκολα υφίστανται στην μοντελοποίηση ενός τηλεπικοινωνιακού δίαυλου. Παραδείγματος χάριν, στον τομέα των κινητών επικοινωνιών, υπάρχουν περιβάλλοντα που προσανατολίζονται σε ισχυρές διαλείψεις στην πλευρά του δέκτη, πράγμα που το καθιστά δύσκολο στο να μοντελοποιηθεί μέσω λεπτομερών σχέσεων.

Αποτέλεσμα στα όσα αναφέρθηκαν, μπορεί να αποφέρει επιτυχώς η προσομοίωση. Με βασικό σύμμαχο την εξέλιξη των υπολογιστών, είναι δυνατόν να πραγματοποιηθεί μελέτη του συστήματος έπειτα από μία διαδικασία που περιέχει αρκετά μεγάλο αριθμό επαναλήψεων, από τις οποίες η κάθε μία κάνει έναν υπολογισμό για την αποδοτικότητα του συστήματος. Μέσω αυτού του πλήθους των επαναλήψεων, ο οποίος θα ήταν ανεπίτευκτος χωρίς τη διάθεση υπολογιστικού συστήματος, μας επιτρέπει να εξάγουμε μία έγκυρη άποψη αυτού. Ταυτόχρονα, η υποχρέωση για αποδοτική προσομοίωση έχει φέρει σαν αποτέλεσμα τη δημιουργία ισχυρών και αποτελεσματικών λογισμικών.

Διακρίνουμε ότι η εξέλιξη της προσομοίωσης προέρχεται από την ανάπτυξη δύο βασικών τομέων: των υπολογιστών και των τηλεπικοινωνιών. Επομένως τμήματα που περιλαμβάνονται σε αυτούς τους τομείς, όπως είναι η αριθμητική ανάλυση και η θεωρία επικοινωνιών συνεισφέρουν στην πρόοδο της προσομοίωσης. Στο Σχήμα 3.1 διακρίνονται οι βασικού τομείς που συμβάλλουν στη μελέτη και ανάπτυξη της προσομοίωσης σε ένα τηλεπικοινωνιακό σύστημα. Έπειτα, θα ακολουθήσει μία περιληπτική περιγραφή του κάθε ρόλου καθ' ενός ξεχωριστά από τους εννέα εικονιζόμενους τομείς.



Σχήμα 3.1 Επιστημονικοί τομείς σχετικοί με την προσομοίωση

Η θεωρία γραμμικών συστημάτων (linear system theory) παρέχει τις απαραίτητες πληροφορίες για την περιγραφή της σχέσης εισόδου – εξόδου των γραμμικών συστημάτων. Συνεπώς, κατά τη διάρκεια της προσομοίωσης έχουμε τη δυνατότητα μέσω αυτών των σχέσεων να μοντελοποιήσουμε τη λειτουργία ενός συστατικού ή υποσυστήματος του συνολικού συστήματος με τη βοήθεια είτε του πεδίου του χρόνου είτε της συχνότητας.

Η θεωρία επικοινωνιών (communication theory) είναι σίγουρα ένα βασικό κομμάτι αυτής της διαδικασίας. Η δομή του συστήματος επικοινωνιών, τα λειτουργικά χαρακτηριστικά των υποσυστημάτων, όπως και οι αποκωδικοποιητές και ο τηλεπικοινωνιακός διάυλος, είναι δεδομένα που πρέπει να γνωρίζουμε για να γίνει η μοντελοποίηση και να οδηγήσει σε αξιόπιστα αποτελέσματα μετά την ολοκλήρωση της προσομοίωσης. Είναι σημαντικό να τονίσουμε ότι αυτό που πιστεύεται είναι ότι το πιο απαιτητικό στη μοντελοποίηση είναι τα χαρακτηριστικά του τηλεπικοινωνιακού διαύλου, ειδικά μάλιστα στην περίπτωση που στη ζεύξη υπάρχουν διαλείψεις, ένα συχνό φαινόμενο στο περιβάλλον των κινητών επικοινωνιών όπως έχουμε ήδη αναφέρει.

Όσον αφορά τη θεωρία επεξεργασία σημάτων (digital signal processing), συμβάλλει κατά κύριο λόγο στην ανάπτυξη προσεγγίσεων διακριτού χρόνου για συστήματα συνεχούς χρόνου, όπως είναι τα φίλτρα.

Τώρα, σχετικά με την αριθμητική ανάλυση (numerical analysis), είναι αλληλένδετη με την ανάπτυξη αποδοτικών υπολογιστικών τεχνικών. Περίπλοκα συστήματα, πολυδαίδαλες λύσεις διαφορικών εξισώσεων μπορούν να προσεγγίσουν με αποτελεσματικό τρόπο και συνήθως μέσω επαναληπτικών διαδικασιών, κάνοντας χρήση των τεχνικών αριθμητικής ανάλυσης.

Ξεχωριστό ρόλο στην προσομοίωση των τηλεπικοινωνιακών συστημάτων διατηρεί και η θεωρία πιθανοτήτων (probability theory). Αρχικά, αρκετά συχνά η απόδοση των τηλεπικοινωνιακών συστημάτων αποδίδεται με όρους πιθανοτικούς, όπως για παράδειγμα η πιθανότητα σφάλματος η οποία επιβάλλεται να είναι μικρότερη από 0.001 για 98% του χρόνου λειτουργίας. Επιπλέον, η προσομοίωση ως επί το πλείστον, έχει σαν αποτέλεσμα μια τυχαία μεταβλητή και για να θεωρηθεί η προσομοίωση επιτυχής ή ανεπιτυχής, πρέπει να μελετηθεί η τυπική απόκλιση της. Μια μεγάλη τυπική απόκλιση επιφέρει μεταβολές στα αποτελέσματα της προσομοίωσης, κάτι που θεωρείται μειονέκτημα. Στο ίδιο πεδίο, εφαρμόζεται και η θεωρία στοχαστικών διαδικασιών (stochastic process theory). Ένα παράδειγμα στοχαστικής διαδικασίας είναι ο Λευκός Προσθετικός Θόρυβος Gauss (Additive White Gaussian Noise – A.W.G.N) και εμείς στην προσομοίωση κάθε φορά δημιουργούμε ένα δείγμα αυτής της στοχαστικής διαδικασίας, η οποία εκφράζεται στην προκειμένη περίπτωση μέσω της συνάρτησης πυκνότητας πιθανότητας.

Στον σχηματισμό τυχαίων ακολουθιών μπορεί να συμβάλλει η θεωρία αριθμών (number theory). Παραδείγματος χάριν, στην προσομοίωσή μας, θέλουμε η είσοδος του κωδικοποιητή να αποτελείται από μία τυχαία ακολουθία από 0 ή 1.

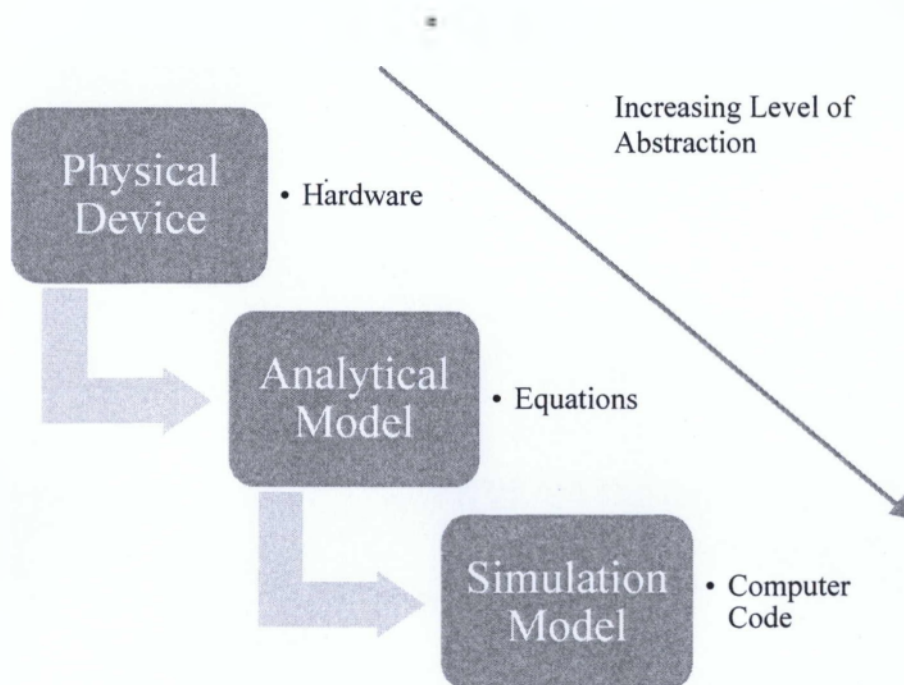
Ορισμένα ουσιώδεις στοιχεία της επιστήμης υπολογιστών (computer science) μπορούν ομοίως να εφαρμοστούν. Μερικά από αυτά τα στοιχεία είναι η επιλογή της ενδεδειγμένης γλώσσας προγραμματισμού για κάθε προσομοίωση όπως επίσης και οι τεχνικές των αποδοτικών αλγορίθμων οι οποίες έχουν τη δυνατότητα να μειώσουν όχι μόνο τον συνολικό χρόνο της προσομοίωσης αλλά και την αναγκαία υπολογιστική μνήμη.

Τέλος, έχουμε τη θεωρία εκτίμησης (estimation theory), η οποία μας δίνει τα εφόδια για την εκτίμηση ενός αποτελέσματος προσομοίωσης. Αυτό είναι απαραίτητο, διότι όπως προαναφέρθηκε, το αποτέλεσμα αντιμετωπίζεται ως μια τυχαία μεταβλητή.

Για να θεωρηθεί εύστοχη η εκτίμηση πρέπει να είναι αμερόληπτη (unbiased) και συνεπής (consistent).

3.2 Μοντελοποίηση

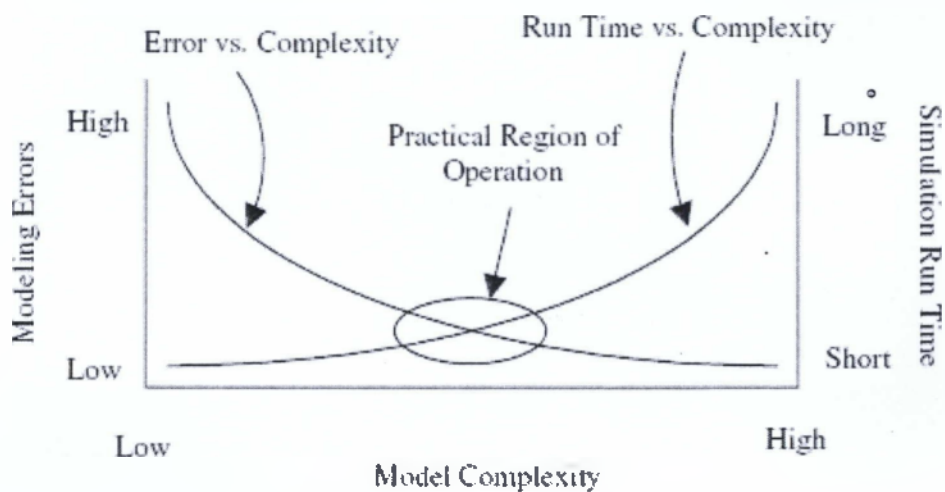
Το πρώτο στάδιο για την εκτέλεση μιας προσομοίωσης ενός συστήματος είναι η μοντελοποίηση του. Η μοντελοποίηση περιέχει δύο κατηγορίες μοντέλων, οι οποίες διαχωρίζονται σύμφωνα με τη χρονική σειρά ανάπτυξης: τα προσομοιωτικά μοντέλα και τα αναλυτικά μοντέλα. Αυτή η διαδικασία διακρίνεται στο Σχήμα 3.2.



Σχήμα 3.2 Διαδικασία ανάπτυξης μοντέλων

Όπως διακρίνεται, πάντα αρχίζουμε τη μελέτη ενός πραγματικού στοιχείου το οποίο μπορεί να είναι για παράδειγμα ένα φίλτρο, μπορεί όμως να είναι και ένα ολοκληρωμένο τηλεπικοινωνιακό σύστημα. Η μελέτη του μας οδηγεί στην εξέλιξη ενός λεπτομερούς μοντέλου που το περιγράφει και που συνήθως εκφράζεται μέσω εξισώσεων. Οι εξισώσεις που παράχθηκαν είναι μία αφαιρετική περιγραφή του στοιχείου το οποίο είναι προς μελέτη και περιλαμβάνουν πολλές υποθέσεις και προσεγγίσεις. Έπειτα, ένα αφαιρετικό επίπεδο παραπάνω, το αναλυτικό μοντέλο, διαμορφώνεται σε προσομοιωτικό μοντέλο, το οποίο τελικά θα εκφραστεί ως κώδικας κάποιας γλώσσας προγραμματισμού. Και σε αυτή τη μετάβαση υπάρχουν πολλές

μεταβολές και προσεγγίσεις. Ένας ενδεχόμενος λόγος είναι ότι ένα ακριβές αναλυτικό μοντέλο ίσως οδηγήσει σε αυξημένο χρόνο εκτέλεσης και προσομοίωσης, λόγω της περιπλοκότητάς τους, κάτι που απαιτεί μείωση της πολυπλοκότητας μέσω προσεγγίσεων. Τέτοια ισοζύγια (trade – offs) συναντώνται στην πλειοψηφία των προσομοιώσεων και διακρίνονται συνοπτικά στο Σχήμα 3.3. Όπως και να έχει, σε κάθε περίπτωση πρέπει να εξασφαλίζεται ότι το μοντέλο της προσομοίωσης είναι έγκυρο και τα αποτελέσματά του αντικατοπτρίζουν την πραγματικότητα.



Σχήμα 3.3 Ακρίβεια αποτελεσμάτων και χρόνος εκτέλεσης συναρτήσει της πολυπλοκότητας του μοντέλου

3.3 Ντετερμινιστικές και Στοχαστικές Προσομοιώσεις

Υπάρχουν δύο κατηγορίες προσομοιώσεων: οι στοχαστικές και οι ντετερμινιστικές. Λέγοντας ντετερμινιστικές, μιλάμε για τις προσομοιώσεις (και γενικότερα για τις διαδικασίες) οι οποίες έχουν προκαθορισμένο αποτέλεσμα. Παραδείγματος χάριν, έχουμε την προσομοίωση ηλεκτρικού κυκλώματος, το οποίο αποτελείται από γνωστές πηγές τάσης σταθερής τιμής καθώς επίσης και από γνωστές αντιστάσεις σταθερής τιμής. Το αποτέλεσμα ενδείκνυται να είναι πάντα το ίδιο. Το ίδιο αποτέλεσμα θα απόρρεε ακόμα και αν το επιλύαμε με την κλασσική μέθοδο, δηλαδή το χαρτί και το μολύβι. Ο λόγος που θα προτιμούσαμε να γίνει η προσομοίωση στον υπολογιστή, είναι λόγω των χρονοβόρων υπολογιστικών πράξεων.

Υπάρχουν διαδικασίες που δεν έχουν προκαθορισμένα αποτελέσματα, διότι περιλαμβάνουν και την συνιστώσα της τυχαιότητας. Σε αυτού του είδους τις διαδικασίες, τα αποτελέσματα αντιμετωπίζονται ως τυχαίες μεταβλητές. Αυτό το είδος διαδικασιών και προσομοιώσεων ονομάζονται στοχαστικές. Ένα παράδειγμα το οποίο έχει άμεση σχέση με την προσομοίωση που θα εκτελέσουμε παρακάτω, είναι ένα ψηφιακό τηλεπικοινωνιακό σύστημα, από το οποίο το σήμα που λαμβάνεται θα είναι το άθροισμα ενός τυχαιού θορύβου και του μεταδιδόμενου σήματος. Αν υποθέσουμε ότι έχουμε να υπολογίσουμε την πιθανότητα λάθους ανά σύμβολο, γνωρίζοντας ότι αν χρησιμοποιηθεί BPSK διαμόρφωση και ο δίαυλος είναι της μορφής AWGN, η πιθανότητα ορίζεται ως εξής:

$$P_E = Q\left(\sqrt{\frac{2E_b}{N_0}}\right) \quad (3.1)$$

Όπου $Q(x)$ η συνάρτηση Gauss η οποία ως:

$$Q(x) = \frac{1}{\sqrt{2\pi}} \int_x^{\infty} \exp\left(\frac{-y^2}{2}\right) dy \quad (3.2)$$

Όσον αφορά την προσομοίωση, προκύπτει ένα πρόβλημα και αυτό είναι ότι η προαναφερθείσα πιθανότητα έχει προκύψει θεωρητικά, μελετώντας άπειρο αριθμό συμβόλων. Όμως, είναι εμφανές ότι σε μία προσομοίωση είναι δυνατόν να προσομοιωθεί συγκεκριμένος αριθμός συμβόλων, που στη προκειμένη περίπτωση αναφερόμαστε σε bits.

Η στρατηγική σε αυτές τις περιπτώσεις είναι να γίνει η προσομοίωση ενός μεγάλου πλήθους bits και να μετρηθούν τα λάθη στην έξοδο του δέκτη. Στην περίπτωση που μεταδώσουμε N ψηφία και σημειωθούν N_e λάθη, η εκτίμηση για την πιθανότητα είναι:

$$\bar{P}_E = \frac{N_e}{N} \quad (3.3)$$

Αυτός ο ρυθμός λαθών ορίζεται ως BER (Bit Error Rate). Μέσω του ορισμού της πιθανότητας και χρησιμοποιώντας τη σχετική συχνότητα συμπεραίνεται ότι:

$$P_E = \lim_{N \rightarrow \infty} \frac{N_e}{N} \quad (3.4)$$

Όπως ήδη αναφέρθηκε, έχουμε τη δυνατότητα να προσομοιάσουμε συγκεκριμένο αριθμό ψηφίων, επομένως η ζητούμενη πιθανότητα μπορεί να

προσεγγιστεί μόνο από την προσομοίωση. Συνεπώς, η επιτυχία της προσομοίωσης εξαρτάται από το πόσο ακριβής είναι η προσέγγιση – εκτίμησή της.

Αφού εξετάσουμε το κάθε αποτέλεσμα της προσομοίωσης ξεχωριστά, το οποίο αντιμετωπίζεται σαν μια τυχαία μεταβλητή στη στοχαστική προσομοίωση, για να αξιολογηθεί ως επιτυχής η ολοκλήρωση της προσομοίωσης, πρέπει να μελετήσουμε τα στατιστικά χαρακτηριστικά της. Κατά κύριο λόγο, μας ενδιαφέρουν δύο στατιστικά χαρακτηριστικά της εκτίμησης μίας προσομοίωσης. Αρχικά, θέλουμε την τυπική απόκλιση της εκτίμησης να τείνει στο μηδέν όταν το N τείνει στο άπειρο, γεγονός το οποίο καθιστά την εκτίμηση συνεπή (consistent). Επιπλέον, θέλουμε τη μέση τιμή της εκτίμησης να ισούται με το πραγματικό μέγεθος, το οποίο στην προκειμένη περίπτωση είναι $E\left\{\hat{P}_E\right\} = P_E$, όταν το N τείνει στο μηδέν. Αυτή η ιδιότητα καθιστά την εκτίμηση αμερόληπτη (unbiased).

3.4 Επιλογή της Πλατφόρμας για την Προσομοίωση μας

Ένας από τους λόγους που πραγματοποιείται η εργασία αυτή, είναι η ανάπτυξη προσομοιωτή για την αποτίμηση της επίδοσης των κωδικών LDPC, τους οποίους θα αναλύσουμε στο επόμενο κεφάλαιο, υπό διάφορα σχήματα διαμόρφωσης (BPSK, QAM, κ.λ.π) και υπό διαφορετικούς τύπους διαύλων (AWGN, Rice, Rayleigh). Το πρώτο θέμα που πρέπει να απαντηθεί είναι τι πλατφόρμα προσομοίωσης θα επιλέξουμε, δηλαδή πάνω σε ποιο προγραμματιστικό περιβάλλον θα πραγματοποιηθεί η προσομοίωση.

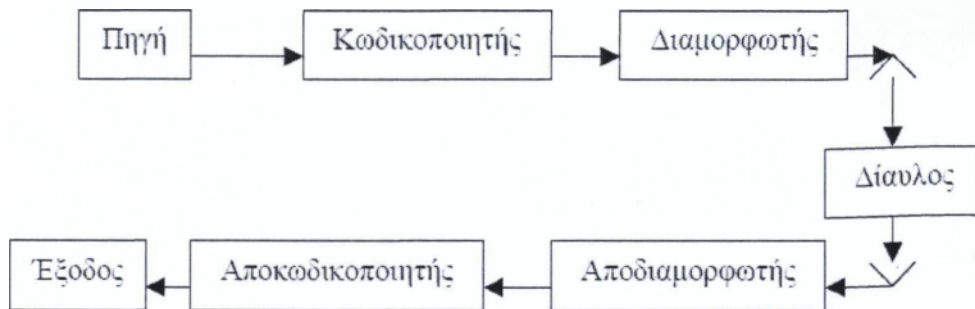
Οι κύριοι υποψήφιοι είναι το περιβάλλον Matlab και η γλώσσα προγραμματισμού C. Ο κύριος λόγος είναι η ποιότητα της απόδοσης του στις απαιτητικές υπολογιστικές διεργασίες, όπως είναι η προσομοίωσή μας. Όπως ήδη έχει γίνει κατανοητό, η εκτίμηση της επίδοσης ενός κώδικα συνιστά στοχαστική διαδικασία, εφόσον η είσοδος τόσο στον κωδικοποιητή όσο και στον αποκωδικοποιητή της προσομοίωσης είναι μία τυχαία μεταβλητή σε κάθε της βήμα. Στον κωδικοποιητή, τα ψηφία φτάνουν κατά τυχαίο τρόπο από τη ψηφιακή πηγή πληροφορίας και ο αποκωδικοποιητής στην είσοδό του περικλείει και τη τυχειότητα που έχει εισαχθεί από τη στιγμιαία τιμή των χαρακτηριστικών του διαύλου. Όντας στοχαστική προσομοίωση, απαιτείται μεγάλος αριθμός επαναλήψεων, επομένως απαιτούνται και πολλοί υπολογισμοί.

Λόγω του ότι η γλώσσα προγραμματισμού C είναι γλώσσα χαμηλού επιπέδου, είναι η πιο αποδοτική σε σχέση με τον χρόνο εκτέλεσης του Matlab. Το Matlab όμως έχει στοιχεία τα οποία υπερτερούν. Αρχικά είναι ευρέως αποδεκτό από την κοινότητα των μηχανικών. Επιπλέον, από τα έμφυτα χαρακτηριστικά του περιβάλλοντος Matlab (MATrix LABoratory) υπάρχει μεγάλη ευχέρεια και αποτελεσματικότητα στη χρήση πινάκων, κάτι το οποίο θα φανεί χρήσιμο (γεννήτορας πίνακας, πίνακας ελέγχου ισοτιμίας, εύρεση αντιστρόφου κ.α). Γενικότερα, από τη χρήση πινάκων, σαφώς και έχει πιο εμπλουτισμένη βιβλιοθήκη με έτοιμες ρουτίνες από τη γλώσσα C παράλληλα με πολύ βοηθητικά πρόσθετα όπως το Simulink. Συγκεκριμένα, το Matlab εμπλουτίζεται συνεχώς, κάτι το οποίο φαίνεται από τις συνεχείς ανανεώσεις των εκδόσεών του. Χαρακτηριστικό παράδειγμα είναι η έκδοση του Matlab R2007, η οποία είναι η πρώτη που περιλαμβάνει προσομοιωτικό μοντέλο για το νέο πρότυπο DVB-S2 που χρησιμοποιεί LDPC κωδικοποίηση. Σαφώς και τέτοιες δυνατότητες δεν μπορούν να προσφερθούν από μια γλώσσα προγραμματισμού χαμηλού επιπέδου όπως η C. Αξίζει να σημειώσουμε ότι και αυτό που προσφέρει καλύτερα η C, δηλαδή ταχύτητα εκτέλεσης, μπορεί εν μέρει να χρησιμοποιηθεί, καθώς υπάρχει η δυνατότητα ενσωμάτωσης στο περιβάλλον Matlab αρχείων γραμμένα σε C. Τέλος, στην προσομοίωσή μας σημαντικό ρόλο παίζει και η οπτική παρουσίαση των αποτελεσμάτων και το γραφικό περιβάλλον του Matlab είναι αναμφίβολα από τα πιο εξελιγμένα.

3.5 Μοντελοποίηση ενός απλού πομποδέκτη για την εκτίμηση της επίδοσής του

Στην ενότητα αυτή θα περιγράψουμε το τηλεπικοινωνιακό σύστημα που θα προσομοιώσουμε στο επόμενο κεφάλαιο, με σκοπό να μελετήσουμε την απόδοσή του σε όρους ρυθμού λαθών ανά ψηφίο (BER). Η συγκεκριμένη μελέτη θα πραγματοποιηθεί με διαφορετικές διαμορφώσεις και με διαφορετικούς τύπους διαύλων και όλα αυτά συναρτήσει του σηματοθορυβικού λόγου (SNR ή E_b/N_0). Η κωδικοποίηση LDPC είναι το αμετάβλητο στοιχείο, εφόσον κάτι τέτοιο αποτελεί τον βασικό στόχο της εργασίας. Όμοια και στην LDPS κωδικοποίηση θα μελετηθούν διαφορετικοί παράγοντες όπως είναι ο λογαριθμικός ή μη λογαριθμικός αποκωδικοποιητής και διάφορες τιμές για το μέγιστο αριθμό επαναλήψεων του

επαναληπτικού αλγορίθμου αποκωδικοποίησης, ο οποίος θα μελετηθεί στο επόμενο κεφάλαιο. Στο κεφάλαιο 5 θα παρουσιαστούν και θα αναλυθούν τα αποτελέσματα της προσομοίωσης, της οποίας ο κώδικας είναι γραμμένος στο περιβάλλον Matlab. Έπειτα θα κάνουμε ανάλυση της μοντελοποίησης του κάθε συστήματος που εμπλέκεται στην προσομοίωσή μας. Η ανάλυση θα πραγματοποιηθεί βάσει του μπλοκ διαγράμματος του τηλεπικοινωνιακού συστήματος που θα χρησιμοποιήσουμε για την προσομοίωση μας και το οποίο διακρίνεται στο Σχήμα 3.4.



Σχήμα 3.4 Μπλοκ διάγραμμα προσποιούμενου συστήματος

3.5.1 Πηγή Ψηφιακής Πληροφορίας

Σαν πηγή του συστήματος έχουμε την ακολουθία δυαδικών ψηφίων η οποία καταλήγει με τυχαία σειρά στην είσοδο του κωδικοποιητή. Το τι είδος είναι η πληροφορία, δηλαδή ήχος, εικόνα κ.λ.π δεν μας απασχολεί, καθώς επίσης ούτε και τα ενδιάμεσα βήματα πριν τον κωδικοποιητή (βλ. Σχήμα 1.2), διότι δεν επιδρούν στο μέγεθος με το οποίο θα ασχοληθούμε, δηλαδή τον ρυθμό λαθών ανά ψηφίο (BER) στην έξοδο του αποκωδικοποιητή στο δέκτη.

Μέσω της συνάρτησης του Matlab `randint` μπορεί να περιγραφεί η προαναφερθείσα ακολουθία η οποία συνάρτηση έχει ίσες πιθανότητες να επιστρέψει 0 ή 1.

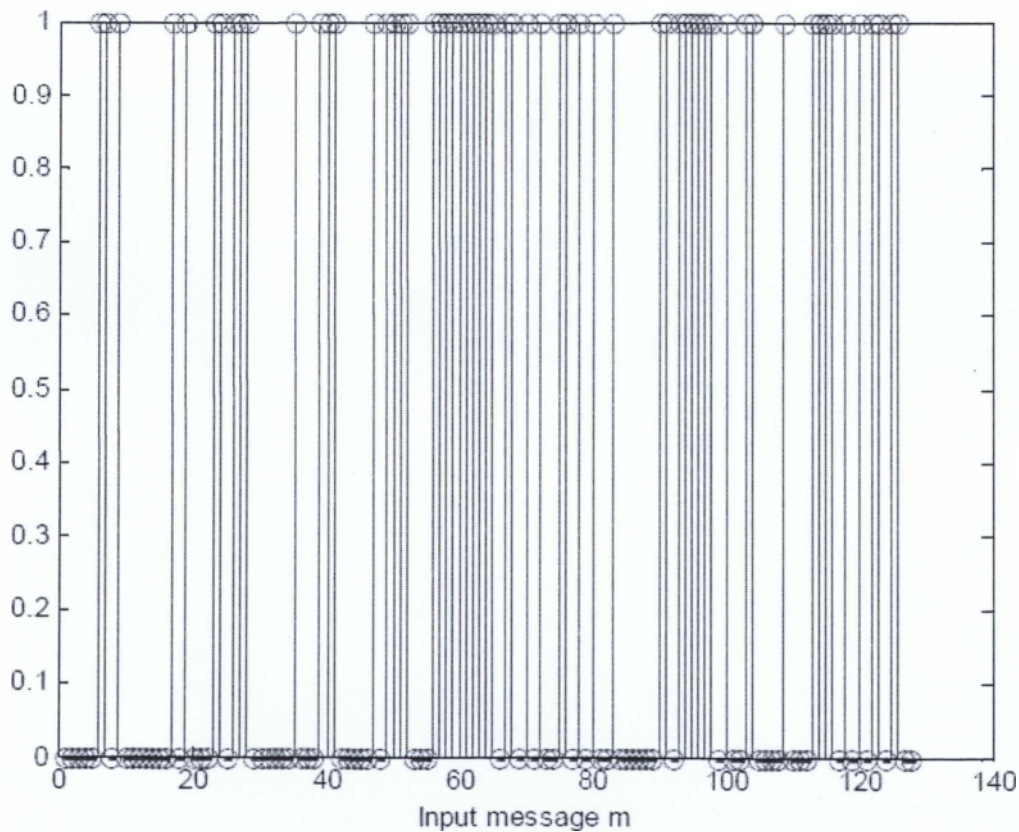
3.5.2 Κωδικοποιητής

Ο κώδικας που δημιουργήσαμε είναι κώδικας LDPC (256, 128), το οποίο δηλώνει ότι ο κωδικοποιητής δέχεται την είσοδο από την πηγή των 128 bits και στην έξοδό του δίνει πακέτα των 256 bits. Αν ορίσουμε με \mathbf{m} την ακολουθία των δυαδικών ψηφίων με μήκος 128 bits, στην έξοδο του αποκωδικοποιητή θα εμφανιστεί το κωδικοποιημένο μήνυμα \mathbf{c} , το οποίο έχει μήκος 256 bits και προκύπτει από την εξής σχέση:

$$\mathbf{c} = \mathbf{m} \circ \mathbf{G}$$

όπου \mathbf{G} είναι ο γεννήτορας πίνακας του κώδικα.

Με τη συνάρτηση stem του Matlab, δίνουμε μια ενδεχόμενη εκδοχή του \mathbf{m} στο Σχήμα 3.5.

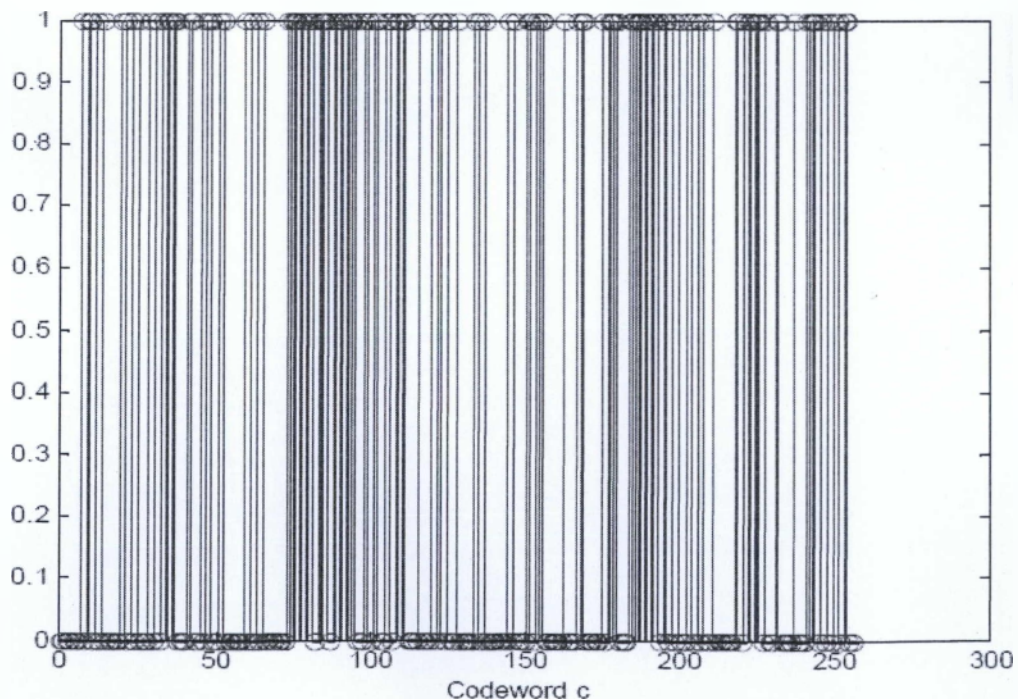


Σχήμα 3.5 Τυχαία είσοδος στον κωδικοποιητή

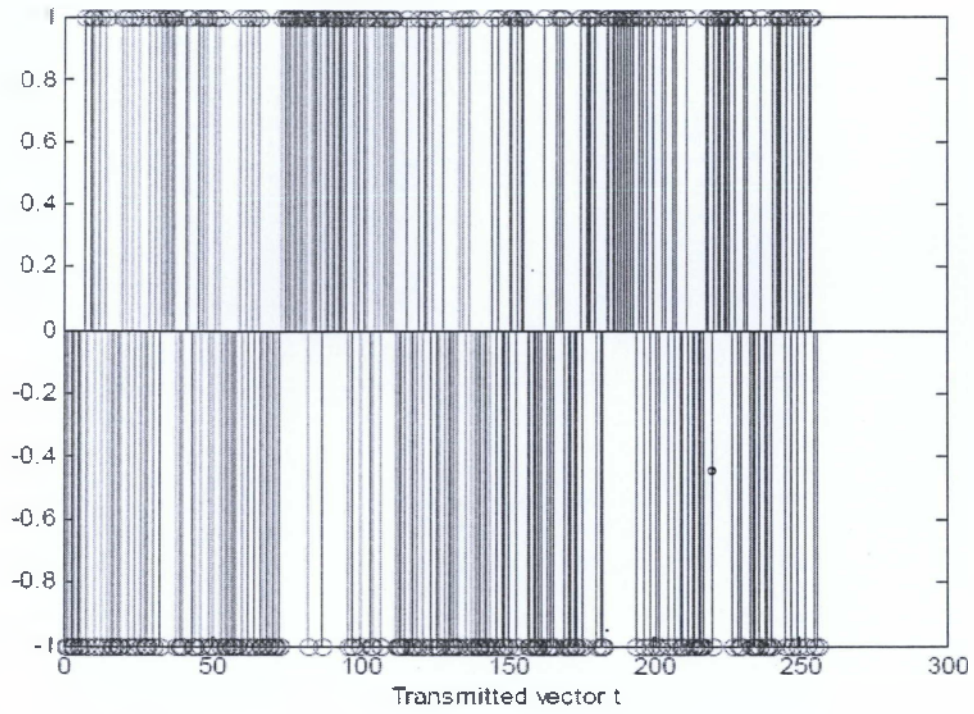
Στο Σχήμα 3.6 διακρίνεται το κωδικοποιημένο μήνυμα που αναλογεί στο προαναφερθέν διάλυσμα \mathbf{m} .

3.5.3 Διαμορφωτής

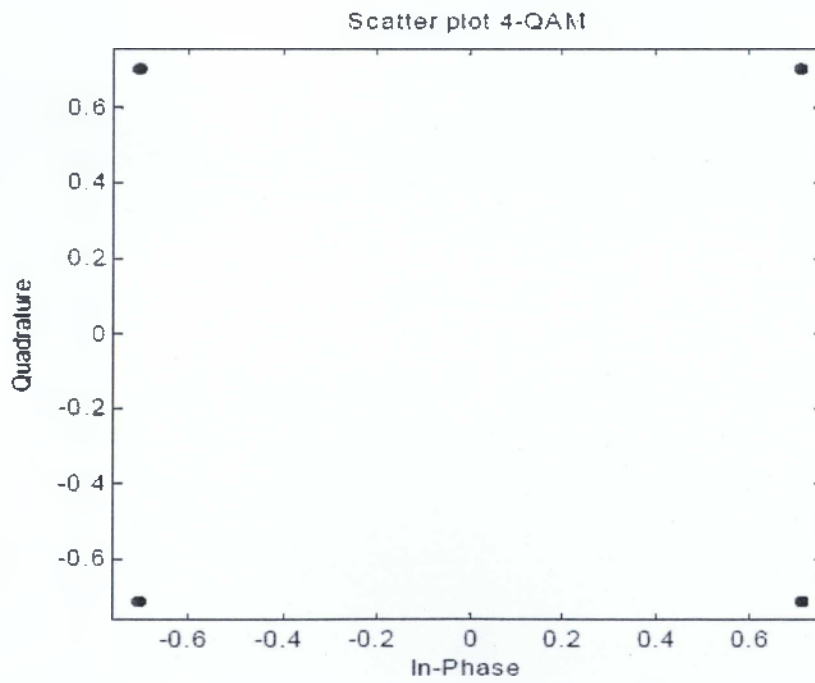
Ο κωδικοποιητής από την έξοδό του στέλνει κατά σειρά τις κωδικές λέξεις προς την είσοδο του διαμορφωτή. Το κάθε σχήμα διαμόρφωσης αναλογεί τα ληφθέντα δυαδικά ψηφία σε σύμβολα-παλμούς για την διαδικασία της μετάδοσης. Η διαμόρφωση BPSK που θα χρησιμοποιήσουμε έχει το πιο απλό σχήμα διαμόρφωσης στην οποία το ψηφίο 1 ισοδυναμεί σε παλμό πλάτους $+A$ και το ψηφίο 0 σε παλμό πλάτους $-A$. Στα παραδείγματά μας θα χρησιμοποιήσουμε την κανονικοποιημένη τιμή $+1$ και -1 (σχήμα 3.7). Ένα άλλο σχήμα διαμόρφωσης που θα χρησιμοποιήσουμε είναι η διαμόρφωση M-QAM στην οποία σε κάθε $\log_2 M$ ισοδυναμεί ένα σύμβολο. Για παράδειγμα στην διαμόρφωση 4-QAM ο διαμορφωτής ανά δύο ψηφία (00, 01, 10, 11) ισοδυναμεί και ένα σύμβολο, άρα συνολικά έχουμε 4 διαφορετικά σύμβολα. Με τον ίδιο τρόπο και για QAM διαμορφώσεις ανώτερης τάξης. Τα σχήματα που είναι για μετάδοση περιλαμβάνονται στο μιγαδικό επίπεδο και ερμηνεύονται σχηματικά στα σχήματα 3.8, 3.9.



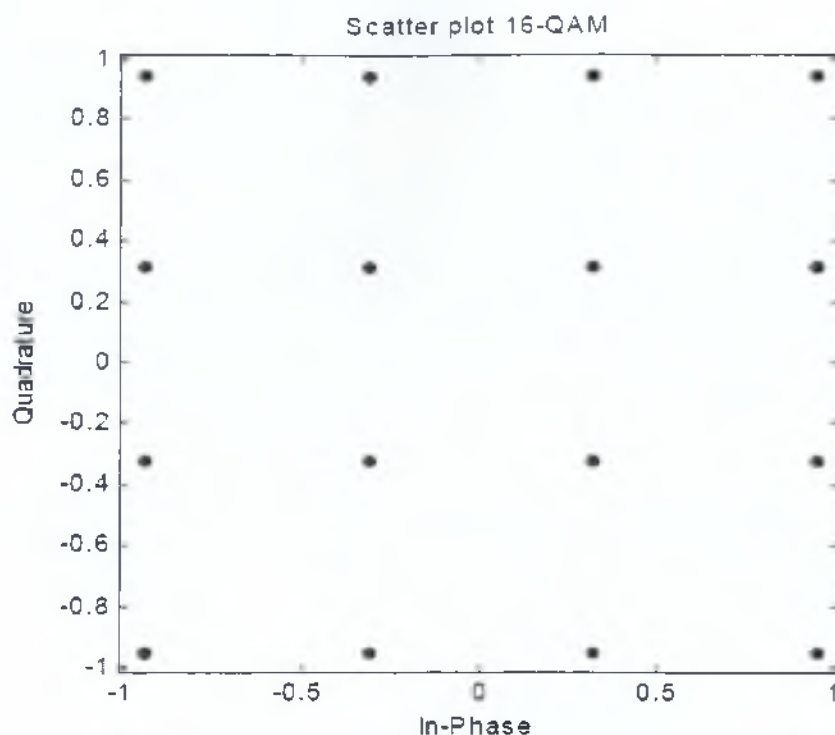
Σχήμα 3.6 Κωδική λέξη που αντιστοιχεί στο μήνυμα πληροφορίας του σχήματος 3.5



Σχήμα 3.7 BPSK διαμόρφωση στην κωδική λέξη του σχήματος 3.6



Σχήμα 3.8 Αστερισμός 4-QAM όπου φαίνονται τα 4 διαφορετικά μεταδιδόμενα σύμβολα



Σχήμα 3.9 Αστερισμός 16-QAM όπου φαίνονται τα 16 διαφορετικά μεταδιδόμενα σύμβολα

3.5.4 Διάυλος

Οι διάυλοι που θα ασχοληθούμε σχετίζονται κυρίως με τις ασύρματες επικοινωνίες. Σαν πρώτη κατηγορία επιμερίζουμε τους διαύλους σε τύπου προσθετικού λευκού θορύβου Gauss (AWGN) και σε διαύλους που προκαλούν διαλείψεις, δηλαδή απότομες μεταβολές στην ένταση του λαμβανόμενου σήματος. Και σαν δεύτερη κατηγορία θα ασχοληθούμε πρώτα με διαύλους τύπου Rayleigh και έπειτα με διαύλους τύπου Rice, καθώς επίσης θα αναφερθούμε και με την μεταβλητότητα του διαύλου λόγω κίνησης.

3.5.4.1 Προσθετικός Λευκός Θόρυβος Gauss (AWGN)

Για την περιγραφή του θορύβου που εισάγει ο διάυλος ορίστηκε το θεωρητικό μοντέλο AWGN, το οποίο χρησιμοποιείται για την ανάλυση τηλεπικοινωνιακών συστημάτων και διακρίνεται ως Gauss διότι έπεται της κατανομής Gauss.

Γενικά λέμε ότι η τυχαία μεταβλητή Y έχει κατανομή Gauss αν η συνάρτηση πυκνότητας πιθανότητας έχει την εξής μορφή:

$$p_Y(y) = \frac{1}{\sqrt{2\pi\sigma_Y^2}} \exp\left[-\frac{(y-m_Y)^2}{2\sigma_Y^2}\right] \quad (3.5)$$

Το m_Y είναι η μέση τιμή και το σ_Y^2 είναι η μεταβλητότητα της τυχαίας μεταβλητής Y . Η μοντελοποίηση θορύβου με βάση την κατανομή Gauss εκτελεί πρακτική εφαρμογή διότι προέρχεται από το θεώρημα του κεντρικού ορίου, το οποίο δηλώνει ότι αν μια μεταβλητή (ο θόρυβος για την συγκεκριμένη περίπτωση), είναι αποτέλεσμα ενός αθροίσματος μεγάλου αριθμού τυχαίων μεταβλητών ανεξάρτητων μεταξύ τους, τότε η κατανομή πιθανότητας της τυχαίας μεταβλητής προσεγγίζεται από τη κατανομή Gauss. Για παράδειγμα, έστω ο θερμικός θόρυβος, ο ηλεκτρικός θόρυβος δηλαδή θόρυβος που οφείλεται στην τυχαία κίνηση των ηλεκτρονίων σε έναν αγωγό. Λόγω του μεγάλου αριθμού των ηλεκτρονίων σε μια αντίσταση και των στατιστικών ανεξάρτητων τυχαίων κινήσεών του, το θεώρημα κεντρικού ορίου αναφέρει ότι ο θερμικός θόρυβος ακολουθεί κατανομή Gauss με μηδενική μέση τιμή.

Ο χαρακτηρισμός του θορύβου ως λευκός είναι μια ιδανική μορφή θορύβου, που η πυκνότητα του φάσματος ισχύος του οποίου είναι ανεξάρτητη της συχνότητας λειτουργίας του. Χαρακτηρίστηκε λευκός θόρυβος διότι το λευκό φως περιέχει ίσες ποσότητες ακτινοβολίας από όλες τις ορατές συχνότητες. Σύμφωνα με τον ορισμό του λευκού θορύβου έχουμε έπειτα από ανάλυση ότι οποιαδήποτε δυο διαφορετικά δείγματα λευκού θορύβου, ανεξάρτητα από το πόσο κοντά στο χρόνο λαμβάνονται, είναι ασυσχέτιστα και ανεξάρτητα. Επίσης, έχει αναφερθεί και ως το ζενίθ της τυχαιότητας για αυτό τον λόγο.

Λέγοντας, προσθετικός εννοούμε ότι το λαμβανόμενο σήμα $y(t)$ προκύπτει ως άθροισμα του μεταδιδόμενου σήματος $x(t)$ με το θόρυβο $n(t)$, για κάθε χρονική στιγμή t .

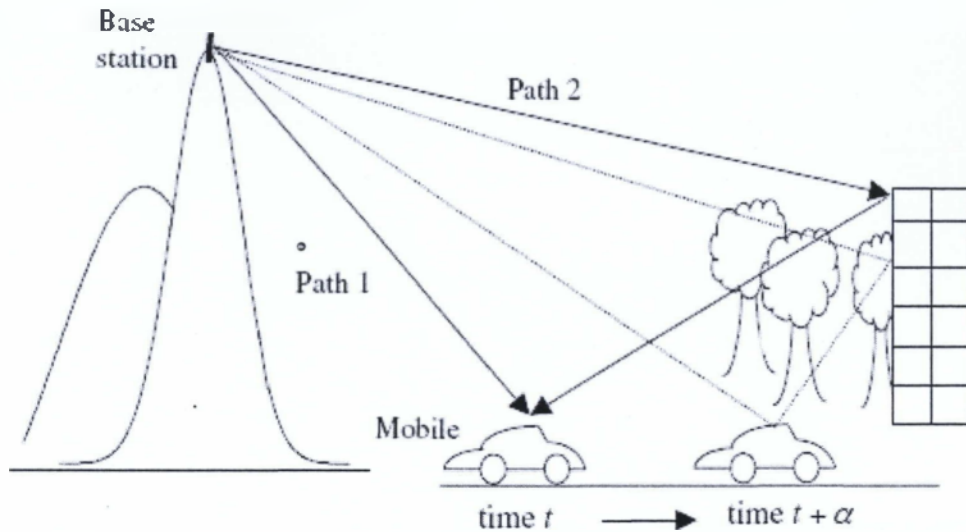
$$y(t) = x(t) + n(t) \quad (3.6)$$

Να σημειώσουμε ότι το Matlab περιέχει τη συνάρτηση AWGN, η οποία προσθέτει λευκό θόρυβο στο σήμα μας.

3.5.4.2 Δίαυλος τύπου Rayleigh

Δίαυλοι που προκαλούν διαλείψεις στο λαμβανόμενο σήμα εμφανίζονται συχνά στις τηλεπικοινωνίες και προκύπτουν κατά κύριο λόγο από την πολυδιαδρομική διάδοση του σήματος και την κίνηση του δέκτη (σχήμα 3.10).

Εμφανίζονται επίσης και στις δορυφορικές επικοινωνίες, όπου οι διαλείψεις προκύπτουν από τις ιδιαιτερότητες που εμφανίζει η διάδοση ηλεκτρομαγνητικών κυμάτων στην ατμόσφαιρα.



Σχήμα 3.10 Παράδειγμα πολυδιαδρομικής διάδοσης και κινητικότητας δέκτη που οδηγούν σε διαλείψεις.

Έχει αποδειχθεί ότι η κατανομή της περιβάλλουσας των σημάτων που προσβάλλονται διαλείψεις λόγω πολυδιαδρομικής διάδοσης όταν δεν υπάρχει οπτική επαφή πομπού – δέκτη (NLOS) περιγράφεται από την κατανομή Rayleigh. Η πυκνότητα πιθανότητας της κατανομής Rayleigh δίνεται από τη σχέση:

$$P_x(x) = \frac{2x}{b} \exp\left(-\frac{x^2}{b}\right), x > 0$$

$$P_x(x) = 0, x \leq 0$$
(3.7)

3.5.4.3 Διάυλος τύπου Rice

Όταν το σήμα μας προσβάλλεται από διαλείψεις και υπάρχει οπτική επαφή πομπού – δέκτη (LOS), τότε η περιβάλλουσα του σήματος ακολουθεί την κατανομή Rice. Η πυκνότητα πιθανότητας της κατανομής Rice δίνεται από την σχέση (a, b>0)

$$P_x(x) = \frac{x}{b} \exp\left[\frac{-x^2 + a^2}{2b}\right] I_0\left(\frac{ax}{b}\right), x > 0$$
$$P_x(x) = 0, x \leq 0 \quad (3.8)$$

όπου

- x , η λαμβανόμενη τιμή της περιβάλλουσας του σήματος (σε Volts)
- a , το πλάτος του κυρίαρχου σήματος (LOS)
- $I_0(x)$, η τροποποιημένη συνάρτηση Bessel πρώτου είδους

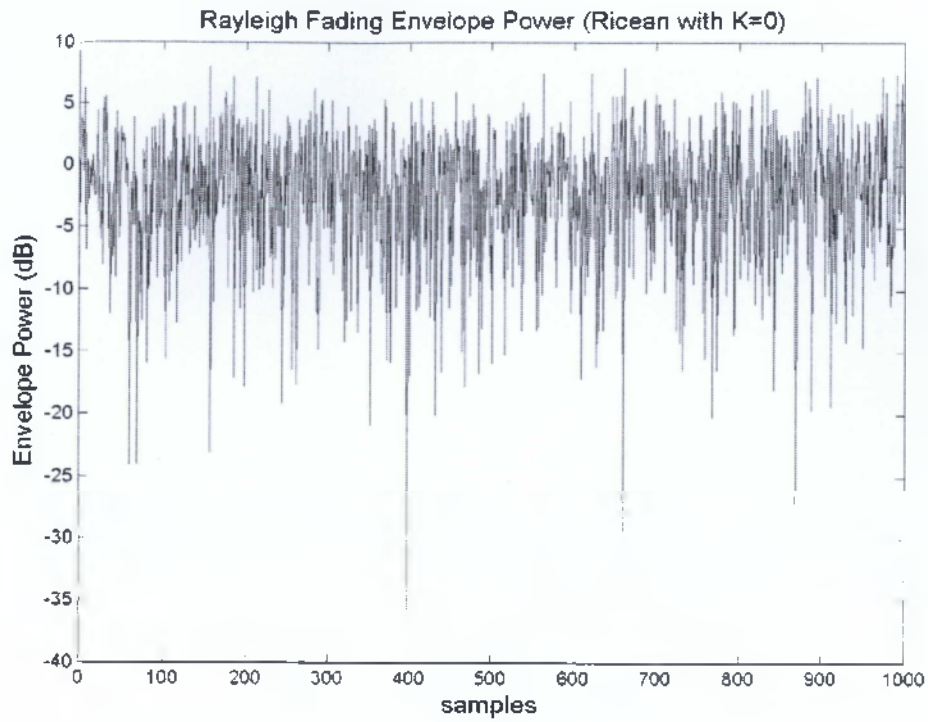
Για τον λόγο ότι υπάρχει οπτική επαφή πομπού - δέκτη περιμένουμε να έχουμε μικρότερες επιβαρύνσεις σε διαύλους τύπου Rice σε σχέση με διαύλους τύπου Rayleigh. Για να έχουμε έναν πιο ακριβή χαρακτηρισμό ενός διαύλου Rice χρησιμοποιούμε τον παράγοντα K , όπου:

$$K = \frac{\text{ΙΣΧΥΣ ΑΠΕΥΘΕΙΑΣ ΚΥΜΑΤΟΣ}}{\text{ΙΣΧΥΣ ΣΚΕΔΑΣΜΕΝΩΝ ΚΥΜΑΤΩΝ}}$$

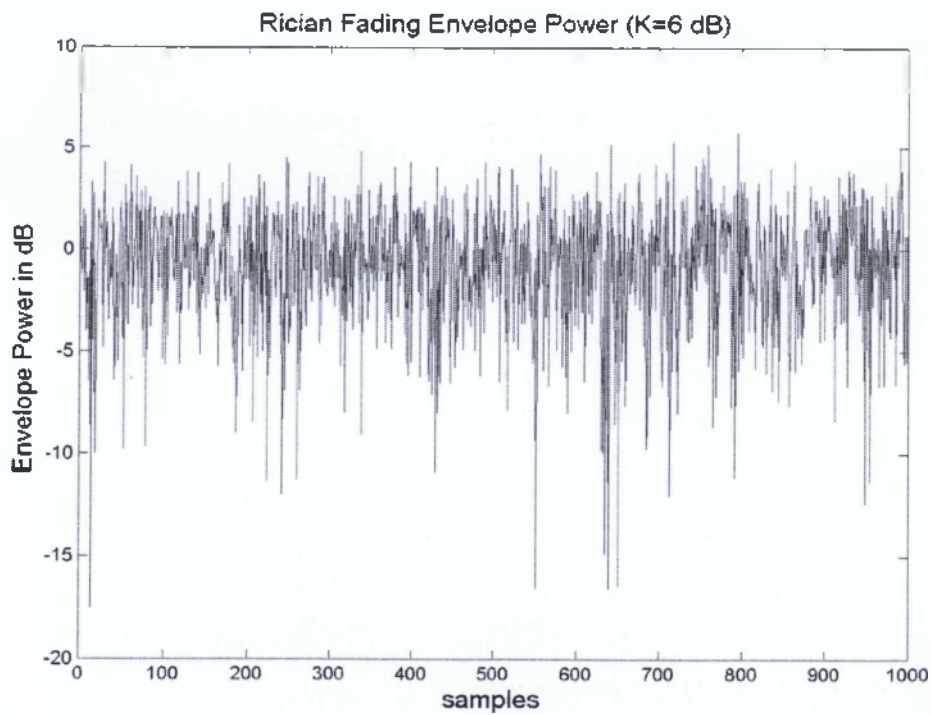
ή

$$K = \frac{a^2}{1b}$$

Σύμφωνα με τα παραπάνω, αν προκύψει $K=0$, ο διάυλος τύπου Rice ισοδυναμεί με διάυλο τύπου Rayleigh. Στα σχήματα που ακολουθούν απεικονίζουμε τα αποτελέσματα της προσομοίωσης που εκτελέσαμε στο Matlab για διαύλους Rayleigh (σχήμα 3.11) και Rice (σχήμα 3.12). Οι τιμές ισχύος της περιβάλλουσας του σήματος έχουν δημιουργηθεί σύμφωνα με τις πυκνότητες πιθανότητας του κάθε διαύλου, οι οποίες ορίζονται στις εξισώσεις 3.7, 3.8.



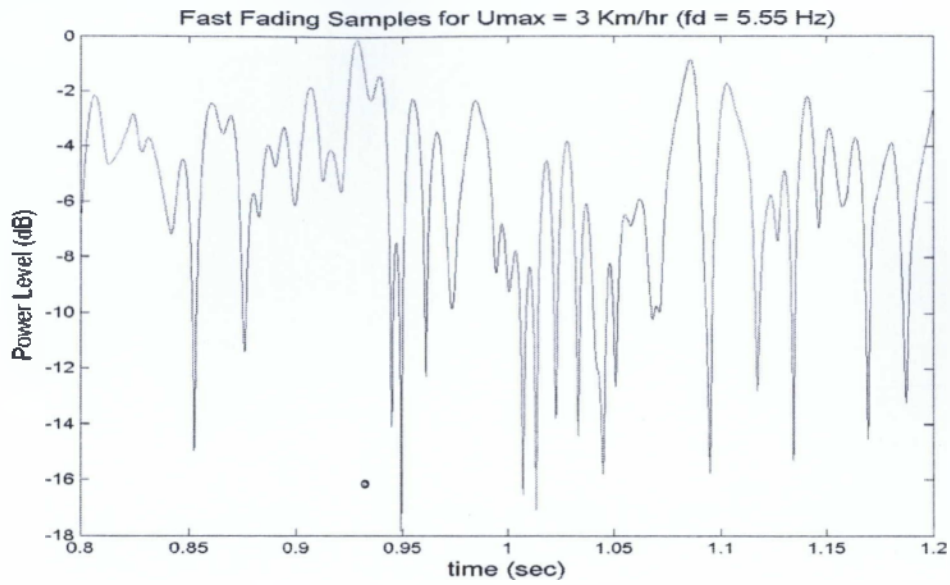
Σχήμα 3.11 Περιβάλλουσα Rayleigh με διαλείψεις



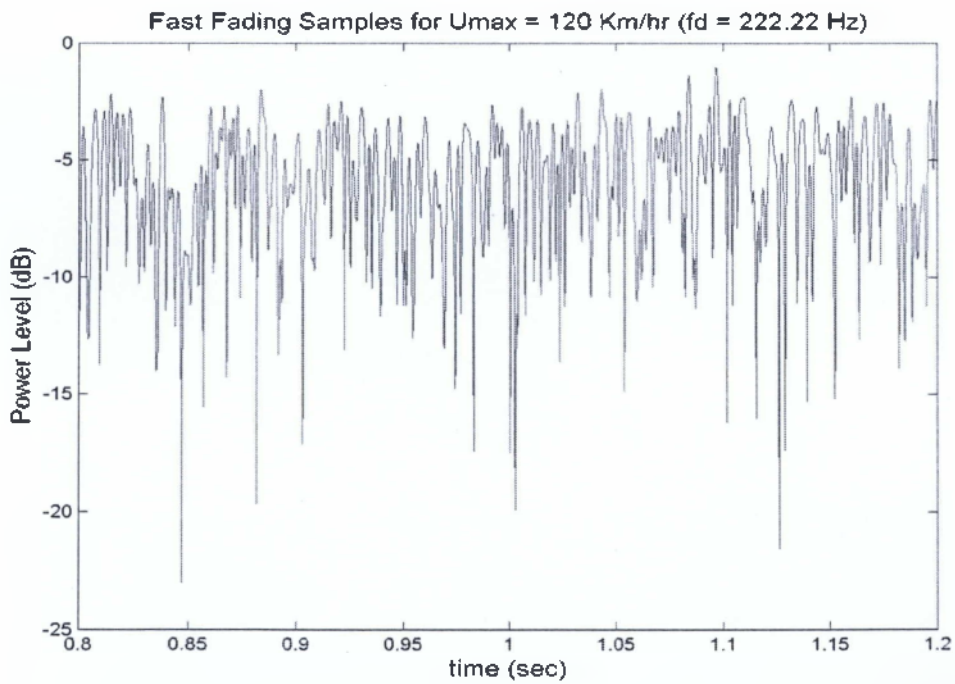
Σχήμα 3.12 Περιβάλλουσα Rice (K=6 dB) με διαλείψεις

3.5.4.4 Διαλείψεις λόγω κίνησης του δέκτη

Στις προηγούμενες ενότητες περιγράψαμε τις διαλείψεις που προκύπτουν από την πολυδιαδρομική διάδοση του σήματος και τη βασική κατηγοριοποίηση σε διάυλο Rayleigh και Rice. Στις κινητές επικοινωνίες προκαλούνται διαλείψεις, όπως αναφερθήκαμε προηγουμένως, και από την συνεχή μεταβολή της θέσης του δέκτη με την οποία έχει ως αποτέλεσμα και την συνεχή μεταβολή στα μονοπάτια διάδοσης που οδηγεί σε διαλείψεις. Για παράδειγμα, παρατηρώντας το σχήμα 3.10 τα μονοπάτια διάδοσης τη χρονική στιγμή t είναι διαφορετικά με τα μονοπάτια διάδοσης τη χρονική στιγμή $t+a$ λόγω της κίνησης του δέκτη. Η χρονική μεταβολή που δημιουργείται ορίζεται από την φασματική πυκνότητα ισχύος Doppler και η περιγραφή της γίνεται από διάφορα μοντέλα. Για την προσομοίωσή μας θα χρησιμοποιήσουμε το μοντέλο Jakes. Για την εξέταση των διαλείψεων λόγω κίνησης κύρια παράμετρος είναι η ταχύτητα του κινητού δέκτη. Στα παρακάτω σχήματα (3.13 και 3.14) δίνουμε τα αποτελέσματα της προσομοίωσης για διάυλο τύπου Rayleigh για μέγιστη ταχύτητα δέκτη $V_{\max} = 3 \text{ km/hr}$ και για $V_{\max} = 120 \text{ km/hr}$. Το συμπέρασμα που διακρίνουμε έπειτα από την προσομοίωση είναι ότι όσο αυξάνεται η μέγιστη ταχύτητα του δέκτη, τόσο ο ρυθμός διαλείψεων λόγω κίνησης θα αυξάνεται. Να επισημάνουμε ότι στη προσομοίωση για τη μελέτη επίδοσης LDPC κωδίκων συμπεριλαμβάνοντας τις διαλείψεις, όποια ανάλυση και να έχουμε Rayleigh ή Rice η μέγιστη ταχύτητα του δέκτη είναι σταθερή παράμετρος ορισμένη ως $V_{\max} = 10 \text{ km/hr}$.



Σχήμα 3.13 Διαλείψεις για $V_{max} = 3 \text{ km/hr}$



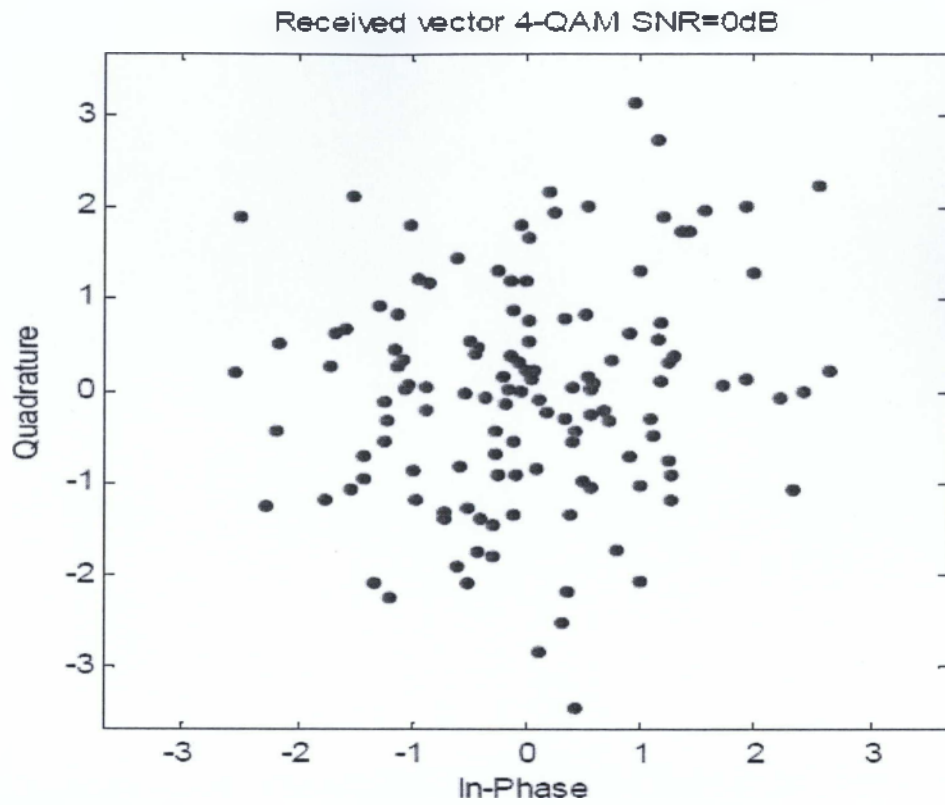
Σχήμα 3.14 Διαλείψεις για $V_{max} = 120 \text{ km/hr}$

3.5.5 Αποδιαμορφωτής

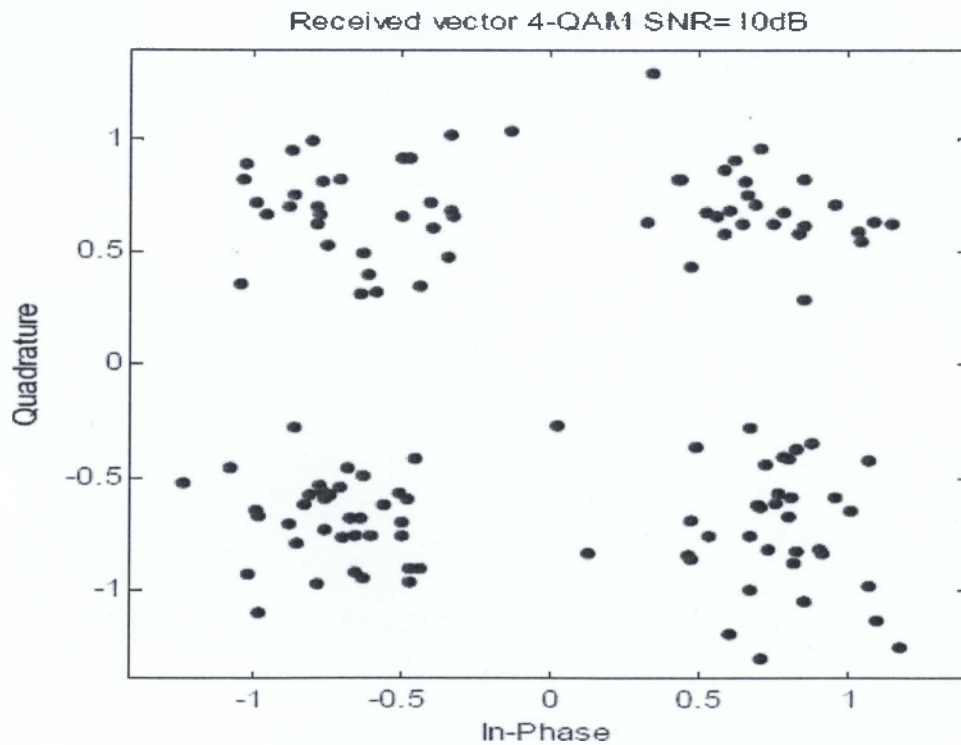
Ο αποδιαμορφωτής εκτελεί την αντίστροφη διαδικασία από τον διαμορφωτή. Στην είσοδό του λαμβάνει σύμβολα και στην έξοδο του δημιουργεί την ανάλογη ακολουθία bits. Το σχήμα διαμόρφωσης πρέπει να είναι αντίστοιχο της αποδιαμόρφωσης (δηλαδή διαμόρφωση PSK, αποδιαμόρφωση PSK). Στην ενότητα όπου αναφερθήκαμε για τον διαμορφωτή, είδαμε ότι έπειτα από την διαδικασία της διαμόρφωσης εκπέμπονται σύμβολα, στα οποία το καθένα αναλογεί σε μια από τις επιτρεπόμενες τιμές κάθε σχήματος διαμόρφωσης. Ύστερα από την διαδικασία της μετάδοσης, τα σήματα λαμβάνονται αλλοιωμένα εξαιτίας των επιβαρύνσεων που έχει εισάγει ο δίαυλος. Στόχος του αποδιαμορφωτή είναι για κάθε λαμβανόμενο σύμβολο να εντοπίσει ποιο πλησιάζει τα αρχικά σύμβολα και να κάνει την αντιστοίχηση. Ως ευκλείδεια απόσταση στο μιγαδικό επίπεδο των συμβόλων χαρακτηρίζεται η έννοια της απόστασης. Επομένως, αν η αλλοίωση είναι μεγάλη, υπάρχει περίπτωση ο αποδιαμορφωτής να εκτελέσει εσφαλμένη εκτίμηση για το σύμβολο που στάλθηκε. Έπειτα από την εκτίμηση για το κάθε σύμβολο δημιουργείται μια ακολουθία ψηφίων που αναλογεί στο σύμβολο που καταλήγει μετά στον αποκωδικοποιητή.

Την εκτίμηση των συμβόλων θα την παρουσιάσουμε γραφικά χρησιμοποιώντας σχήματα από την προσομοίωση που εκτελέσαμε. Στα ακόλουθα σχήματα εισάγουμε ληφθέντα σύμβολα τα οποία αναλογούν σε διαμόρφωση 4-QAM και σηματοθορυβικό λόγο 0 dB (σχήμα 3.15) και 10 dB (σχήμα 3.16). Ενημερωτικά να σημειώσουμε, ότι το συνολικό πλήθος των ληφθέντων συμβόλων αναλογεί σε μια κωδική λέξη (256bits) του κώδικά μας, άρα το πλήθος τους είναι $(256/\log_2 4)=128$. Παρατηρώντας, ότι για SNR 0 dB και σύμφωνα με το σχήμα 3.8 έχουμε σημαντική αλλοίωση στις θέσεις των συμβόλων στο μιγαδικό επίπεδο, το οποίο θα έχει σαν αποτέλεσμα τον εντοπισμό πολλαπλών σφαλμάτων και την μετέπειτα δύσκολη εργασία του αποκωδικοποιητή. Ενώ για SNR 10 dB τα σύμβολα εμφανίζονται σε περιοχές πιο αναγνωρίσιμες και κοντά από τα αρχικά σύμβολα.

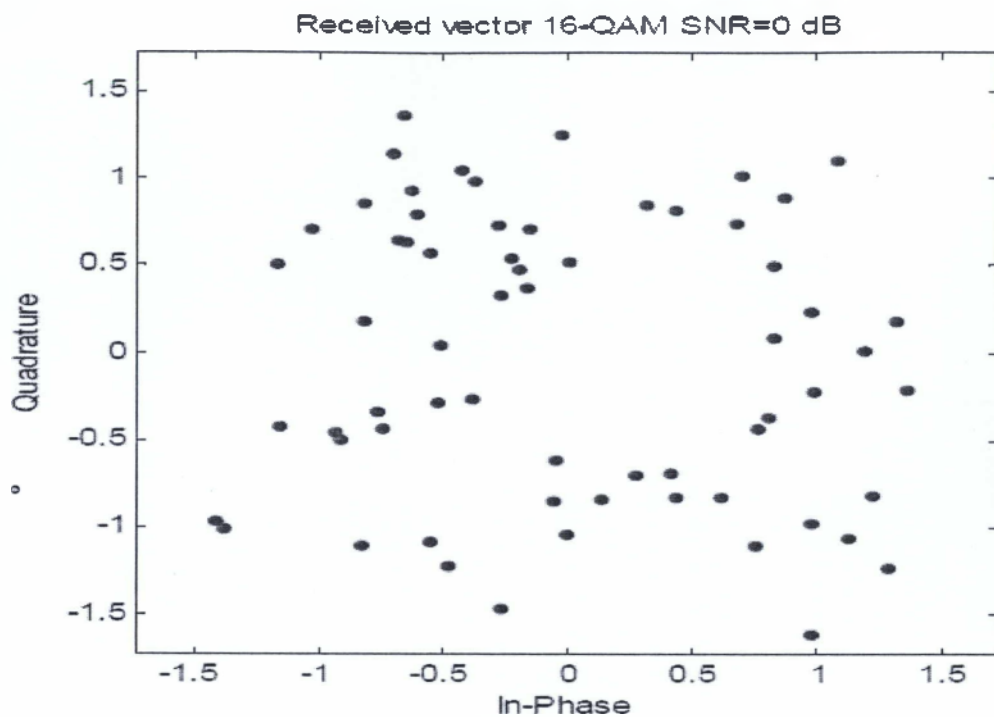
Επομένως, αν τυχόν δημιουργηθεί κάποια αλλοίωση στο σύμβολο έπειτα από την διαδικασία της μετάδοσης, ο αποκωδικοποιητής θα εκτελέσει περισσότερες φορές ορθή εκτίμηση επειδή το ληφθέν σύμβολο εμφανίζεται πιο κοντά στο αρχικό σύμβολο παρά στα υπόλοιπα.



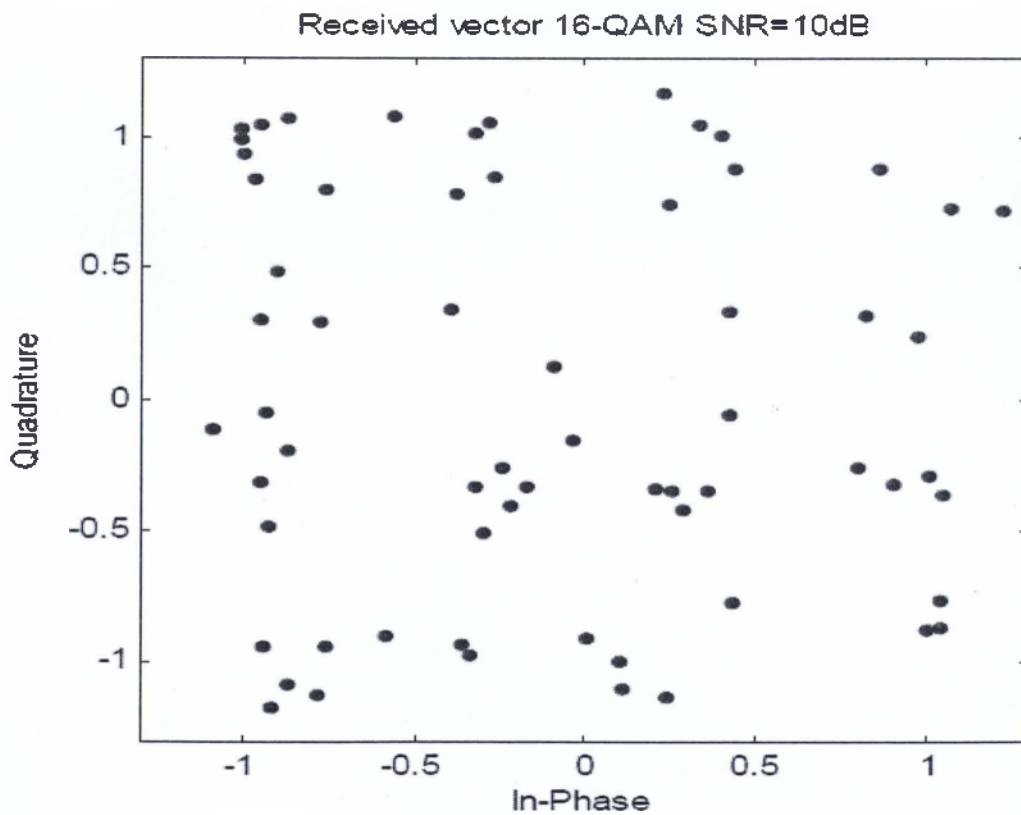
Σχήμα 3.15 Ληφθέντα σύμβολα για 4-QAM και σηματοθορυβικό λόγο 0 dB



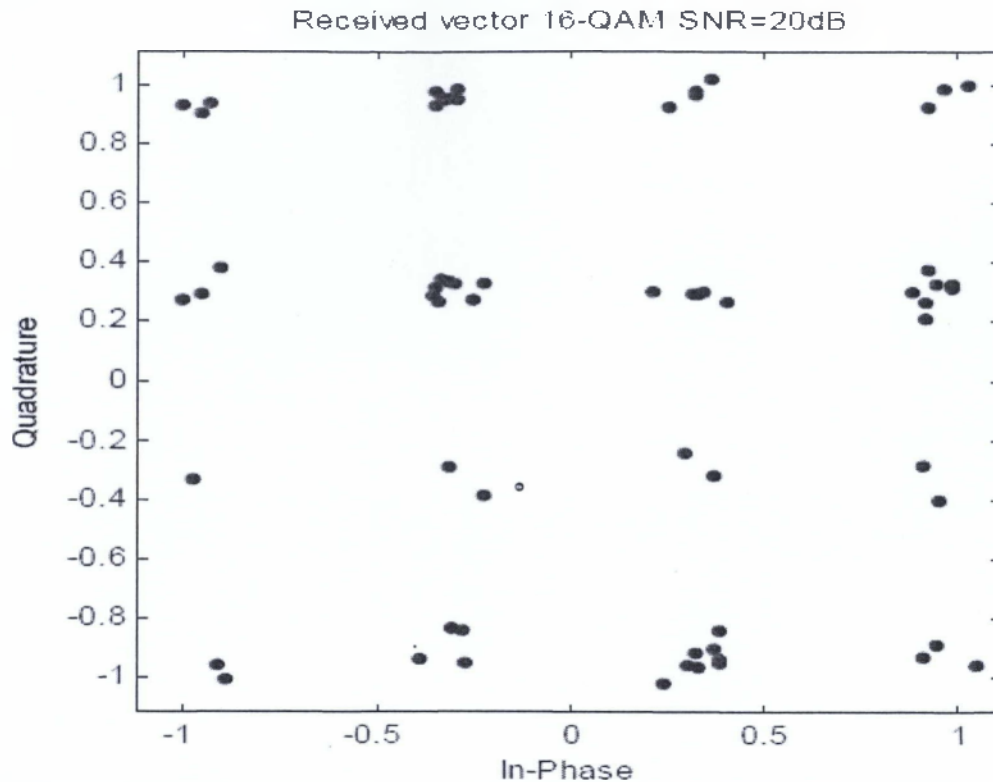
Σχήμα 3.16 Ληφθέντα σύμβολα για 4-QAM και σηματοθορυβικό λόγο 10 dB



Σχήμα 3.17 Ληφθέντα σύμβολα για 16-QAM και σηματοθορυβικό λόγο 0 dB



Σχήμα 3.18 Ληφθέντα σύμβολα για 16-QAM και σηματοθορυβικό λόγο 10 dB



Σχήμα 3.19 Ληφθέντα σύμβολα για 16-QAM και σηματοθορυβικό λόγο 20 dB

Ομοίως και στα σχήματα 3.17-3.19 (διαμόρφωση 16-QAM) παρατηρούμε τα ίδια συμπεράσματα, διότι με την βελτίωση του σηματοθορυβικού λόγου οι περιοχές που καταλαμβάνουν τα ληφθέντα σύμβολα γίνονται και πιο αναγνωρίσιμες γύρω από τα αρχικά σύμβολα καταλήγοντας έτσι τον αποκωδικοποιητή να έχει λιγότερα ποσοστά εσφαλμένης εκτίμησης.

Παρατηρώντας τις διαμορφώσεις 4-QAM και 16-QAM διαπιστώνουμε ότι η 4-QAM εμφανίζει καλύτερες ιδιότητες απόστασης, δηλαδή με το ίδιο σηματοθορυβικό λόγο ένα σύμβολο 4-QAM έχει λιγότερες πιθανότητες να μεταφερθεί σε περιοχή άλλου συμβόλου από ότι ένα σύμβολο διαμόρφωσης 16-QAM. Σύμφωνα με το παράδειγμά μας χρησιμοποιώντας διαμόρφωση 16-QAM το κέρδος είναι η καλύτερη φασματική απόδοση. Τα ληφθέντα σύμβολα που απεικονίζονται στα σχήματα και στην πρώτη περίπτωση και στην δεύτερη αναλογούν σε μια κωδική λέξη. Στην περίπτωση 4-QAM ο αριθμός των συμβόλων είναι $(256 / \log_2 4) = 128$, ενώ στην περίπτωση 16-QAM ο αριθμός των συμβόλων είναι $(256 / \log_2 16) = 64$, άρα υποδιπλασιάζεται το καταναλισκόμενο εύρος ζώνης. Η σύγκριση που έγινε σε αυτή τη παράγραφο μπορεί να γενικευτεί για διαμορφώσεις χαμηλής και υψηλής φασματικής απόδοσης.

Γενικά στα σύγχρονα τηλεπικοινωνιακά συστήματα χρησιμοποιείται η τεχνική της προσαρμοστικής διαμόρφωσης. Στόχος της είναι να αξιοποιήσει τα θετικά και τα δύο κατηγοριών διαμόρφωσης, χαμηλής και υψηλής φασματικής απόδοσης. Η προσαρμοστική διαμόρφωση προτείνεται για χρήση διαμορφώσεων υψηλής φασματικής απόδοσης κατά τη διάρκεια ευνοϊκών συνθηκών στο διάυλο και μετάβαση σε περισσότερο ανθεκτικά / εύρωστα σχήματα μικρότερης φασματικής απόδοσης κατά τη διάρκεια δυσμενών συνθηκών στο διάυλο.

3.5.6 Αποκωδικοποιητής

Από τον αποδιαμορφωτή φεύγουν τα bits και καταλήγουν στην είσοδο του αποκωδικοποιητή LDPC(256,128), ο οποίος τα λαμβάνει σε πακέτα των 256 bits ίδιο το μήκος της κωδικής λέξης του κώδικα που έχουμε ορίσει. Κάθε κωδική λέξη αποκωδικοποιείται μόνη της. Στον τερματισμό της διαδικασίας της αποκωδικοποίησης απομονώνονται τα bits της κωδικής λέξης που αναλογούν στο μήνυμα πληροφορίας και σύμφωνα με τον κώδικά μας είναι 128.

Ο κύριος στόχος της διαδικασίας της αποκωδικοποίησης θα περιγραφεί αναλυτικά στο κεφάλαιο 4^ο στο οποίο θα αναφερθούμε πιο αναλυτικά στους LDPC κώδικες. Γενικά όμως ο αλγόριθμος αποκωδικοποίησης LDPC κωδικών εντοπίζει την καλύτερη πιθανοτικά εκτίμηση για κάθε bit της κωδικής λέξης, όχι όμως και την καλύτερη εκτίμηση για την κωδική λέξη ως σύνολο.

3.5.7 Εκτίμηση Λαθών

Ύστερα από την διαδικασία της αποκωδικοποίησης του κάθε ληφθέντος κωδικοποιημένου μηνύματος, το σύστημα απομονώνει το μήνυμα πληροφορίας. Ο στόχος του αποδιαμορφωτή και του αποκωδικοποιητή είναι το μήνυμα να ταυτίζεται με το αρχικό μήνυμα, κάτι το οποίο δεν συμβαίνει πάντα. Ήδη έχουμε αναφερθεί για παράδειγμα στους περιορισμούς πάνω στη διορθωτική ικανότητα των κωδικών. Ο εκτιμητής λαθών συγκρίνει το αρχικό μήνυμα πληροφορίας στον πομπό με το τελικό μήνυμα στο δέκτη καταγράφοντας τα σφάλματα κατά την διαδικασία της μετάδοσης. Για την ορθή εκτίμηση της πιθανότητας λάθους, όπως αναφερθήκαμε και σε άλλη ενότητα θα πρέπει να συγκεντρώσουμε την μέση τιμή σφαλμάτων που έχουν καταγραφεί από την προσομοίωση πολλών πακέτων. Τέλος, θα πρέπει να

επισημάνουμε ότι ο εκτιμητής λαθών στο δέκτη δεν μπορεί να υπάρξει σε πραγματικό σύστημα διότι ο δέκτης δεν έχει την ικανότητα να γνωρίζει με σιγουριά το αρχικό μήνυμα πληροφορίας.

4 Τεχνικές Κωδικοποίησης LDPC (Low Density Parity Check)

4.1 Γενικά

Στο κεφάλαιο αυτό θα μας απασχολήσουν οι κώδικες LDPC (Low Density Parity Check), που είναι από τους πιο καρποφόρους κώδικες διόρθωσης σφαλμάτων. Το κύριο στοιχείο αυτών των κωδίκων είναι ότι αφορούν γραμμικούς συμπαγείς κώδικες οι οποίοι δημιουργούνται με βάση τον αραιό πίνακα ελέγχου ισοτιμίας \mathbf{H} . Τώρα, σχετικά με την δυαδική περίπτωση (binary case), ο αραιός πίνακας είναι ο πίνακας \mathbf{H} ο οποίος συνίσταται από μικρό αριθμό άσσων που είναι διασκορπισμένοι μεταξύ των μηδενικών ή διαφορετικά υπάρχει χαμηλή πυκνότητα (low – density) από άσσους. Η αποδοτικότητα των κωδίκων LDPC έχει άμεση σχέση με την σχεδίαση του πίνακα \mathbf{H} και τον αλγόριθμο κωδικοποίησης που χρησιμοποιείται. Γενικά, η αποκωδικοποίηση των κωδίκων LDPC βασίζεται στον αλγόριθμο belief – propagation που δουλεύει επαναληπτικά, δηλαδή σε κάθε του βήμα αναμορφώνει τη πιθανότητα κάθε bit να είναι 1 ή 0, εφόσον υπολογίζει την πληροφορία που πηγάζει από την ανάλογη πιθανότητα των άλλων bits.

Από ιστορική άποψη, το 1962 προτάθηκε για πρώτη φορά η συγκεκριμένη κατηγορία κωδίκων με τα προαναφερθέντα χαρακτηριστικά από τον Gallager. Η περιπλοκότητα του επαναληπτικού αλγορίθμου όμως υπερίσχυε από τις ήδη υπάρχουσες δυνατότητες των ηλεκτρονικών επεξεργασιών. Για τον λόγο αυτό οι κώδικες LDPC ήταν ξεχασμένοι μέχρι και το 1996, ενώ το 1981 είχε γίνει μία προσπάθεια από τον Tanner να τους επαναφέρει στο προσκήνιο, το οποίο και κατάφεραν ο Neal και ο MacKay και πλέον αναφερόμαστε σε κώδικες που φτάνουν το θεωρητικό όριο του Shannon που έχει να κάνει με την απόδοση των σχημάτων για την αποκατάσταση των σφαλμάτων, το οποίο ισχύει για τους κώδικες turbo και LDPC. Επίσης, έχει αποδειχτεί πως οι σωστά σχεδιασμένοι κώδικες LDPC υπερβαίνουν την απόδοση των κωδίκων turbo οι οποίοι χρησιμοποιούνται από τα συστήματα κινητών επικοινωνιών 3^{ης} γενιάς (3G).

Η πολύ καλή επίδοση των κωδίκων αυτών έχει οδηγήσει στον ενστερνισμό τους από αρκετά πρότυπα επικοινωνιών από τα οποία τα πιο χαρακτηριστικά είναι τα DVB – S2, IEEE80216.e, IEEE802.3an(10BASE – T) κ.α. . Ακόμα, προβλέπεται να χρησιμοποιηθούν και σε άλλα σύγχρονα συστήματα επικοινωνιών, όπως είναι τα

συστήματα πολλαπλών εισόδων – πολλαπλών εξόδων (MIMO), τα οποία είναι ικανά να αυξήσουν την χωρητικότητα σε ικανοποιητικό βαθμό. Βέβαια, σε οποιαδήποτε περίπτωση και σε σύγκριση με την δεδομένη αποδοτικότητα που έχουν οι κώδικες LDPC, είναι αναγκαίο να πραγματοποιείται ειδικός έλεγχος για διάφορους περιοριστικούς παράγοντες, όπως είναι η περιπλοκότητα, οι απαιτήσεις της μνήμης, ο μέγεθος του κώδικα και η συμβατότητα του ρυθμού. Επομένως, έχουν προταθεί κάποιες μέθοδοι για την δημιουργία και κάποιες αλλαγές του βασικού αλγορίθμου αποκωδικοποίησης. Παρακάτω, θα παρουσιάσουμε τα κύρια χαρακτηριστικά των κωδίκων LDPC και στο 5^ο κεφάλαιο θα παρουσιάσουμε τα αποτελέσματα από την προσομοίωση που πραγματοποιήσαμε, της οποίας ο στόχος είναι να μελετηθεί η επίδοσή τους.

4.2 Κωδικοποίηση LDPC

4.2.1 Περιγραφή LDPC κωδίκων

Οι κώδικες LDPC ως επί το πλείστον σχεδιάζονται ως δυαδικοί και γραμμικοί συμπαγείς κώδικες. Σ' αυτή τη περίπτωση, ο πίνακας \mathbf{G} ο οποίος είναι ο γεννήτορας πίνακας, τροποποιεί το μήνυμα \mathbf{m} με μήκος \mathbf{k} στο κωδικοποιημένο διάνυσμα \mathbf{c} με τη βοήθεια του πολλαπλασιασμού των πινάκων στο δυαδικό πεδίο, το οποίο και έχει μήκος \mathbf{n} . Δηλαδή ισχύει $\mathbf{c} = \mathbf{m} \circ \mathbf{G}$, ούτως ώστε το κωδικοποιημένο διάνυσμα \mathbf{c} να ικανοποιεί την εξίσωση συνδρόμου, η οποία είναι $\mathbf{c} \circ \mathbf{H}^T = \mathbf{0}$. Απολύτως ισοδύναμη είναι και η περίπτωση κωδικοποίησης κατά την οποία ισχύει $\mathbf{c} = \mathbf{G}^T \circ \mathbf{m}$ και $\mathbf{H} \circ \mathbf{c} = \mathbf{0}$.

Όπως ήδη έχει αναφερθεί, οι κώδικες LDPC σχεδιάζονται μέσω κατάλληλης κατασκευής του πίνακα \mathbf{H} . Βλέπουμε ότι υπάρχουν δύο μεγάλες κατηγορίες με βάση τον συγκεκριμένο πίνακα: οι κανονικοί κώδικες (regular) και οι μη – κανονικοί κώδικες (irregular). Η πρώτη κατηγορία χαρακτηρίζεται από τον σταθερό αριθμό μηδενικών και άσσων που έχουν ανά γραμμή και ανά στήλη σε αντίθεση με τη δεύτερη κατηγορία η οποία δεν πληροί αυτή τη προϋπόθεση. Ο Gallager πρότεινε τη πρώτη κατηγορία για πρώτη φορά όπου και ανέδειξε τις ιδιότητες των κωδίκων αυτών: για αυξανόμενο μέγεθος κώδικα η πιθανότητα σφάλματος μειώνεται εκθετικά και η αύξηση του μεγέθους του κώδικα προκαλεί αύξηση της ελάχιστης απόστασης.

4.2.2 Κανονικοί (regular) LDPC κώδικες

Οι κώδικες οι οποίοι έχουν σταθερό αριθμό άσων ανά γραμμή u και σταθερό αριθμό άσων ανά στήλη s , σε σχέση με τον πίνακα \mathbf{H} ονομάζονται κανονικοί κώδικες (regular) και συμβολίζονται $\mathbf{C}_{LDPC}(n, s, u)$. Εξαιτίας της αραιότητας του πίνακα \mathbf{H} , οι παράγοντες s και u είναι μικροί αριθμοί σε σύγκριση με το μήκος που έχει ο κώδικας n .

Προκειμένου να κατανοήσουμε καλύτερα τα όσα αναφέραμε, παραθέτουμε σαν παράδειγμα τον πίνακα \mathbf{H} ενός $\mathbf{C}_{LDPC}(8, 2, 4)$ κώδικα.

$$\mathbf{H} = \begin{bmatrix} 1 & 0 & 0 & 1 & 0 & 0 & 1 & 1 \\ 0 & 1 & 1 & 0 & 1 & 1 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 1 & 1 & 0 & 1 & 0 & 1 & 0 \end{bmatrix}$$

Γνωρίζοντας ότι ο πίνακας \mathbf{H} έχει διαστάσεις $(n - k) \times n$ και στην περίπτωση που όλες οι γραμμές του είναι μεταξύ τους γραμμικώς ανεξάρτητες, ισχύει η παρακάτω σχέση για τον ρυθμό κώδικα k/n ο οποίος προκύπτει από τις δύο διαφορετικές εκφράσεις για τον συνολικό αριθμό άσων που βρίσκονται στον πίνακα:

$$ns = (n - k)u \Rightarrow \frac{k}{n} = \frac{u-s}{u} \quad (4.1)$$

Στην περίπτωση όμως που υφίστανται γραμμικώς εξαρτημένες γραμμές, ο ρυθμός κώδικα δίνεται από την εξής σχέση:

$$code_rate = \frac{n-s'}{n} \quad (4.2)$$

Όπου με s' ορίζεται η πραγματική διάσταση του υποχώρου γραμμών του πίνακα \mathbf{H} . Μέχρι τώρα, οι περιορισμοί που είχαμε θέσει στους κανονικούς κώδικες LDPC δεν οδηγούσαν στη σχεδίαση ενός πίνακα \mathbf{H} σε συστηματική μορφή. Επομένως, στις περισσότερες περιπτώσεις είναι αναγκαία η χρήση της μεθόδου απαλοιφής Gauss με σκοπό να γίνει η μετατροπή σε συστηματικό πίνακα ελέγχου ισοτιμίας, δηλαδή να έχει τη μορφή: $\mathbf{H}' = [\mathbf{I}_{n-k} \mathbf{P}^T]$, όπου με \mathbf{I}_{n-k} ορίζουμε τον μοναδιαίο πίνακα μεγέθους $(n - k) \times (n - k)$. Ο πίνακας ελέγχου ισοτιμίας του κώδικα είναι ο αρχικά σχεδιασμένος \mathbf{H} και ο γεννήτορας πίνακας \mathbf{G} έχει την ακόλουθη μορφή που προκύπτει από τον \mathbf{H}' : $\mathbf{G} = [\mathbf{P} \mathbf{I}_k]$.

Αν αναλύσουμε τη σχεδιαστική πορεία ενός κώδικα LDPC, πρώτα έχουμε την κατασκευή ενός αραιού πίνακα ελέγχου ισοτιμίας $\mathbf{H}=[\mathbf{A} \ \mathbf{B}]$ που ακολουθεί τους περιορισμούς που προκύπτουν σχεδιαστικά. Στη γενική περίπτωση, ο συγκεκριμένος πίνακας δεν είναι σε συστηματική μορφή. Οι υποπίνακες \mathbf{A} και \mathbf{B} είναι αμφοτέροι αραιοί. Ο υποπίνακας \mathbf{A} έχει διαστάσεις $(n - k) \times (n - k)$ και πρέπει να είναι αντιστρέψιμος με αντίστροφο \mathbf{A}^{-1} , επομένως ο \mathbf{B} πρέπει να έχει διαστάσεις $(n - k) \times k$. Η μέθοδος απαλοιφής Gauss, πάνω στο δυαδικό πεδίο (binary field), μετατρέπει τον πίνακα $\mathbf{H}=[\mathbf{A} \ \mathbf{B}]$ στη μορφή $\mathbf{H}' = [\mathbf{I}_{n-k} \ \mathbf{A}^{-1} \mathbf{B}] = [\mathbf{I}_{n-k} \mathbf{P}^T]$. Εφόσον έχει σχηματιστεί ο ισοδύναμος πίνακας ελέγχου ισοτιμίας \mathbf{H}' , ο ανάλογος γεννήτορας πίνακας \mathbf{G} μπορεί να σχηματιστεί με τη βοήθεια των υποπινάκων \mathbf{H}' και να δώσει $\mathbf{G} = [\mathbf{P} \ \mathbf{I}_k]$. Το τελευταίο βήμα είναι επακόλουθο της βασικής σχέσης η οποία συνδέει τον γεννήτορα πίνακα με τον πίνακα ελέγχου ισοτιμίας στη θεωρία της κωδικοποίησης:

$$\mathbf{G} \circ \mathbf{H}' = \mathbf{0}$$

Μετά από όσα αναφέραμε, ορίσαμε και τον πίνακα ελέγχου ισοτιμίας καθώς επίσης και τον γεννήτορα πίνακα, επομένως ο κώδικας LDPC σχεδιαστεί, στο σημείο αυτό, να αναφέρουμε ότι οι πίνακες που μας ενδιαφέρουν είναι οι \mathbf{H} και \mathbf{G} με τον \mathbf{H}' να έχει βοηθητικό ρόλο.

Αναλόγως με την τακτική που χρησιμοποιείται για τη δημιουργία του πίνακα \mathbf{H} , οι κώδικες LDPC διαχωρίζονται σε δομημένους και τυχαίους. Γενικότερα, μπορεί οι τυχαίοι κώδικες να εμφανίζουν καλύτερες επιδόσεις σε σχέση με το ποσοστό σφαλμάτων BER αλλά έχουν πιο περίπλοκη κωδικοποίηση.

4.2.3 Μη-Κανονικοί (Irregular) LDPC κώδικες

Όταν ο πίνακας \mathbf{H} δεν έχει σταθερό αριθμό μονάδων ανά γραμμή ή ανά στήλη, τότε οι κώδικες LDPC ονομάζονται μη – κανονικοί. Χαμηλότερες τιμές BER εμφανίζουν οι μη – κανονικοί κώδικες οι οποίοι έχουν σχεδιαστεί κατάλληλα σε σχέση με τους κανονικούς κώδικες. Για τη σχεδίαση αυτής της κατηγορίας υπάρχουν διάφορες μέθοδοι.

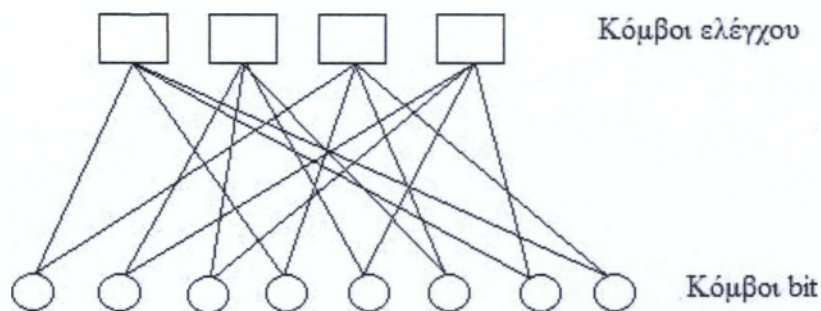
4.3 Αποκωδικοποιητής LDPC

4.3.1 Ο γράφος Tanner

Ο γράφος Tanner είναι ένα επωφελές εργαλείο για την αναπαράσταση ενός LDPC κώδικα. Είναι ένας γράφος αποτελούμενος από δύο μέρη και απεικονίζει εξισώσεις ισοτιμίας (parity – equations) οι οποίες δημιουργούνται από τον πίνακα ισοτιμίας \mathbf{H} . Η ονομασία του πίνακα ελέγχου ισοτιμίας \mathbf{H} προέρχεται από το γεγονός ότι κάνει $m = n - k$ διαφορετικούς ελέγχους ισοτιμίας για κάθε κωδικοποιημένη λέξη που λαμβάνεται.

Ο γράφος Tanner ο οποίος αντιστοιχεί σε έναν κώδικα (n, k) , συνίσταται από n κόμβους bits (bit nodes), από $m = n - k$ κόμβους ελέγχου (check nodes) καθώς επίσης και από έναν αριθμό ακμών ανάμεσα στις δύο κατηγορίες κόμβων. Ένα bit κωδικοποιημένης λέξης απεικονίζεται από ένα bit κόμβου και ένας έλεγχος ισοτιμίας αναπαρίσταται από έναν κόμβο ελέγχου. Αν η ανάλογη θέση του \mathbf{H} έχει «1», τότε υπάρχει ακμή ανάμεσα στον κόμβο bit και στον κόμβο ελέγχου. Πιο τυπικά, για να υφίσταται η σύνδεση ανάμεσα στον κόμβο ελέγχου i και στον κόμβο bit j πρέπει να ισχύει $H_{ij}=1$.

Παρακάτω, διακρίνεται ο γράφος Tanner για τον πίνακα \mathbf{H} της ενότητας 4.2.2

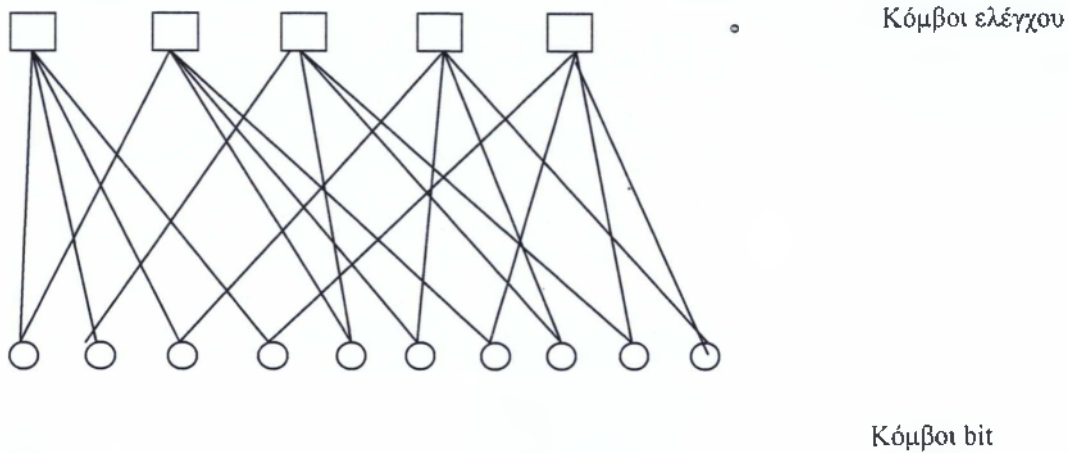


Σχήμα 4.1 Γράφος Tanner για τον πίνακα \mathbf{H} της ενότητας 4.2.2

Στην ουσία, όπως αναφέραμε, αφορά τη γραφική απεικόνιση των εξισώσεων ισοτιμίας του πίνακα που είναι να αναλύσουμε, οι οποίες και είναι οι ακόλουθες:

$$\begin{aligned}
c_1 \oplus c_4 \oplus c_7 \oplus c_8 &= 0 \\
c_2 \oplus c_3 \oplus c_5 \oplus c_6 &= 0 \\
c_1 \oplus c_4 \oplus c_6 \oplus c_8 &= 0 \\
c_2 \oplus c_3 \oplus c_5 \oplus c_7 &= 0
\end{aligned}$$

Στον γράφο αυτό, ο κύκλος μήκους ν ορίζεται ως ένα μονοπάτι που περιέχει ν διαφορετικές ακμές καθώς επίσης αρχίζει και τελειώνει στον ίδιο κόμβο. Ως girth ορίζεται επίσης το ελάχιστο μήκος κύκλου. Παρακάτω, ακολουθεί ένα παράδειγμα για girth ίσο με 6.



Σχήμα 4.2 Γράφος Tanner ενός LDPC κώδικα (10, 5) με girth ίσο με 6

Γενικά, είναι εμφανές πως η χαμηλότερη τιμή που μπορεί να πάρει το girth σε έναν γράφο Tanner είναι 4, όπου στη συγκεκριμένη περίπτωση τα τέσσερα «1» είναι στις γωνίες του υποπίνακα του \mathbf{H} όπως διακρίνουμε και στο σχήμα παρακάτω.



Σχήμα 4.3 Παράδειγμα πίνακα ελέγχου ισοτιμίας με κύκλο μήκους 4

Όταν συνδυάζουμε τον αλγόριθμο κωδικοποίησης belief - propagation με τους κώδικες LDPC έχουμε τις βέλτιστες επιδόσεις των τελευταίων, όπου η πιθανότητα κάθε bit να διαδίδεται μέσω των ακμών και χρησιμοποιείται ως εξωτερική πληροφορία για την ανανέωση της πιθανότητας άλλων bits. Έχει αποδειχθεί ότι όσο

πιο «εξωτερική» είναι αυτή η πληροφορία, τόσο αυξάνει και η επίδοση του κώδικα. Στην περίπτωση που το μήκος κύκλου είναι μικρό, τότε η πληροφορία γυρίζει πιο σύντομα στον αρχικό κόμβο περιλαμβάνοντας μικρό μέρος εξωτερικής πληροφορίας, κάτι το οποίο δεν συμβάλλει στην απόδοση του κώδικα. Επομένως, το ελάχιστο μήκος κύκλου είναι ένας σημαίνων παράγοντας για τους κώδικες LDPC και υφίσταται αξιόλογη προσπάθεια για τη δημιουργία κωδίκων με αυξημένο girth.

Σύμφωνα με τα παραπάνω οδηγούμαστε στο συμπέρασμα ότι η καλύτερη περίπτωση θα ήταν γράφος δενδρικής δομής δηλαδή χωρίς κύκλους. Στην συγκεκριμένη περίπτωση ο belief-propagation (BP) που θα αναφερθούμε πιο αναλυτικά στην συνέχεια τερματίζει ύστερα από πεπερασμένο αριθμό επαναλήψεων. Γενικά όμως, κώδικες με (ή ακυκλικούς) γράφους (ή χωρίς κύκλους) παρουσιάζουν χαμηλές τιμές BER λόγω της μικρής ελάχιστης απόστασης τους. Η ελάχιστη απόστασή τους είναι δύο για ρυθμούς κώδικα $R > 1/2$ ($k/n > 1/2$). Με αυτόν τον τρόπο, η εμφάνιση κύκλων σε αποδοτικούς LDPC κώδικες είναι επιτακτική. Πρέπει όμως να γίνεται η αφαίρεση κύκλων μικρού μήκους για παράδειγμα 4, 6, 8 κλπ ή να γίνεται κάποια σημαντική ελάττωσή τους. Κλείνοντας την ανάλυση μας για την σχέση του γράφου Tanner ενός LDPC κώδικα με τις ιδιότητες που παρουσιάζει, πρέπει να σημειώσουμε ότι η αρνητική επιρροή κύκλων μικρού μήκους μειώνεται για αυξανόμενο μέγεθος κώδικα και μειώνεται δραστικά για μεγάλα μεγέθη κώδικα (>1000 bits).

4.3.2 Ο αλγόριθμος BP

Στην ενότητα αυτή θα ασχοληθούμε με τον κύριο αλγόριθμο αποκωδικοποίησης LDPC κωδίκων belief-propagation (BP), κοινός και ως sum-product. Γενικά το μεγαλύτερο μέρος των εν χρήσει αλγορίθμων αποκωδικοποίησης LDPC κωδίκων είναι απλοποιήσεις ή παραλλαγές του κύριου αλγορίθμου, με τον οποίο θα περιγράψουμε παρακάτω.

Σκοπός του αλγορίθμου είναι να καθορίσει με τον καλύτερο τρόπο την εκ των υστέρων (a posteriori) πιθανότητα κάθε bit να είναι 0 ή 1, έχοντας γνωστά το σήμα που έχει λάβει ο δέκτης, τα χαρακτηριστικά του κώδικα που εδώ εκφράζονται ως εξισώσεις ισοτιμίας καθώς επίσης και τα χαρακτηριστικά του τηλεπικοινωνιακού διαύλου της διαδρομής από τον πομπό στο δέκτη. Εδώ να επισημάνουμε, ότι ο αλγόριθμος belief-propagation έχει την ικανότητα να εντοπίζει την βέλτιστη εκτίμηση

για κάθε bit της κωδικής λέξης που φθάνει, όμως όχι την βέλτιστη εκτίμηση για την συνολική κωδική λέξη. Συνεπάγεται ότι είναι ένας εκ των υστέρων αλγόριθμος σε σχέση με τους αλγόριθμους μέγιστης πιθανοφάνειας, όπως ο αλγόριθμος Viterbi, που βελτιώνει την αποκωδικοποίηση ολόκληρου του ληφθέντος κωδικοποιημένου διανύσματος.

Σε πρώτο στάδιο, γίνεται ενημέρωση των τιμών Q_i^x και θέτονται ίσες εκ των προτέρων εκτιμήσεις για τα ληφθέντα bits. Με f_j^x ορίζεται η πιθανότητα j-οστό bit να είναι $x=0$ ή $x=1$. Οι παραπάνω πιθανοτικές εκφράσεις εξαρτώνται από το είδος του διαύλου και από την μοντελοποίηση που έχουμε κάνει στο διάυλο. Αν για παράδειγμα έχουμε διάυλο με προσθετικό λευκό θόρυβο θα χρησιμοποιήσουμε εκφράσεις που έχουν βάση στις γκαουσιανές συναρτήσεις πυκνότητας πιθανότητας.

Μετά από το στάδιο της αρχικοποίησης, ξεκινάει η ανταλλαγή πληροφορίας ανάμεσα στους κόμβους bit και στους κόμβους ελέγχου. Για να δείξουμε την πιθανότητα να ικανοποιείται η i-οστή εξίσωση ελέγχου ισοτιμίας (h_i) γνωρίζοντας ότι ο κόμβος bit j βρίσκεται στην κατάσταση x (d_j^x) χρησιμοποιούμε το συμβολισμό R_j^x . Από την πληροφορία που στέλνεται από κάθε κόμβο ελέγχου που είναι συνδεδεμένος με τον κόμβο bit j προς τον κόμβο bit j συνεπάγεται αυτή η πιθανότητα, η οποία για να ικανοποιεί και την h_i βρίσκεται με την παρακάτω σχέση:

$$P(h_i / d_j = x) = \sum_{d: d_j = x} P(h_i / d) P(d / d_j = x) \quad (4.3)$$

Προϋποθέτοντας ότι ο κόμβος bit j είναι στην κατάσταση x , η παραπάνω πιθανότητα υπολογίζεται για όλα τα πιθανά αποκωδικοποιημένα διανύσματα d για τα οποία ικανοποιείται η i-οστή εξίσωση ελέγχου ισοτιμίας.

Η παρακάτω σχέση υπολογίζει για κάθε τιμή του x , της πληροφορίας που θα σταλθεί στον συνδεδεμένο με αυτόν κόμβο bit j για τον κόμβο ελέγχου ισοτιμίας h_i .

$$R_j^x = \sum_{d: d_j = x} P(h_i / d) \prod_{k \in N(i) \setminus j} Q_k^* \quad (4.4)$$

Στην σχέση 4.4 το $N(i)$ περιγράφει το σύνολο των δεικτών των κόμβων bit που είναι συνδεδεμένοι με τον h_i , ενώ το $N(i)/j$ περιγράφει το ίδιο σύνολο αλλά με την αφαίρεση του κόμβου bit j. Παρατηρούμε, ότι η πληροφορία που φθάνει στον κόμβο bit j είναι γνησίως εξωτερική, δηλαδή δεν είναι συνάρτηση της τρέχουσας κατάστασης του κόμβου. Η πιθανότητα $P(h_i / d)$ πρέπει να ικανοποιεί την h_i για δοθέν

διάνυσμα με 0 ή 1. Η παράμετρος Q_j^x ορίζει την πιθανότητα ο κόμβος d_j να βρίσκεται στην κατάσταση x και από τον κόμβο bit d_j στέλνεται σε όλους τους συνδεδεμένους με αυτόν με αυτόν κόμβους ελέγχου h_i . Όπως παρατηρούμε από την σχέση 4.5, μέσω της πιθανότητας R_k^x η τιμή της παραμέτρου ανανεώνεται σε κάθε στάδιο του αλγορίθμου και από την πληροφορία που έχει σταλεί στον κόμβο d_j από τους συνδεδεμένους με αυτόν κόμβους ελέγχου h_i .

$$Q_j^x = a_{ij} f_j^x \prod_{k \in M(i) \setminus j} R_k^x \quad (4.5)$$

Όπως αναφέραμε και προηγουμένως, ο συντελεστής f_j^x περιγράφει την εκ των προτέρων πιθανότητα ο κόμβος d_j να είναι στην κατάσταση x , ενώ η σταθερά a_{ij} συνεπάγεται από απαίτηση να ισχύει η κανονικοποιημένη συνθήκη (το άθροισμα όλων των πιθανοτήτων ίσο με 1):

$$\sum_x Q_j^x = 1 \quad (4.6)$$

Μετά τον υπολογισμό των πιθανοτήτων Q_j^x, R_k^x υπολείπεται να βρούμε την εκτίμηση της τιμής κάθε bit ξεχωριστά με την χρήση της παρακάτω σχέσης:

$$\hat{d}_j = \arg_x \max f_j^x \prod_{k \in M(j)} R_k^x$$

Η τιμή d_j υπολογίζεται στη δυαδική περίπτωση που για x έχουμε 0 ή 1 και σύμφωνα με το ποια τιμή θα είναι μεγαλύτερη από τις δύο λαμβάνει τιμή και το j -οστό bit (0 ή 1) του διάνυσματος προς αποκωδικοποίηση. Για να προκύψει η εκτίμηση για το διάνυσμα d , πρέπει να γίνει η παραπάνω διαδικασία για όλες τις τιμές του j . Αν $c \circ H^T = 0$, συνεπάγεται ότι το διάνυσμα d λαμβάνεται ως έγκυρο αποκωδικοποιημένο διάνυσμα και ο επαναληπτικός αλγόριθμος τελειώνει δίνοντας ως έξοδο το διάνυσμα d . Αλλιώς, η διαδικασία επαναλαμβάνεται χωρίς το στάδιο της αρχικοποίησης μέχρι να βρεθεί έγκυρο διάνυσμα ή μέχρι να φτάσει ο αλγόριθμος ένα άνω όριο επαναλήψεων. Στην τελική περίπτωση, το διάνυσμα που θα καταλήξει στην έξοδο του αποκωδικοποιητή θα είναι λανθασμένο αλλά ο αποκωδικοποιητής θα έχει εκτελέσει την καλύτερη εκτέλεση της κατάστασης για κάθε bit. Γενικά η διαχείριση του κάθε εσφαλμένου πακέτου από το σύστημα εξαρτάται από την κάθε υλοποίηση.

Ανακεφαλαιώνοντας, ο belief-propagation αλγόριθμος για αποκωδικοποίηση LDPC κωδίκων έχει την ακόλουθη γενική μορφή:

- Αρχικοποίηση: Υπολογίζονται οι εκ των προτέρων πιθανότητες κατάστασης κάθε bit με βάση τη λαμβανόμενη τιμή του σήματος που αναλογεί για κάθε bit και τα χαρακτηριστικά του διαύλου, έστω f_j^x . Αυτές οι τιμές χρησιμοποιούνται για την αρχικοποίηση των συντελεστών Q_v^x , $Q_v^x = f_j^x$. Επίσης, προσδιορίζεται ο μέγιστος δυνατός αριθμός επαναλήψεων του αλγορίθμου αποκωδικοποίησης, έστω `max_iter`.
- Επαναληπτική Αποκωδικοποίηση: ◦

Βήμα 1: Ροή πληροφορίας από κόμβους bit προς κόμβους ελέγχου.

Αποστολή προς κόμβους ελέγχου των συντελεστών Q_v^x . Ενημέρωση συντελεστών R_v^x .

Βήμα 2: Ροή πληροφορίας από κόμβους ελέγχου προς κόμβους bit

Αποστολή προς κόμβους ελέγχου των συντελεστών R_v^x . Ενημέρωση των συντελεστών Q_v^x . Ενημέρωση της πιθανότητας κατάστασης για κάθε bit. Σύμφωνα με ποια τιμή είναι πιο πιθανή, κάθε bit παίρνει την τιμή 0 ή 1, οπότε παίρνουμε μια εκτίμηση για το όλο διάνυσμα, έστω `vector_est`.

Βήμα 3: Έλεγχος εγκυρότητας

1. Εάν $vector_est \circ H^T = 0$, ο αλγόριθμος διακόπτεται και το `vector_est` θεωρείται έγκυρο διάνυσμα.
2. Αλλιώς, ο αλγόριθμος αρχίζει μια νέα επανάληψη αρχίζοντας από το Βήμα 1.
3. Αν όμως ο αριθμός επαναλήψεων καταλήξει στην τιμή `max_iter`, ο αλγόριθμος τερματίζεται.

Ο παρακάτω πίνακας δίνεται για την διευκόλυνση ανάγνωσης των συμβολισμών.

<u>Σύμβολο</u>	<u>Ερμηνεία</u>
Q_j^x	Έκφραση της πιθανότητας ο κόμβος bit j να είναι στην κατάσταση x, όπως προκύπτει από την πληροφορία που στέλνουν οι συνδεδεμένοι με τον j κόμβοι ελέγχου ισοτιμίας εξαιρουμένου του κόμβου i
R_j^x	Έκφραση της πιθανότητας να ικανοποιείται η i-οστή εξίσωση ελέγχου ισοτιμίας δεδομένου ότι ο κόμβος bit j βρίσκεται στην κατάσταση x
f_j^x	A priori πιθανότητα το j-οστό bit να είναι x

4.3.3 Ένα παράδειγμα εκτέλεσης BP αλγορίθμου

Στην συγκεκριμένη ενότητα, με βάση την παραπάνω παρουσίαση σχετικά με την κωδικοποίηση και την αποκωδικοποίηση LDPC κωδίκων θα παρουσιάσουμε ένα αριθμητικό παράδειγμα, με το οποίο θα γίνει περισσότερο κατανοητή η λειτουργία του BP αλγορίθμου.

Ο κώδικας μας χαρακτηρίζεται από τους παρακάτω πίνακες **H**, **G**

$$H = \begin{bmatrix} 0 & 1 & 0 & 1 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 1 \\ 1 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 & 1 & 0 & 0 & 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 & 1 & 0 & 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 1 & 1 \end{bmatrix}$$

$$G = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 \\ 1 & 1 & 1 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 & 1 & 1 & 0 & 1 & 0 & 0 & 0 & 1 \end{bmatrix}$$

Όπως περιγράψαμε προηγουμένως, πρωταρχική σχεδίαση γίνεται μόνο με τον \mathbf{H} , άρα ο \mathbf{G} είναι αποτέλεσμα του \mathbf{H} με διαδικασία που αναφέρεται σε προηγούμενη ενότητα. Σύμφωνα με τις διαστάσεις του $\mathbf{H}(8 \times 12)$ κάθε ακωδικοποιητό μήνυμα έχει μήκος $m = 12 - 8 = 4$ και κάθε κωδικοποιημένο διάνυσμα έχει μήκος $p=12$. Άρα ο ρυθμός κώδικα είναι $1/3$, ο οποίος αποτελεί πολύ σημαντικό μέγεθος για κάθε σχήμα κωδικοποίησης. Επίσης, ο συγκεκριμένος LDPC κώδικας είναι μη-κανονικός. Σε αυτό το σημείο να σημειώσουμε, ότι γενικά στην πράξη οι πίνακες που χρησιμοποιούνται είναι πολύ μεγαλύτεροι σε μέγεθος, αλλά στην συγκεκριμένη ενότητα μας ενδιαφέρει η κατανόηση της λειτουργίας κωδικοποίησης και αποκωδικοποίησης και όχι τόσο η αποδοτική σχεδίασή τους που θα ασχοληθούμε αργότερα στη προσομοίωση μας.

Στο παράδειγμά μας θα θεωρήσουμε $m = (0 \ 1 \ 0 \ 1)$. Το \mathbf{m} κωδικοποιείται ως \mathbf{c} , με βάση τη γνωστή μας σχέση $\mathbf{c} = \mathbf{m} \cdot \mathbf{G}$ και προκύπτει το κωδικοποιημένο διάνυσμα $\mathbf{c} = (1 \ 0 \ 1 \ 0 \ 1 \ 1 \ 0 \ 0 \ 0 \ 1 \ 0 \ 1)$. Έπειτα το κωδικοποιημένο διάνυσμα διαμορφώνεται για μετάδοση με BPSK διαμόρφωση και σε πολική μορφή μεταδίδεται το διάνυσμα $\mathbf{t} = (+1 \ -1 \ +1 \ -1 \ +1 \ +1 \ -1 \ -1 \ -1 \ +1 \ -1 \ +1)$.

Υποθέτοντας ότι το σήμα διέρχεται από διάυλο ο οποίος εισάγει προσθετικό λευκό θόρυβο Gauss με τυπική απόκλιση $\sigma = 0.8$ και στην είσοδο του αποκωδικοποιητή φθάνει το παρακάτω διάνυσμα \mathbf{r} ως αποτέλεσμα της μετάδοσης και της διαδικασίας δειγματοληψίας.

$$\mathbf{r} = (+1.2322 \ +1.5462 \ +0.9832 \ -2.3245 \ +0.8743 \ -0.9225 \ -0.5432 \ -2.4326 \\ -0.9432 \ +0.5451 \ -0.8216 \ +1.2371)$$

Αν είχε χρησιμοποιηθεί αποκωδικοποιητής σκληρής απόφασης (hard decision), το αποκωδικοποιημένο διάνυσμα θα ήταν:

$$\mathbf{d} = (1 \ 1 \ 1 \ 0 \ 1 \ 0 \ 0 \ 0 \ 0 \ 1 \ 0 \ 1)$$

Συγκρίνοντας τα διανύσματα \mathbf{c} και \mathbf{d} παρατηρούμε ότι ο διάυλος επικοινωνίας έχει εισάγει δύο λάθη στις θέσεις 2,6.

Επειδή ο διάυλος μας είναι τύπου AWGN, η έκφραση για την εκ των προτέρων πιθανότητα κατάστασης κάθε bit του λαμβανόμενου διανύσματος θα

ορίζεται από την γκαουσιανή συνάρτηση πυκνότητας πιθανότητας. Έτσι, ορίζοντας όπως και πριν με f_j^0 την πιθανότητα το bit στη θέση j του διανύσματος να είναι 0 και με f_j^1 την πιθανότητα το bit στη θέση j του διανύσματος να είναι 1, προκύπτουν οι παρακάτω σχέσεις:

$$f_j^0 = \frac{1}{\sqrt{2\pi\sigma}} e^{-(r_j+1)^2/2\sigma^2} \quad (4.7)$$

$$f_j^1 = \frac{1}{\sqrt{2\pi\sigma}} e^{-(r_j-1)^2/2\sigma^2} \quad (4.8)$$

Από τις σχέσεις 4.7 και 4.8 παρατηρούμε ότι πρέπει να γνωρίζουμε την τυπική απόκλιση σ του διαύλου. Ο δέκτης μπορεί με διάφορες τεχνικές όπως για παράδειγμα της μεθόδου ελαχιστοποίησης του τετραγωνικού σφάλματος (MMSE) να κάνει εκτίμηση της τυπικής απόκλισης σ . Επομένως στην προσομοίωσή μας θα θεωρούμε ότι το σύστημα του δέκτη γνωρίζει την τιμή της σταθεράς σ .

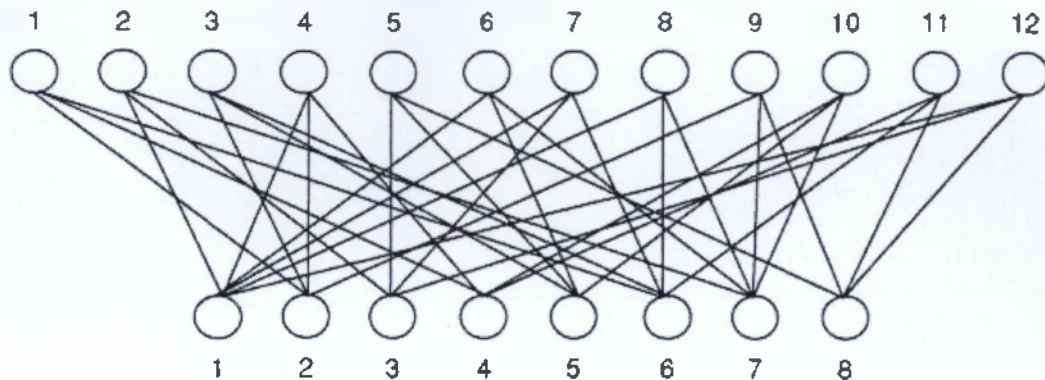
Σύμφωνα με τις παραπάνω σχέσεις για τα f_j^0 και f_j^1 , παραθέτουμε τον παρακάτω πίνακα:

1	2	3	4	5	6	7	8	9	10	11	12
+1.2322	+1.5462	+0.9832	-2.3245	+0.8743	-0.9225	-0.5432	-2.4326	-0.9432	+0.5451	-0.8216	+1.2371
+1	-1	+1	-1	+1	+1	-1	-1	-1	+1	-1	+1
0.0932	0.0032	0.0343	0.0005	0.0434	0.4992	0.1546	0.2137	0.0005	0.0654	0.3982	0.1549
0.4435	0.0235	0.3496	0.3295	0.4291	0.0265	0.2033	0.0145	0.3491	0.2964	0.0002	0.3954

Έως αυτό το σημείο βρισκόμαστε στο στάδιο της αρχικοποίησης. Για την καλύτερη κατανόηση της επαναληπτικής διαδικασίας παραθέτουμε τον γράφο Tanner του υπό εξέταση κώδικα και τις ανάλογες εξισώσεις ισοτιμίας, έπειτα από μελέτη του πίνακα **H**.

Γράφος Tanner:

Κόμβοι bit



Κόμβοι ελέγχου ισοτιμίας

Σχήμα 4.4 Γράφος Tanner για το παράδειγμά μας

Εξισώσεις ισοτιμίας

$$c_2 \oplus c_4 \oplus c_6 \oplus c_7 \oplus c_8 \oplus c_{12} = 0$$

$$c_1 \oplus c_3 \oplus c_4 \oplus c_9 = 0$$

$$c_2 \oplus c_5 \oplus c_7 \oplus c_{12} = 0$$

$$c_1 \oplus c_4 \oplus c_{10} \oplus c_{11} = 0$$

$$c_3 \oplus c_5 \oplus c_6 \oplus c_{10} = 0$$

$$c_1 \oplus c_3 \oplus c_7 \oplus c_8 \oplus c_{11} = 0$$

$$c_2 \oplus c_6 \oplus c_8 \oplus c_9 \oplus c_{10} = 0$$

$$c_5 \oplus c_9 \oplus c_{11} \oplus c_{12} = 0$$

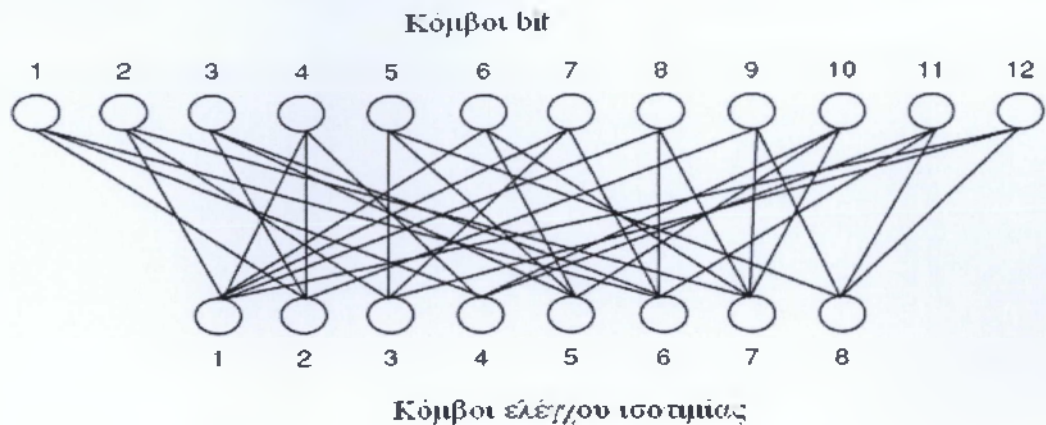
Κάθε γραμμή του πίνακα H αντιστοιχεί και σε μια εξίσωση ισοτιμίας, αρά και σε έναν κόμβο ελέγχου ισοτιμίας. Η αντιστοιχία του γράφου Tanner και των εξισώσεων ισοτιμίας είναι ευδιάκριτη για παράδειγμα από την πρώτη εξίσωση ισοτιμίας $c_2 \oplus c_4 \oplus c_6 \oplus c_7 \oplus c_8 \oplus c_{12} = 0$ προκύπτει ότι ο κόμβος ελέγχου 1 είναι συνδεδεμένος με τους κόμβους bit 2, 4, 6, 7, 8, 12.

Στην προηγούμενη ενότητα όπως περιγράψαμε έχουμε αρχικοποίηση των συντελεστών Q_i^x ως $Q_i^x = f_j^x$. Άρα στο παράδειγμα μας τα Q_i^x παίρνουν τις τιμές τους από τον πίνακα της προηγούμενης σελίδας που παραθέτει τα f_j^0 και f_j^1 .

Μέχρι αυτό το σημείο έχει ολοκληρωθεί το στάδιο της αρχικοποίησης του αλγορίθμου δηλαδή το στάδιο αρχικοποίησης των συντελεστών Q_j^x . Επόμενο στάδιο είναι το Βήμα 1 του επαναληπτικού αλγορίθμου αποκωδικοποίησης δηλαδή ανάθεση τιμών στους συντελεστές R_j^x . Να υπενθυμίσουμε ότι κάθε συντελεστής R_j^x εκφράζει την πιθανότητα να ικανοποιείται η εξίσωση ισοτιμίας i γνωρίζοντας ότι ο κόμβος bit j είναι στην κατάσταση x . Επομένως στο παράδειγμά μας, η τιμή R_2^0 εκφράζει την πιθανοτική εκτίμηση να ικανοποιείται η εξίσωση ισοτιμίας 1, δηλαδή $c_2 \oplus c_4 \oplus c_6 \oplus c_7 \oplus c_8 \oplus c_{12} = 0$ όταν το bit στη θέση 2 του διανύσματος έχει την τιμή 0. Από την η παραπάνω εξίσωση βλέπουμε ότι ικανοποιείται για οποιοδήποτε συνδυασμό των bits $c_4, c_6, c_7, c_8, c_{12}$ στον οποίο ο αριθμός των άσπων είναι άρτιος σύμφωνα από τους κανόνες της modulo-2 αριθμητικής. Για να υπολογίσουμε την πιθανότητα R_2^0 θα αθροίσουμε την πιθανότητα να ισχύει κάθε ένας από τους προαναφερθέντες συνδυασμούς. Για κάθε έναν από προαναφερθέντες συνδυασμούς για παράδειγμα τα bits $c_4, c_6, c_7, c_8, c_{12}$ να είναι όλα 0, η πιθανότητα του συνδυασμού ισούται με το γινόμενο των πιθανοτήτων κάθε bit να βρίσκεται στην κατάσταση που εξετάζουμε λόγω ανεξαρτησίας. Ήδη γνωρίζουμε ότι η πιθανότητα το bit j να βρίσκεται στην κατάσταση x εκφράζεται μέσω του συντελεστή Q_j^x . Στο σημείο αυτό μπορούμε τώρα να υπολογίσουμε την τιμή R_2^0 ως εξής:

$$\begin{aligned}
 R_2^0 = & Q_{14}^0 Q_{16}^0 Q_{17}^0 Q_{18}^0 Q_{1,12}^0 + Q_{14}^0 Q_{16}^0 Q_{17}^0 Q_{18}^0 Q_{1,12}^1 + Q_{14}^0 Q_{16}^0 Q_{17}^0 Q_{18}^1 Q_{1,12}^0 + \\
 & Q_{14}^0 Q_{16}^0 Q_{17}^1 Q_{18}^0 Q_{1,12}^0 + Q_{14}^0 Q_{16}^0 Q_{17}^1 Q_{18}^0 Q_{1,12}^1 + Q_{14}^0 Q_{16}^0 Q_{17}^1 Q_{18}^1 Q_{1,12}^0 + \\
 & Q_{14}^0 Q_{16}^0 Q_{17}^1 Q_{18}^1 Q_{1,12}^1 + Q_{14}^0 Q_{16}^1 Q_{17}^0 Q_{18}^0 Q_{1,12}^0 + Q_{14}^0 Q_{16}^1 Q_{17}^0 Q_{18}^0 Q_{1,12}^1 + \\
 & Q_{14}^0 Q_{16}^1 Q_{17}^0 Q_{18}^1 Q_{1,12}^0 + Q_{14}^0 Q_{16}^1 Q_{17}^0 Q_{18}^1 Q_{1,12}^1 + Q_{14}^0 Q_{16}^1 Q_{17}^1 Q_{18}^0 Q_{1,12}^0 + \\
 & Q_{14}^0 Q_{16}^1 Q_{17}^1 Q_{18}^0 Q_{1,12}^1 + Q_{14}^0 Q_{16}^1 Q_{17}^1 Q_{18}^1 Q_{1,12}^0 + Q_{14}^0 Q_{16}^1 Q_{17}^1 Q_{18}^1 Q_{1,12}^1 + \\
 & Q_{14}^1 Q_{16}^0 Q_{17}^0 Q_{18}^0 Q_{1,12}^0 + Q_{14}^1 Q_{16}^0 Q_{17}^0 Q_{18}^0 Q_{1,12}^1 + Q_{14}^1 Q_{16}^0 Q_{17}^0 Q_{18}^1 Q_{1,12}^0 + \\
 & Q_{14}^1 Q_{16}^0 Q_{17}^0 Q_{18}^1 Q_{1,12}^1 + Q_{14}^1 Q_{16}^0 Q_{17}^1 Q_{18}^0 Q_{1,12}^0 + Q_{14}^1 Q_{16}^0 Q_{17}^1 Q_{18}^0 Q_{1,12}^1 + \\
 & Q_{14}^1 Q_{16}^0 Q_{17}^1 Q_{18}^1 Q_{1,12}^0 + Q_{14}^1 Q_{16}^0 Q_{17}^1 Q_{18}^1 Q_{1,12}^1 + Q_{14}^1 Q_{16}^1 Q_{17}^0 Q_{18}^0 Q_{1,12}^0 + \\
 & Q_{14}^1 Q_{16}^1 Q_{17}^0 Q_{18}^0 Q_{1,12}^1 + Q_{14}^1 Q_{16}^1 Q_{17}^0 Q_{18}^1 Q_{1,12}^0 + Q_{14}^1 Q_{16}^1 Q_{17}^0 Q_{18}^1 Q_{1,12}^1 + \\
 & Q_{14}^1 Q_{16}^1 Q_{17}^1 Q_{18}^0 Q_{1,12}^0 + Q_{14}^1 Q_{16}^1 Q_{17}^1 Q_{18}^0 Q_{1,12}^1 + Q_{14}^1 Q_{16}^1 Q_{17}^1 Q_{18}^1 Q_{1,12}^0 + \\
 & Q_{14}^1 Q_{16}^1 Q_{17}^1 Q_{18}^1 Q_{1,12}^1
 \end{aligned}$$

Με αντίστοιχη ανάλυση συμπεραίνουμε ότι η τιμή R_{12}^1 προκύπτει για όλους τους συνδυασμούς με περιττό αριθμό άσπων. Παρακάτω παρουσιάζουμε σχηματικά την ανταλλαγή πληροφορίας που συντελείται για τον υπολογισμό των R_{12}^0 , R_{12}^1 και γενικά δίνουμε στον αναγνώστη να κατανοήσει καλύτερα τη διαδικασία υπολογισμού των συντελεστών R_j^x .



Σχήμα 4.5 Σχηματική απεικόνιση της διαδικασίας υπολογισμού των συντελεστών R_{12}^0, R_{12}^1

Με αυτόν τον τρόπο υπολογίζουμε τις τιμές των R_{ij}^0, R_{ij}^1 αντίστοιχα κατά την πρώτη επανάληψη.

Με βάση τη γνωστή μας σχέση:

$$d_j = \arg_x \max_j^x \prod_{k \in M(j)} R_{kj}^x$$

έχουμε τη δυνατότητα να δώσουμε την εκτίμηση που απορρέει για το διάνυσμα κατά τη πρώτη επανάληψη.

Για παράδειγμα για το 1^ο bit έχουμε:

$$d_1 = \left\{ \begin{array}{l} \hat{0} \rightarrow f_1^0 \times R_{21}^0 \times R_{41}^0 \times R_{61}^0 = 3.32 \times 10^{-7} \\ \hat{1} \rightarrow f_1^1 \times R_{21}^1 \times R_{41}^1 \times R_{61}^1 = 1.21 \times 10^{-5} \end{array} \right\} \Rightarrow '1'$$

προκύπτει ότι η πιθανότητα να είναι «0» είναι μικρότερη από τη πιθανότητα να είναι «1». Εφόσον κάνουμε την ίδια διαδικασία για όλα τα bits θα πάρουμε την πρώτη εκτίμηση για το αποκωδικοποιημένο διάνυσμα :

$$d = (111011010101)$$

Αν συγκρίνουμε το αρχικό διάνυσμα $c = (101011000101)$ με την εκτίμηση, παρατηρούμε ότι προκύπτουν 2 σφάλματα.. Στην παρούσα φάση, ο αλγόριθμος πραγματοποιεί έλεγχο εγκυρότητας ή διαφορετικά όπως έχουμε ήδη αναφέρει, ελέγχει

εάν το σύνδρομο $d \circ H^T = 0$. Εφόσον το σύνδρομο δεν είναι μηδενικό διάνυσμα, ξεκινάει μια νέα επανάληψη.

Αρχικά, οι ανανεωμένες τιμές των συντελεστών Q_i^x στέλνονται προς τους κόμβους.

Οι τιμές των συντελεστών υπολογίζονται από την εξής σχέση:

$$Q_{ij}^x = a_{ij} f_j^x \prod_{k \in M(j)^x} R_k^x$$

Παραδείγματος χάριν,

$$Q_{12}^0 = a_{12} f_2^0 R_{32}^0 R_{72}^0$$

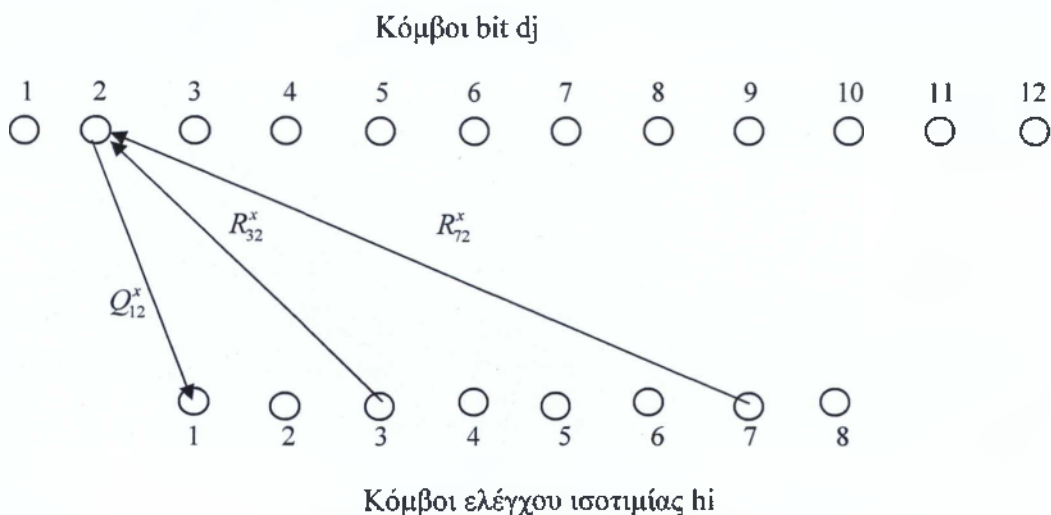
και

$$Q_{12}^1 = a_{12} f_2^1 R_{32}^1 R_{72}^1$$

Το k στον συμβολισμό R_{kj} , αναλογεί σε όλους τους κόμβους ελέγχου ισοτιμίας οι οποίοι είναι συνδεδεμένοι με τον d_i bit κόμβο, εκτός του κόμβου i . Επίσης, η κανονικοποιητική σταθερά a_{12} εισάγεται προκειμένου να ισχύει η παρακάτω εξίσωση:

$$Q_{12}^0 + Q_{12}^1 = 1, \text{ επομένως προκύπτει } a_{12} = \frac{1}{f_2^0 R_{32}^0 R_{72}^0 + f_2^1 R_{32}^1 R_{72}^1}$$

Για την περίπτωση που αναφέραμε, παραθέτουμε τη σχηματική παράσταση, με τη συνδρομή του διμερούς γράφου, όπου μέσω αυτής πραγματοποιείται ο υπολογισμός των συντελεστών Q_{12}^0 και Q_{12}^1 επομένως προκύπτει και ο υπολογισμός όλων των συντελεστών Q_i^x .



Σχήμα 4.6 Σχηματική απεικόνιση της διαδικασίας υπολογισμού των συντελεστών Q_{12}^0, Q_{12}^1

Επομένως, στο Βήμα 1 του επαναληπτικού αλγορίθμου κατά τη δεύτερη επανάληψη στέλνονται οι ακόλουθες τιμές των συντελεστών Q_0^0 και Q_1^1 αντίστοιχα.

Στη συγκεκριμένη περίπτωση, εν αντιθέσει με τη πρώτη επανάληψη, στους συντελεστές Q_j^i εμπεριέχονται και πληροφορία κώδικα εκτός από πληροφορία διαύλου. Όπως προκύπτει, αυτό το χαρακτηριστικό θα υφίσταται σε όλες τις επαναλήψεις. Καθώς προχωράμε με τα βήματα της δεύτερης επανάληψης, μέσω των γενικών σχέσεων που ισχύουν κάθε φορά, υπολογίζουμε τις τιμές των συντελεστών R_0^0 και R_1^1 .

Εφόσον έχουν υπολογιστεί οι τιμές του R_j^i , θα προκύψει η εκτίμηση του αποκωδικοποιημένου διανύσματος κατά τη δεύτερη επανάληψη ακολουθώντας πάλι τις εξισώσεις που υπολογίζουν τις πιθανότητες d_j^0 και του d_j^1 . Έτσι προκύπτει ότι το εκτιμώμενο κωδικοποιημένο διάνυσμα στο τέλος της δεύτερης επανάληψης είναι $d = (1 \ 0 \ 0 \ 0 \ 1 \ 1 \ 0 \ 0 \ 0 \ 1 \ 0 \ 1)$, το οποίο και διαφέρει από το αρχικό κωδικοποιημένο διάνυσμα μόνο στο 3ο bit. Πάλι το σύνδρομο θα είναι μη μηδενικό, επομένως μία καινούρια επανάληψη αρχίζει, που είναι η τρίτη στη σειρά.

Κατά τη τρίτη επανάληψη, ο αποκωδικοποιητής πετυχαίνει τη διόρθωση όλων των σφαλμάτων τα οποία είχε εισάγει ο δίαυλος και δίνει στην έξοδο το αρχικό μήνυμα m .

Ως εκ τούτου προκύπτει η εκτίμηση $d = (1 \ 0 \ 1 \ 0 \ 1 \ 1 \ 0 \ 0 \ 0 \ 1 \ 0 \ 1) = c$. Μηδενικό σύνδρομο έχει το διάνυσμα d επομένως ο αλγόριθμος μετά τη τρίτη επανάληψη τερματίζει με επιτυχία. Τότε, ο αποκωδικοποιητής οδηγεί στην έξοδο το αποκωδικοποιημένο μήνυμα $m = (0101)$, το οποίο και ταυτίζεται με το αρχικό μήνυμα πληροφορίας.

5 Αποτελέσματα Προσομοίωσης και συζήτηση αποτελεσμάτων

Στο τελευταίο κεφάλαιο θα γίνει παρουσίαση μέσω γραφικών παραστάσεων τα αποτελέσματα της προσομοίωσης μας που έχει σαν στόχο την εκτίμηση της επίδοσης των LDPC κωδίκων. Στα χαρακτηριστικά των κωδίκων και το πως επηρεάζεται η επίδοσή τους μέσα από διάφορα σχήματα διαμόρφωσης και διάφορους τύπου διαύλου θα εκτελεστεί η μελέτη μας. Τα αποτελέσματα που θα παρουσιαστούν είναι όλα από Matlab στο οποίο γράφτηκε και κώδικας της προσομοίωσης.

Πρωταρχικό στάδιο της εργασίας μας ήταν η κατασκευή του πίνακα ελέγχου ισοτιμίας H , ο οποίος κατασκευάστηκε σύμφωνα με τις απαιτήσεις που αναφέρθηκαν στο κεφάλαιο 4. Κύριο χαρακτηριστικό του πίνακα H είναι η αραιότητα του, το οποίο είναι γενικά χαρακτηριστικό στους LDPC κώδικες. Έγινε προσπάθεια ώστε να μην υπάρχουν κύκλοι μικρού μήκους στον αντίστοιχο γράφο Tanner σύμφωνα με την ανάλυση που προηγήθηκε στην ενότητα 4.3.1. Οι διαστάσεις του πίνακα H είναι 128×256 , συνεπώς είναι LDPC(256,128) κώδικας. Είναι κανονικός πίνακας και έχει 6 άσσους σε κάθε γραμμή και 3 σε κάθε στήλη. Τα αποτελέσματα που θα ακολουθήσουν σχετίζονται με τον παραπάνω πίνακα. Επίσης, ο μέγιστος αριθμός επαναλήψεων του επαναληπτικού αλγόριθμου αποκωδικοποίησης είναι ένα σταθερός στοιχείο και αν δεν αναφερθεί κάτι άλλο είναι κάθε φορά $80(\max_iter=80)$.

5.1 Μελέτη πάνω στα χαρακτηριστικά του LDPC κώδικα

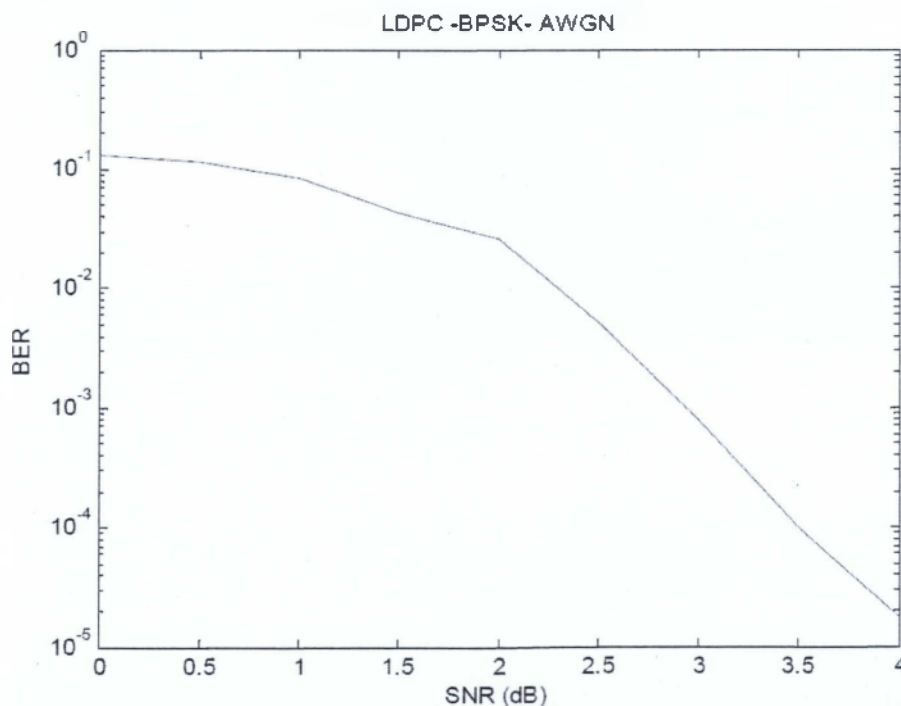
Σαν πρώτο στάδιο επικεντρωθήκαμε στην μελέτη των χαρακτηριστικών του κώδικα και κρατήσαμε σταθερά το σχήμα διαμόρφωσης(BPSK) και το τύπο διαύλου(AWGN), από τα οποία παραθέτουμε τα αποτελέσματα στην συγκεκριμένη ενότητα.

5.1.1 Επίδοση του κώδικα για εύρος SNR από 0-4 dB

Για να γίνει καλύτερη εκτίμηση της επίδοσης του κώδικα αυξήσαμε το εύρος τιμών του σηματοθορυβικού λόγου έως τα 4 dB, το οποίο απεικονίζεται στην γραφική παράσταση του σχήματος 5.1. Υπολογίζοντας το μέγεθος του κώδικα παρατηρείται ότι έχει αρκετά καλή επίδοση. Βέβαια έχουν κατασκευαστεί πολύ ισχυρότεροι

κώδικες LDPC και φυσικά με πολύ μεγαλύτερο μέγεθος κώδικα. Για παράδειγμα να αναφέρουμε ότι στο πρότυπο DVB-S2 το μήκος της κωδικής λέξης LDPC είναι 64800 bits. Για αντίστοιχα μεγέθη κώδικα το κόστος είναι οι υψηλές απαιτήσεις αποθηκευτικού χώρου διότι οι πίνακες ελέγχου ισοτιμίας είναι μεγάλων διαστάσεων, επιπλέον υπάρχει αύξηση της πολυπλοκότητας τόσο της κωδικοποίησης όσο και της αποκωδικοποίησης.

Στη καμπύλη επίδοσης του σχήματος 5.1 παρατηρούμε ότι έπεται της κλασσικής μορφής καμπυλών επίδοσης τεχνικών κωδικοποίησης. Αυτό ήταν σίγουρο λόγω της αύξησης του σηματοθορυβικού λόγου να παρατηρήσουμε μείωση του ρυθμού λαθών. Για παράδειγμα παρατηρώντας την τιμή του SNR για 3.5 dB βλέπουμε ότι ο ρυθμός λαθών για κάθε ψηφίο είναι 10^{-4} , δηλαδή για την συγκεκριμένη τιμή SNR παρουσιάστηκε ένα λάθος για κάθε 1000bits, με άλλα λόγια η εκτίμηση της προσομοίωσης μας για την πιθανότητα λάθους είναι 10^{-4} . Επίσης να αναφέρουμε ότι ο κατακόρυφος άξονας BER δίνεται σε λογαριθμική κλίμακα.



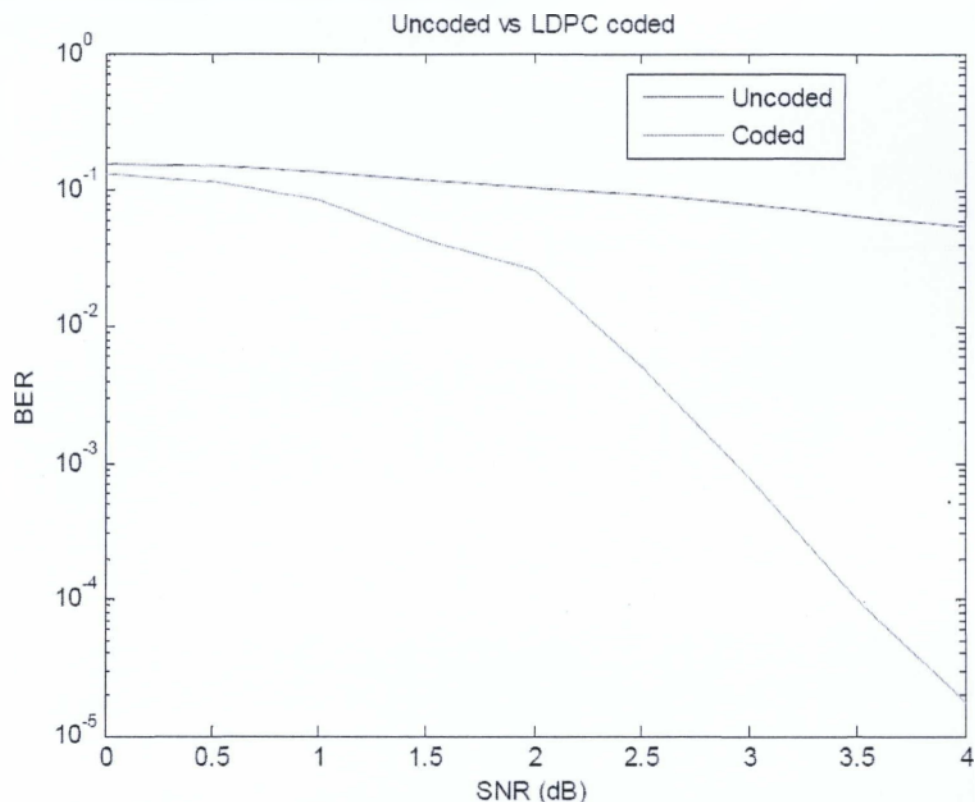
Σχήμα 5.1 Επίδοση LDPC (256,128) υπό διάυλο AWGN και διαμόρφωση BPSK

5.1.2 Κέρδος από τη χρήση κωδικοποίησης

Επόμενο θέμα για προσομοίωση είναι η μελέτη και η σύγκριση του συστήματος το οποίο χρησιμοποιεί την κωδικοποίηση μας με σύστημα το οποίο δεν χρησιμοποιεί κωδικοποίηση. Για να είναι αξιόπιστη η σύγκριση μας πρέπει και τα δύο συστήματα να χρησιμοποιούν την ίδια τεχνική διαμόρφωσης και τον ίδιο τηλεπικοινωνιακό δίαυλο, και επίσης όπως αναφέραμε και προηγουμένως μέχρι στιγμής θα εξετάσουμε μόνο BPSK διαμόρφωση και AWGN δίαυλο.

Στο σχήμα 5.2 απεικονίζεται η σύγκριση των συστημάτων μας από το οποίο μπορούμε να βγάλουμε χρήσιμα συμπεράσματα. Αρχικά παρατηρούμε ότι όσο αυξάνεται ο σηματοθορυβικός λόγος χρησιμοποιώντας κωδικοποίηση έχουμε ένα συνεχόμενο αυξανόμενο κέρδος. Στις μικρές τιμές του SNR η εμφάνιση θορύβου σε σύγκριση με το σήμα μας είναι ιδιαίτερα έντονη. Επομένως, η μεταδιδόμενη κωδική λέξη αλλοιώνεται σημαντικά δυσκολεύοντας πολύ τη δουλειά του αποκωδικοποιητή για την διόρθωση των λαθών, και κατά συνέπεια έως ο σηματοθορυβικός λόγος να είναι ίσο με 1 dB δεν διακρίνεται κάποια σημαντική βελτίωση της επίδοσης. Να σημειώσουμε εδώ, ότι από την γενική και θεωρητική καμπύλη του σχήματος 2.11 υπάρχει περίπτωση να χειροτερεύσει η επίδοση χρησιμοποιώντας κωδικοποίηση για πιο χαμηλές τιμές του SNR. Τέτοια περίπτωση μπορεί να εξηγηθεί αν υπολογίσουμε ότι για σηματοθορυβικό λόγο κάτω από 0 dB η αλλοίωση της κωδικής λέξης υπάρχει πιθανότητα να είναι αρκετά μεγάλη ώστε να 'μετακινείται' από την περιοχή της αρχικής κωδικής λέξης σε περιοχές άλλων κωδικών λέξεων και ο αποκωδικοποιητής θα δυσκολέψει τα πράγματα. Αντιθέτως για τις ικανοποιητικές τιμές του σηματοθορυβικού λόγου παρατηρούμε ότι η κωδική λέξη φθάνει αλλοιωμένη αλλά όχι 'διεφθαρμένη', έχοντας και ο αποκωδικοποιητής την δυνατότητα να διορθώσει τα λάθη. Έτσι, βλέπουμε κέρδος χρησιμοποιώντας κωδικοποίηση που αυξάνοντας το SNR γίνεται τεράστιο. Η τιμή του BER για 4 dB για παράδειγμα, προκύπτει από την προσομοίωση να είναι 0.0552 χωρίς να έχει χρησιμοποιηθεί κωδικοποίηση και 0.000017188 χρησιμοποιώντας κωδικοποίηση, συμπεραίνοντας ότι χρησιμοποιώντας κωδικοποίηση καταφέρνουμε να έχουμε 3200 ρυθμό λαθών μικρότερο από ότι χωρίς χρήση κωδικοποίησης. Χωρίς να εκτελέσουμε την αντίστοιχη προσομοίωση μπορούμε να συμπεράνουμε από την μορφή των καμπυλών ότι για σηματοθορυβικό λόγο πάνω από 4 dB το κέρδος χρησιμοποιώντας κωδικοποίηση θα γίνει μεγαλύτερο. Επίσης, να σημειώσουμε ότι το τίμημα για αυτή

την σημαντική μείωση του ρυθμού λαθών που έχει ως συνέπεια και την αύξηση της ποιότητας του συστήματος χρησιμοποιώντας κωδικοποίηση, ισχύει το γενικό ισοζύγιο στο οποίο αναφερθήκαμε στο κεφάλαιο 2.6.5.



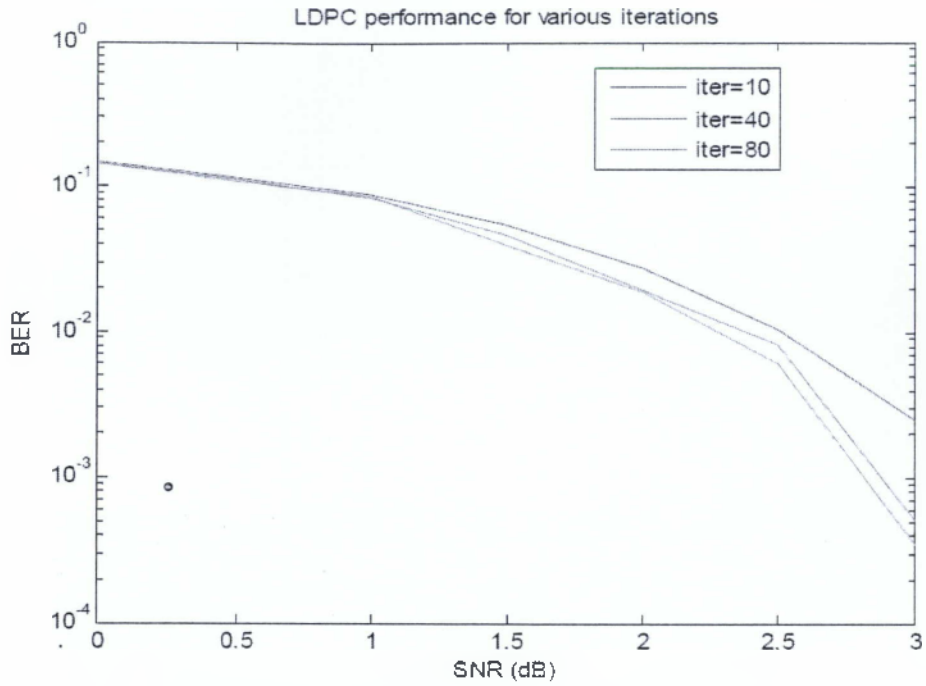
Σχήμα 5.2 Συστήματα με και χωρίς χρήση κωδικοποίησης

5.1.3 Μελέτη επίδοσης του κώδικα για διάφορες τιμές του μέγιστου αριθμού επαναλήψεων του επαναληπτικού αλγορίθμου κωδικοποίησης

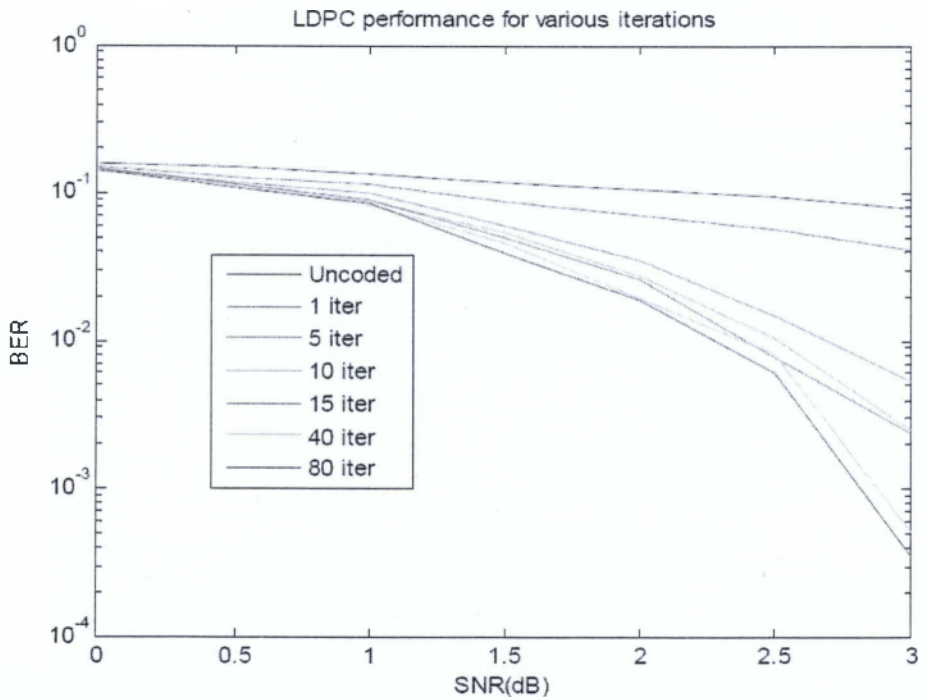
Συνεχίζοντας τη σχεδίαση του LDPC(256,128) κώδικα θέλουμε να συγκρίνουμε την επίδοση του για διάφορες τιμές του μέγιστου αριθμού επαναλήψεων του επαναληπτικού αλγορίθμου αποκωδικοποίησης, για να γίνει αυτό όμως έπρεπε να αλλάξουμε αναλόγως τον εκτελέσιμο κώδικα των προηγούμενων προσομοιώσεων. Στις προηγούμενες προσομοιώσεις αποθηκεύαμε τιμές BER ανά SNR σε μια δομή δεδομένων, ενώ τώρα δημιουργήσαμε μια καινούργια δομή δεδομένων στην οποία θα αποθηκεύουμε τιμές BER ανά SNR και ανά επανάληψη. Έχοντας αυτή την τρισδιάστατη δομή στο τερματισμό της προσομοίωσης, χρησιμοποιώντας το Matlab

έχουμε την δυνατότητα να δημιουργήσουμε γραφικές παραστάσεις επίδοσης του κώδικα για διάφορες τιμές επαναλήψεων του αλγορίθμου αποκωδικοποίησης.

Το πρώτο αποτέλεσμα αυτού του θέματος απεικονίζεται στο σχήμα 5.3, στο οποίο βλέπουμε την επίδοση του LDPC κώδικα για 10, 40, και 80 επαναλήψεις. Αρχικά παρατηρούμε ότι με μεγαλύτερο αριθμό επαναλήψεων έχουμε και καλύτερη επίδοση. Αλλά πρώτον να σημειώσουμε ότι για μικρές τιμές σηματοθορυβικού λόγου δεν υπάρχει κάποια διαφοροποίηση και δεύτερον στις 40 και στις 80 επαναλήψεις δεν υπάρχει κάποια σημαντική διαφορά στην επίδοση. Όπως αναφέραμε και προηγουμένως, το πρώτο συμπέρασμα προκύπτει λόγω της μικρής τιμής σηματοθορυβικού λόγου όπου αλλοιώνει πολύ την κωδική λέξη. Επομένως, αν είναι μεγάλη η αλλοίωση προσπερνάμε τις διορθωτικές ικανότητες του κώδικα και ένα μεγάλο ποσοστό των επαναλήψεων δεν μπορεί να προσφέρει κάτι ουσιώδες το οποίο φυσικά δεν ισχύει για ικανοποιητικές τιμές του SNR. Τώρα για το δεύτερο συμπέρασμα αιτιολογείται και ενισχύεται από το σχήμα 5.4 στο οποίο απεικονίζουμε μελέτη επίδοσης με διάφορες τιμές επανάληψης όπως για 1, 5, 10, 15, 40 και 80, όπως επίσης και επίδοση χωρίς να χρησιμοποιήσουμε κωδικοποίηση. Όπως βλέπουμε στο σχήμα και εξετάζοντας κάθε φορά ικανοποιητικές τιμές σηματοθορυβικού λόγου, παρατηρούμε ότι από την μετάβαση της 1 επανάληψης στις 5 έχουμε μεγάλη αύξηση στον ρυθμό βελτίωσης, ενώ όσο αυξάνονται ο αριθμός των επαναλήψεων μειώνεται αυτό ο ρυθμός. Δηλαδή πάνω από έναν αριθμό επαναλήψεων δεν υπάρχει κάποια σημαντική βελτίωση προκύπτοντας έτσι ένα είδος κορεσμού του κώδικα, ο οποίος αιτιολογεί και την επιλογή μας για αριθμό επαναλήψεων ίσο με 80. Άρα και 1000 επαναλήψεις να εκτελούσαμε στον συγκεκριμένο κώδικα δεν θα βλέπαμε κάποια σημαντική βελτίωση σε σύγκριση με τις 80 επαναλήψεις, τις οποίες θέσαμε για τον κώδικα μας έπειτα από πειραματικά αποτελέσματα αλλά δεν αποτελούν γενική συνταγή για τους LDPC κώδικες λόγω του ότι πιθανόν να υπάρχει διαφορετική συμπεριφορά ανάλογα με τον κώδικα αλλά και με την δημιουργία του πίνακα ελέγχου ισοτιμίας. Συγχρόνως δίνουμε και την δυνατότητα σύγκρισης και με σύστημα χωρίς να χρησιμοποιεί κωδικοποίηση και βλέπουμε ότι το σύστημα με κωδικοποίηση υπερέχει αυξητικά και μόνο με μία επανάληψη για αυξανόμενες τιμές του SNR.



Σχήμα 5.3 Επίδοση για 10, 40 και 80 επαναλήψεις



Σχήμα 5.4 Επίδοση χωρίς κωδικοποίηση και επίδοση για 1, 5, 10, 15, 40, και 80 επαναλήψεις.

5.2 Επίδοση του LDPC (256, 128) Κώδικα σε σχέση με διάφορους Τύπους Διαύλου και διάφορα Σχήματα Διαμόρφωσης

Εφόσον μελετήσαμε τα φυσικά χαρακτηριστικά του κώδικα, ο επόμενος στόχος μας είναι να μελετήσουμε την επίδοση του κώδικα όταν αλλάζουν οι εξωτερικοί παράγοντες. Ορισμένοι από αυτούς είναι η τεχνική διαμόρφωσης ή ο τύπος του διαύλου. Στην ενότητα αυτή χρησιμοποιείται ο ίδιος κώδικας με την προηγούμενη ενότητα, ο κώδικας LDPC(128, 256), του οποίου ο μέγιστος αριθμός επαναλήψεων είναι ίσος με 80. Παρακάτω, έχουμε τα αποτελέσματα της προσομοίωσης μέσω του περιβάλλοντος του Matlab.

5.2.1 Διαμόρφωση 4 – QAM και δίαυλος τύπου Rayleigh

Για να διευρύνουμε την εργασία μας, εκτός από την «απλή BPSK» διαμόρφωση, προσομοιώνουμε και άλλες διαμορφώσεις όπως επίσης και διαύλους οι οποίοι περιγράφουν το μοντέλο AWGN περιβάλλοντα κινητών επικοινωνιών, που όπως είναι γνωστό, η πολυδιαδρομική διάδοση προκαλεί διαλείψεις. Για αρχή, θα εξετάσουμε την επίδοση του κώδικα LDPC με τη χρήση της 4 – QAM διαμόρφωσης και με τη χρήση διαύλου τύπου Rayleigh με διαλείψεις.

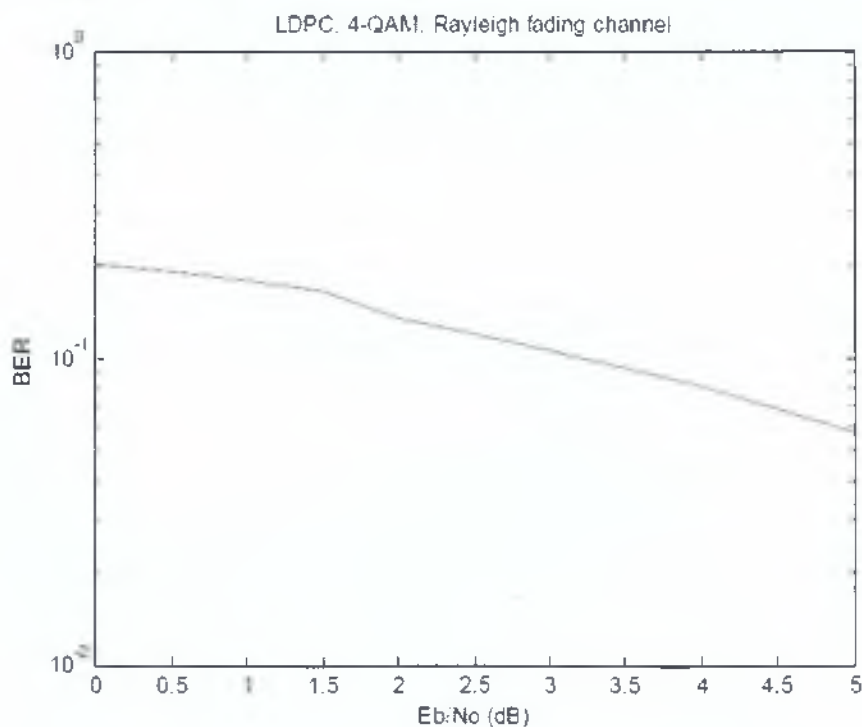
Στα Σχήματα 5.5 και 5.6 διακρίνονται τα αποτελέσματα της προσομοίωσης. Ομοίως και εδώ, η επίδοση μετριέται σε BER (Bit Error Rate) συναρτήσει του λόγου E_b/N_0 , δηλαδή του λόγου της ενέργειας ψηφίου (bit) προς τη πυκνότητα του θορύβου. Να υπενθυμίσουμε ότι ο σηματοθορυβικός λόγος SNR συνδέεται με το λόγο ενέργειας ψηφίου προς πυκνότητα θορύβου E_b/N_0 με την εξής σχέση:

$$SNR = \frac{S}{N} = \frac{E_s}{N_0} \cdot \frac{R_s}{B}$$

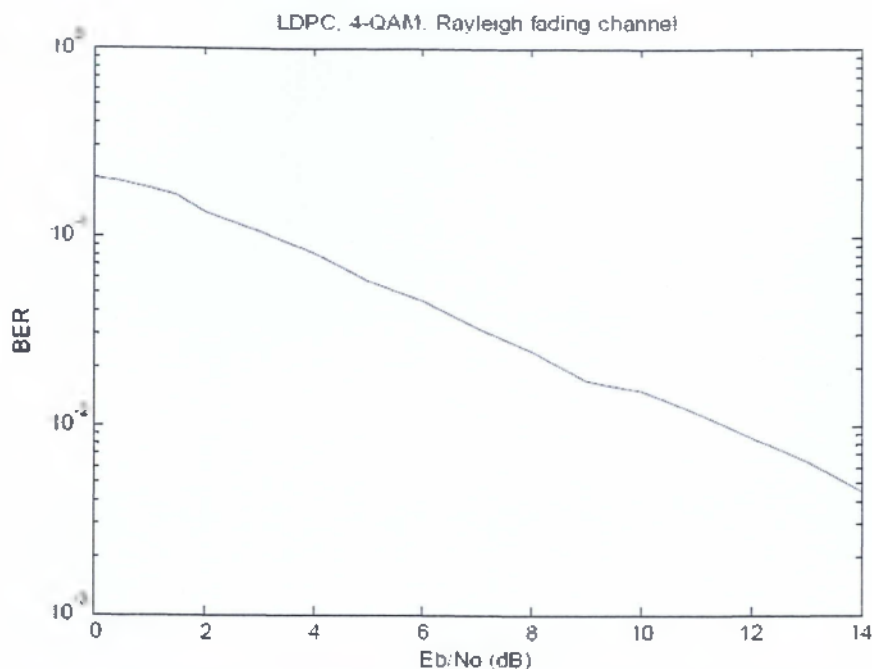
Θεωρήσαμε ότι στην προσομοίωσή μας ο ρυθμός μετάδοσης R_s εκτείνεται σε όλο το διαθέσιμο εύρος του συστήματος B . Επομένως, σε ότι σχετίζεται με τη προσομοίωσή μας, $R_s = B$, συνεπώς οι λόγοι E_b/N_0 και SNR είναι ισοδύναμοι στην παρούσα προσομοίωση. Στο Σχήμα 5.5 βλέπουμε την μείωση του ρυθμού λαθών με την αύξηση του σηματοθορυβικού λόγου, όμως η επίδοση είναι αρκετά χαμηλότερη αν τη συγκρίνουμε με προηγούμενα αποτελέσματα, όπως διακρίνουμε για παράδειγμα στο Σχήμα 5.1. Επίσης, ακόμα και για μεγάλες τιμές του σηματοθορυβικού λόγου δεν διακρίνεται σημαντική βελτίωση της επίδοσης όπως διακρίνεται στο Σχήμα 5.6. Η

συγκεκριμένη πτώση στην επίδοση του συστήματος, σε σύγκριση με την επίδοση για δίαυλο AWGN και για BPSK διαμόρφωση, προέρχεται από τους εξής παράγοντες:

- Ο δίαυλός μας που πλέον χαρακτηρίζεται από διαλείψεις που επιβαρύνει το σήμα περισσότερο από τον δίαυλο AWGN που το σήμα του δεν υποφέρει από διαλείψεις.
- Η 4 – QAM καθ’ ότι είναι τεχνικά υψηλότερης φασματικής απόδοσης από την διαμόρφωση BPSK, προκαλεί αύξηση του ρυθμού λαθών για ίδιο σηματοθορυβικό λόγο (trade – off). Μια ανάλογη ανάλυση είχαμε και στην ενότητα για την αποδιαμόρφωση (Ενότητα 3.5.5).



Σχήμα 5.5 Διαμόρφωση 4 – QAM και δίαυλος τύπου Rayleigh



Σχήμα 5.6 Διαμόρφωση 4 – QAM και διάυλος τύπου Rayleigh

5.2.2 Διαμόρφωση 4 – QAM και διάυλος τύπου Rice

Εξακολουθούμε να χρησιμοποιούμε το ίδιο σχήμα διαμόρφωσης (4 – QAM), αλλάζοντας τον διάυλο σε τύπου Rice. Να σημειώσουμε ότι στο διάυλο Rice υπάρχουν διαλείψεις, όμως υφίσταται και η οπτική επαφή του πομπού με τον δέκτη, κάτι το οποίο δεν συμβαίνει στον διάυλο τύπου Rayleigh. Διακρίνουμε λοιπόν δύο ειδών κύματα τα οποία φτάνουν στον δέκτη. Αρχικά έχουμε το απευθείας κύμα (direct wave) και έπειτα έχουμε τα κύματα που φτάνουν σκεδασμένα (scattered waves). Μείζονος σημασίας για να χαρακτηριστεί ένας διάυλος Rice είναι με πόσο μεγαλύτερη ισχύ φτάνει το απευθείας κύμα σε σχέση με τα σκεδασμένα. Επομένως, ορίζουμε τον παράγοντα K (Rician K – factor) ως:

$$K = \frac{\text{ΙΣΧΥΣ ΑΠΕΥΘΕΙΑΣ ΚΥΜΑΤΟΣ}}{\text{ΙΣΧΥΣ ΣΚΕΔΑΣΜΕΝΩΝ ΚΥΜΑΤΩΝ}} \quad (5.1)$$

όπου όπως ήδη αναφέρθηκε μας ενδιαφέρει η ληφθείσα ισχύς στον δέκτη. Με μία πρώτη ματιά, διαπιστώνουμε ότι για πολύ μικρό K ($K \rightarrow 0$) είναι σαν να μην έχει οπτική επαφή ο πομπός με τον δέκτη, επομένως είναι αναμενόμενο ο διάυλος να εμφανίζει χαρακτηριστικά του διαύλου Rayleigh. Αντιθέτως, για πολύ μεγάλο K (K

→ ∞), επικρατεί η απευθείας μετάδοση δίχως να επηρεάζεται από φαινόμενα πολύοδης διάδοσης. Περιμένουμε λοιπόν ότι σε μία τέτοια περίπτωση, ο διάυλος θα τείνει να αποκτήσει τα χαρακτηριστικά του μοντέλου AWGN.

Στην προσομοίωσή μας πραγματοποιήθηκε διάυλος τύπου Rice με $K=20$. Η γραφική αναπαράσταση του ρυθμού λαθών ανά ψηφίο (BER) συναρτήσει του λόγου E_b / N_0 , αποδίδεται στα σχήματα 5.7 και 5.8. Επαληθεύεται ότι η επίδοση σε διάυλο Rice είναι πιο καλή σε σύγκριση με την επίδοση του διαύλου Rayleigh. Συγκεκριμένα, για $K=20$ η διαφορά είναι σημαντική. Αυτό που μένει είναι να συγκρίνουμε τον διάυλο Rice με τον διάυλο AWGN, κάτι το οποίο λαμβάνει χώρα παρακάτω. Ο τελικός σκοπός αυτής της πορείας είναι η παρουσίαση ενός κοινού συγκριτικού διαγράμματος για διαύλους τύπου AWGN, Rice και Rayleigh.

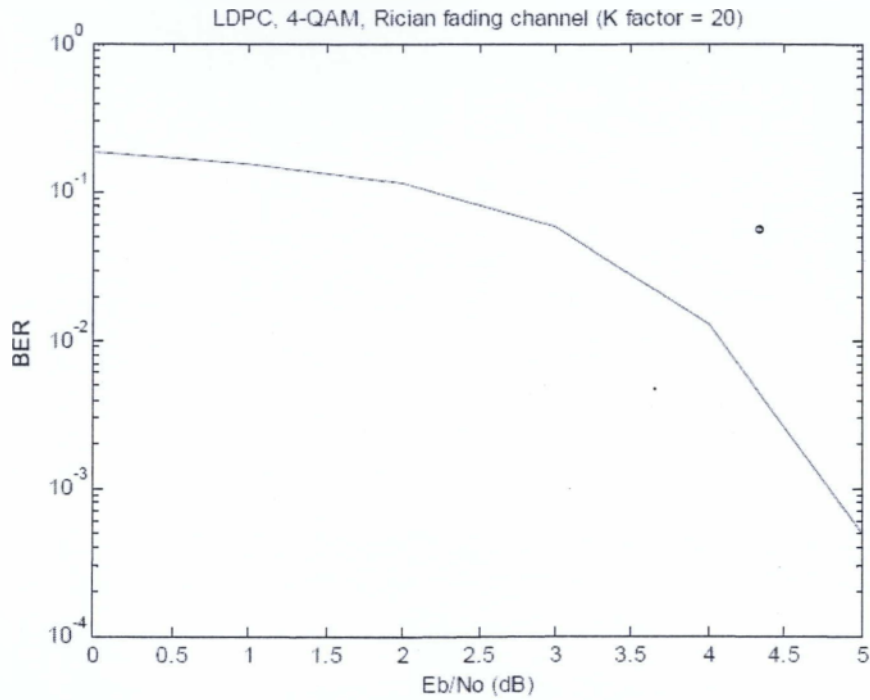
5.2.3 Διάυλος τύπου AWGN και διαμόρφωση 4 – QAM

Συνεχίζουμε τη πορεία της σύγκρισης των τύπων διαύλου κρατώντας σταθερή τη διαμόρφωση σε 4 – QAM και χρησιμοποιώντας την ίδια κωδικοποίηση LDPC. Στο Σχήμα 5.9 φαίνονται τα αποτελέσματα της συγκεκριμένης προσομοίωσης. Η επίδοση είναι σαφέστατα καλύτερη απ' ότι με διάυλο Rayleigh, όμως για να συγκριθεί με διάυλο Rice συνίσταται κοινό διάγραμμα, το οποίο θα γίνει παρακάτω.

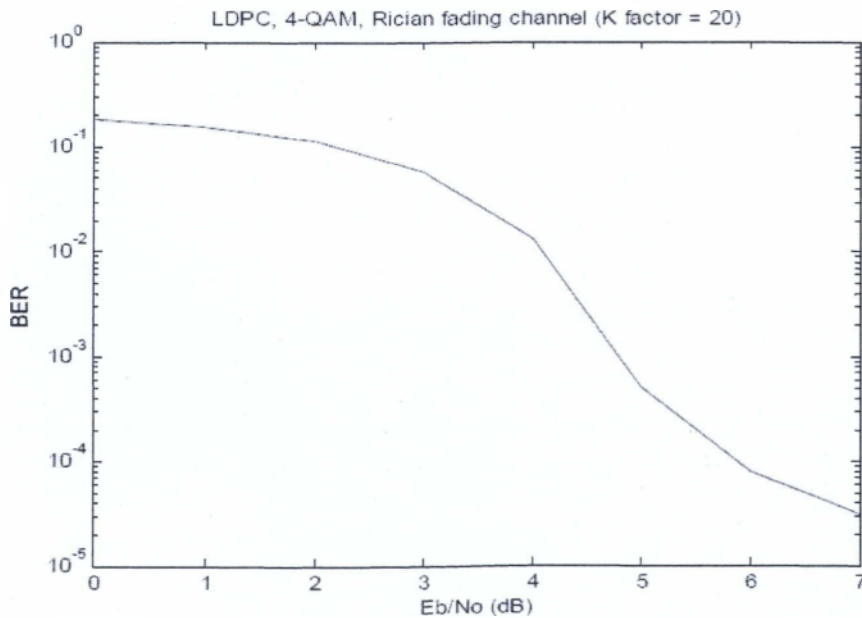
5.2.4 Σύγκριση επίδοσης για διαύλους τύπου AWGN, Rayleigh, Rice.

Στα σχήματα 5.10 και 5.11 διακρίνονται και τα συγκριτικά διαγράμματα επίδοσης. Από το Σχήμα 5.11 φαίνεται ότι ο συγκεκριμένος Rice διάυλος οδηγεί σε συγκρίσιμη επίδοση με διάυλο AWGN. Βέβαια, από $E_b / N_0 > 6$ dB, υπάρχει σαφής διαφορά σε διάυλο AWGN, καθώς από την κλίση των γραφικών παραστάσεων είναι αναμενόμενη η αύξηση της μεταξύ τους διαφοράς. Παρ' όλα αυτά, το Σχήμα 5.11 φανερώνει ότι για $K=20$, ο Rice διάυλος φέρεται περισσότερο σαν διάυλος AWGN παρά σαν Rayleigh. Όπως συμπεραίνεται, κάνοντας λόγο για τους εξεταζόμενους διαύλους, έχουμε στο ένα άκρο διάυλο Rayleigh, με τη μέγιστη επιβάρυνση και στο άλλο άκρο έχουμε διάυλο AWGN, με την ελάχιστη επιβάρυνση. Ο σκοπός της προσομοίωσης, είναι η μελέτη της συμπεριφοράς του Rice διαύλου για τις διάφορες τιμές του K παράγοντα σε σύγκριση με τους άλλους δύο διαύλους. Παραδείγματος

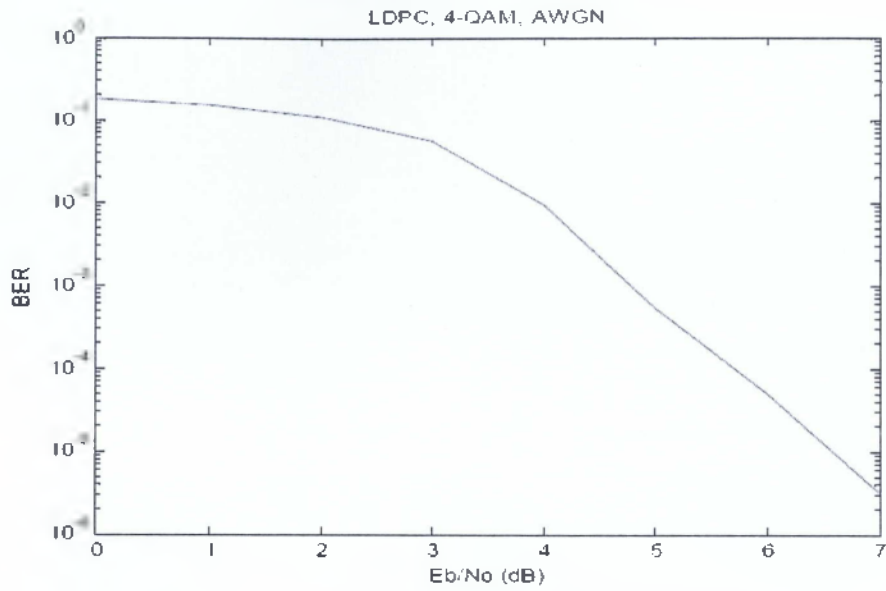
χάριν, εμείς καταλήξαμε στο συμπέρασμα ότι για $K=20$, τουτέστιν η ισχύς του απευθείας κύματος είναι 20 φορές μεγαλύτερη από την ισχύ των κυμάτων που φτάνουν έπειτα από σκέδαση, τότε ο Rice διάυλος εμφανίζει χαρακτηριστικά αρκετά μεγαλύτερης ομοιότητας με διάυλο AWGN παρά με διάυλο Rayleigh.



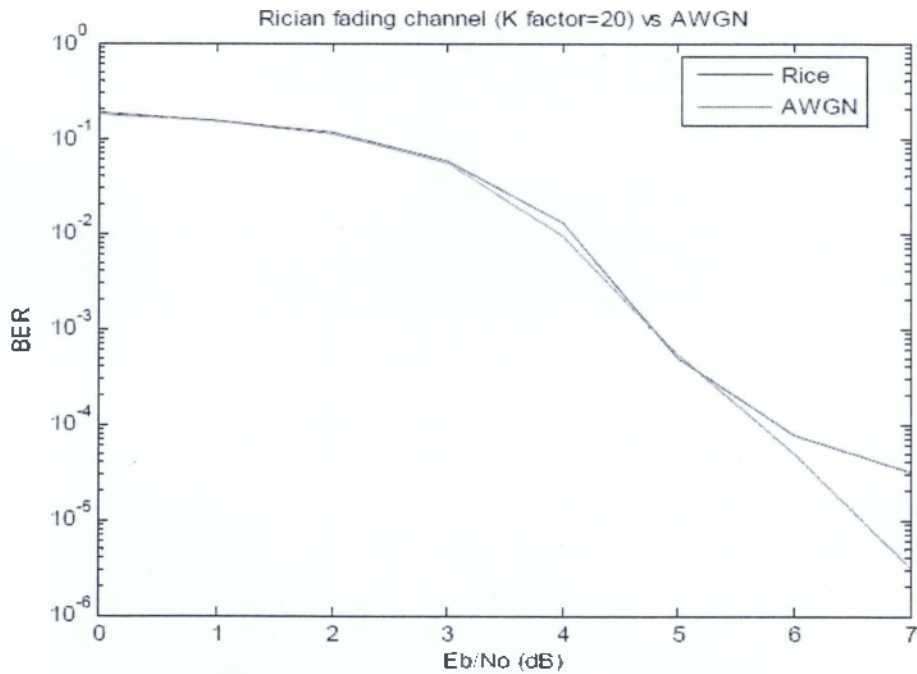
Σχήμα 5.7 Διαμόρφωση 4 – QAM για διαύλους τύπου Rice ($K=20$)



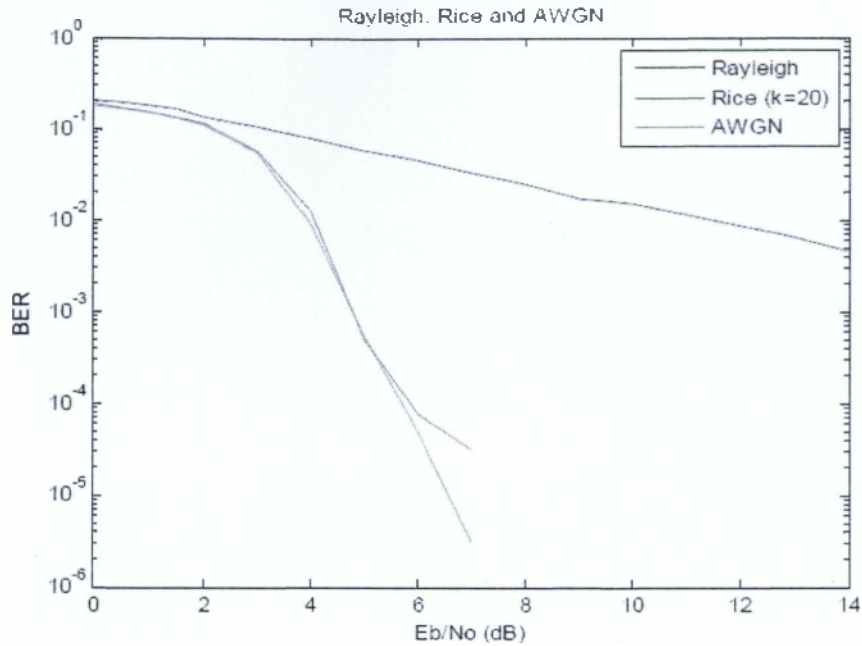
Σχήμα 5.8 Διαμόρφωση 4 – QAM για διαύλους τύπου Rice ($K=20$)



Σχήμα 5.9 Διαμόρφωση 4 – QAM για διαύλους τύπου AWGN



Σχήμα 5.10 Σύγκριση AWGN, Rice (K=20)



Σχήμα 5.11 Σύγκριση AWGN, Rice (K=20) και Rayleigh

5.3 Συζήτηση αποτελεσμάτων

Βασικός στόχος της παρούσας εργασίας ήταν η μελέτη και προσομοίωση LDPC κωδίκων με σκοπό να εξαχθούν ποιοτικά αλλά και ποσοτικά συμπεράσματα για την συμπεριφορά τους. Κάτι τέτοιο έχει ιδιαίτερη αξία καθώς οι LDPC κώδικες χαρακτηρίζονται ως κώδικες αιχμής και βρίσκουν πρακτική εφαρμογή σε ολόένα και περισσότερα τηλεπικοινωνιακά συστήματα. Δώσαμε ιδιαίτερη έμφαση στις τεχνικές διαμόρφωσης καθώς και στη μοντελοποίηση του διαύλου. Παράλληλα με τη μελέτη των κωδίκων λοιπόν, εξάγαμε και συμπεράσματα σχετικά με τη συγκριτική μελέτη τύπων διαύλου, με έμφαση στους διαύλους που απαντώνται στις κινητές και δορυφορικές επικοινωνίες (AWGN, Rayleigh, Rice). Συνοψίζοντας τη συμβολή της εργασίας μπορούμε να αναφερθούμε στα παρακάτω στοιχεία:

- Επισκόπηση των τεχνικών κωδικοποίησης καναλιού με έμφαση στην παρουσίαση των γραμμικών συμπαγών κωδίκων, κατηγορία στην οποία ανήκουν οι LDPC κώδικες. Ανάλυση των δυνατοτήτων των κωδίκων όσον αφορά τον εντοπισμό και τη διόρθωση λαθών και παρουσίαση τρόπων αξιολόγησης τεχνικών κωδικοποίησης.
- Μελέτη των LDPC κωδίκων και ανάλυση των αλγορίθμων αποκωδικοποίησης τους. Προσομοίωση ενός LDPC (256, 128) κώδικα που οδήγησε στην εξαγωγή πολύ

χρήσιμων συμπερασμάτων σχετικά με τις ιδιότητες του. Τα συμπεράσματα μας είναι τα εξής:

- Απόδοση των LDPC κωδίκων πολύ κοντά στα θεωρητικά όρια που θέτει η θεωρία του Shannon
- Η διαφορά απόδοσης των κωδίκων LDPC σε σχέση με την περίπτωση καμίας χρήσης κωδικοποίησης καναλιού (“Uncoded”) αυξάνει όταν ο SNR του καναλιού αυξάνει, και μάλιστα για υψηλούς σηματοθορυβικούς λόγους βλέπουμε ότι αυτή η διαφορά είναι σχεδόν χαώδης
- Το σχήμα LDPC επιτυγχάνει καλύτερη απόδοση από την περίπτωση “Uncoded” ακόμη από την 1^η επανάληψη του αλγορίθμου
- Η απόδοση των κωδίκων είναι πολύ καλύτερη για κανάλια μετάδοσης AWGN και RICE σε σχέση με τα κανάλια RAYLEIGH
- Ο μέγιστος αριθμός επαναλήψεων που επιτρέπουμε να πραγματοποιεί ο αποκωδικοποιητής κατά την λειτουργία του επηρεάζει την απόδοση του συστήματος. Όσο αυξάνει, τόσο βελτιώνεται η απόδοσης του συστήματος έως την τιμή $\text{max_iteration} = 40$. Κατόπιν αυτής της τιμής, πρακτικά η απόδοση του συστήματος μένει σχεδόν αμετάβλητη.
- Ο αλγόριθμος συγκλίνει σχετικά γρήγορα, όπως δείχνει το Σχήμα 5.4 μετά από περίπου 5 επαναλήψεις ο αλγόριθμος καταλήγει στον υπολογισμό του τελικού μεταδιδόμενου μηνύματος.

Παράρτημα – Κώδικας MATLAB

Αρχείο ldpc_main.m

```
% LDPC Code
clear all;
%load parity check matrix H
load 128x256regular.mat H
[rowsh,colsh]=size(H);
n=colsh ; %codeword length
k=n-rowsh ; %message length
A=H(1:n-k,1:n-k);
B=H(1:n-k,n-k+1:n) ; %H=[A B]
I=eye(k); %identity matrix
%I=eye(n-k);
load inverse_of_A.mat invA %inverse of A, over the binary field
p_aux=bin_multi(invA,B);
P=p_aux.';
%get the corresponding generator matrix G
G=[P I];
%run various simulations for various SNR (or Eb/Nc) values
number_of_Packets_per_SNR=1000;
SNRrange=[0:0.5:4];
mean_error_rate=zeros(1, length(SNRrange));
for SNRrange_index=1:length(SNRrange)
if (SNRrange(SNRrange_index)>3.5)
number_of_Packets_per_SNR=5000;
end
error_rate=zeros(1, number_of_Packets_per_SNR);
for packetnumber=1:number_of_Packets_per_SNR
m=randint(1,k); %random input message
c=bin_multi(m,G); %codeword production
t=bpsk(c); %transmitted vector after BPSK mod
%*****
SNR=SNRrange(SNRrange_index);
No=10^(-SNR/10);
sigma=sqrt(No/2);
%fprintf('SNR: %.4f dB \n',SNR)
r=awgn(t,SNR); %Gaussian noise
%*****
%decoding
%init
max_iter=80;
count_iter=0;
f1=1./(1+exp(-2*r/sigma^2));
f0=1-f1;
fnd=find(H==1);
[ii,jj]=ind2sub([rowsh,colsh],fnd);
len=length(ii);
%Mackay
Q0=zeros(rowsh,colsh);
Q1=zeros(rowsh,colsh);
dQ=zeros(rowsh,colsh);
dR=zeros(rowsh,colsh);
R0=zeros(rowsh,colsh);
R1=zeros(rowsh,colsh);
```



```

q0=zeros(1,n);
q1=zeros(1,n);
for i=1:len
Q0(ii(i),jj(i))=f0(jj(i));
end
for i=1:len
Q1(ii(i),jj(i))=f1(jj(i));
end
while (count_iter<max_iter)
dQ=Q0-Q1;
for i=1:len
dR(ii(i),jj(i))=produ(dQ,ii,jj,i); %HORIZONTAL STEP
end
for i=1:len
R0(ii(i),jj(i))=(1/2)*(1+dR(ii(i),jj(i)));
end
for i=1:len
R1(ii(i),jj(i))=(1/2)*(1-dR(ii(i),jj(i)));
end
% VERTICAL
for i=1:len
Q0(ii(i),jj(i))=f0(jj(i))*vert_prod(R0,ii,jj,i);
end
for i=1:len
Q1(ii(i),jj(i))=f1(jj(i))*vert_prod(R1,ii,jj,i);
end
%VERTICAL_END
for i=1:len
[Q0(ii(i),jj(i)),Q1(ii(i),jj(i))]=normalize(Q0(ii(i),jj(i)),Q1(ii(i),
jj(i)));
end
% a posteriori symbol probabilities q0,q1
for i=1:n
q0(i)= f0(i)*column_product(R0,ii,jj,i);
end
for i=1:n
q1(i)= f1(i)*column_product(R1,ii,jj,i);
[q0(i),q1(i)]=normalize(q0(i),q1(i));
end
vector_est=estimate(q0,q1,n);
vector_est;
if bin_multi(vector_est,H')==0
break;
end;
count_iter=count_iter+1;
end
disp('iterations: ');disp(count_iter);
vector_est;
error_counter=0;
final_msg=vector_est(n-k+1:n);
for p=1:k
if final_msg(p)~=m(p)
disp('error in position ');disp(p);
error_counter=error_counter+1;
end
end
%disp('iterations: ');disp(count_iter);
sprintf('Errors: %d %d dB \n',error_counter,
SNRrange(SNRrange_index))
error_rate(1, packetnumber)=error_counter/k;
end

```

```

sprintf('For %d mean error rate is ', 'SNR' )
SNRrange(SNRrange_index);
mean_error_rate(1, SNRrange_index)=double(mean(error_rate));
mean_error_rate(1, SNRrange_index);
disp('new SNR');
end
plot(SNRrange, mean_error_rate);
title('LDPC - BPSK - AWGN');
xlabel('SNR (dB)');
ylabel('BER')

```

Apêndice bin_multi.m

```

function res=bin_multi(A,B)
% sum=0;
[ra,ca]=size(A); %get A dimensions
[rb,cb]=size(B); %get B dimensions
res=zeros(ra,cb);
if (ca==rb) ,
for i=1:ra
for k=1:cb
for j=1:ca
res(i,k)=xor(res(i,k),A(i,j)*B(j,k));
end;
end;
end;
elseif (ca~=rb)
disp('error in matrix multiplication');
else disp('...');
end;

```

Apêndice column_product.m

```

function res=bin_multi(A,B)
% sum=0;
[ra,ca]=size(A); %get A dimensions
[rb,cb]=size(B); %get B dimensions
res=zeros(ra,cb);
if (ca==rb) ,
for i=1:ra
for k=1:cb
for j=1:ca
res(i,k)=xor(res(i,k),A(i,j)*B(j,k));
end;
end;
end;
elseif (ca~=rb)
disp('error in matrix multiplication');
else disp('...');
end;

```

Αρχείο estimate.m

```
% final vector estimation, in current iteration
% if q0(j)>q1(j) -> d(j)=0, else d(j)=1
function d=estimate(q0,q1,n)
for j=1:n
if q0(j)>q1(j)
d(j)=0;
else
d(j)=1;
end;
end;
```

Αρχείο f_minus.m

```
%look-up ffunction
lup = f_minus(x)
lup=abs(log(1-exp(-x)));
end
```

Αρχείο f_plus.m

```
%look-up f+
function lup = f_plus(x)
lup=abs(log(1+exp(-x)));
end
```

Αρχείο myXor.m

```
% modulo-2 addition
function mxr=myXor(a,b)
if ((a==0) && (b==0)), mxr=0;
elseif ((a==0) && (b==1)), mxr=1;
elseif ((a==1) && (b==0)), mxr=1;
elseif ((a==1) && (b==1)), mxr=0;
else disp('error...');
end;
```

Αρχείο normalize.m

```
% normalize Q0,Q1 so that Q0(i,j)+Q1(i,j)=1
function [a,b]=normalize(inp1,inp2)
aux=1./(eps+(inp1+inp2));
a=inp1*aux;
b=inp2*aux;
```

Αρχείο bpsk.m

```
%BPSK modulation / mapping
function tr=bpsk(c)
for i=1:length(c)
136
if c(i)==0
tr(i)=-1;
else
tr(i)=c(i);
end
end
```

Αρχείο produ.m

```
% product calculation, sum-product algorithm
% $\delta R(i,j)=\prod \delta Q(i',j)$ ,  $j'$  is member  $N(i)$ 
% HORIZONTAL STEP
function product=produ(dQ,ii,jj,i)
product=1;
for j=1:length(ii)
if ii(j)==ii(i) %same row
if jj(j)~=jj(i) % j
product=product*dQ(ii(j),jj(j));
end;
end;
end;
```

Αρχείο vert_prod.m

```
% VERTICAL STEP
%SUM-PRODUCT ALGORITHM, MACKAY'S SIMPLIFICATION
% $Q0(i,j)=f0(j) \prod R0(i',j)$ ,  $i'$  is member  $M(j)$ 
%same with Q1
function vp=vert_prod(R,ii,jj,i)
vp=1;
for j=1:length(ii)
if jj(j)==jj(i) %same column
137
if ii(j)~=ii(i) % i
vp=vp*R(ii(j),jj(j));
end;
end;
end;
```

Αρχείο vertical_sum.m

```
% VERTICAL STEP
%SUM-PRODUCT ALGORITHM, MACKAY'S SIMPLIFICATION in logarithmic form
% $LQ0(i,j)=L f0(j)+ \sum LR0(i',j)$ ,  $i'$  is member  $M(j)$ 
%same with LQ1
```

```
function vs=vertical_sum_log(R,ii,jj,i)
vs=0;
for j=1:length(ii)
if jj(j)==jj(i) %same column
if ii(j)~=ii(i) % \i
vs=vs+R(ii(j),jj(j));
end;
end;
end;
```

ΒΙΒΛΙΟΓΡΑΦΙΑ

- [1] R.G. Gallager, “Low-Density Parity-Check Codes”. MIT Press, Cambridge, MA, 1963.
- [2] B. Sklar, “Digital Communications, Fundamentals and Applications”, Prentice-Hall, 1988.
- [3] J. Moreira, P. Farrell, “Essentials of Error Control Coding”, Wiley, 2006.
- [4] William H. Tranter, K. Sam Shanmugan, Theodore S. Rappaport, Kurt L. Kosbar, “Principles of Communications Systems Simulation with Wireless Applications”, Prentice Hall Communications Engineering and Emerging Technologies Series, 2004.
- [5] Χ. Καψάλης, Π. Κωπτή, “Δορυφορικές Επικοινωνίες”, Εκδόσεις Τζιόλα, 2006.
- [6] MacKay, D. J. C. and Neal, R. M., “Near Shannon limit performance of low density parity check codes,” *Electron. Lett.*, vol. 33, no. 6, March 13, 1997.
- [7] A. Morello and V. Mignone, “DVB-S2: The second generation standard for satellite broad-band services,” *Proc. IEEE*, vol.94, pp.210–227, Jan. 2006.
- [8] Tang, H., Xu, J., Kou, Y., Lin, S. and Abdel-Ghaffar, K., “On algebraic construction of Gallager and circulant low-density parity-check codes,” *IEEE Trans. Inf. Theory*, vol. 50, no. 6, pp. 1269–1279, June 2004.
- [9] T. Brack, M. Alles, T. Lehnigk-Emden, F. Kienle, N. When, N.E. L’Insalata, F. Rossi, M. Rovini, L. Fanucci, “Low Complexity LDPC Code Decoders for Next Generation Standards”.