

ΤΕΙ ΠΕΛΟΠΟΝΗΣΣΟΥ
ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΠΛΗΡΟΦΟΡΙΚΗΣ
ΣΧΟΛΗ ΤΕΧΝΟΛΟΓΙΚΩΝ ΕΦΑΡΜΟΓΩΝ

Πτυχιακή εργασία

ΘΕΜΑ:

« Turbo Codes σε Επικοινωνίες 3G και 4G »

Καρανδινάκης Εμμανουήλ

A.M: 2006155

Επιβλέπων Καθηγητής:

ΠΕΡΙΕΧΟΜΕΝΑ

Εισαγωγή.....	4
1. Turbo codes, η αρχή.....	6
1.1 Κωδικοποίηση.....	19
1.2 Πρώτη Αποκωδικοποίηση.....	24
1.3 Τι σημαίνει turbo στους κώδικες Turbo.	26
1.4 Παράδειγμα απόδοσης.....	28
1.5 Αποκωδικοποίηση APP	29
2. Συστήματα επικοινωνίας 3G.....	35
2.1 Turbo κώδικες σε συστήματα 3G.....	35
2.2 Προτεινόμενος κώδικας για συνδυασμό με τον αναδιατάκτη.....	37
2.3 Θεωρητική προσομείωση.....	43
3. Συστήματα επικοινωνίας 4G.....	45
3.1 Turbo Κώδικες σε συστήματα 4G.....	47
3.2 Δυαδικός Κώδικας Turbo σε πρότυπο 3GPP LTE.....	48
3.3 Αλγόριθμος Αποκωδικοποίησης.....	50
4. Αρχιτεκτονική αποκωδικοποιητών Turbo πολλαπλών προτύπων.....	54
4.1 Σχετική έρευνα για ευέλικτες αρχιτεκτονικές Turbo αποκωδικοποιητών.....	54
4.1.1 Αποκωδικοποίηση των Δυαδικών Κωδικών Turbo στο τομέα καταγραφής.....	55
4.1.2 Αποκωδικοποίηση μέσω μονοεπίπεδου προβλέψιμου μετασχηματισμού (τετραγωνική ρίζα)	56
4.1.3 Αποκωδικοποίηση διπλών δυαδικών κωδικών Turbo (τετραγωνική ρίζα)	58

ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΠΛΗΡΟΦΟΡΙΚΗΣ

4.1.4	Ενιαία Αρχιτεκτονική του Αποκωδικοποιητή Log-MAP.....	60
4.1.5	Βελτιστοποίηση Χώρου.....	65
4.2	Εσωτερική αρχιτεκτονική διεμπλοκέα Turbo.....	66
4.2.1	Contention-free Διεμπλοκείς.....	66
4.2.2	Τετραγωνικός πολυωνυμικός μεταθετικός διεμπλοκέας στο πρότυπο 3GPP LTE.....	67
4.2.3	Πρακτική εφαρμογή του Τετραγωνικού Πολυωνυμικού Μεταθετικού διεμπλοκέα.....	69
4.2.4	Διεμπλοκέας Σχεδόν Τακτικής Μετάθεσης στο IEEE 802.16e πρότυπο WiMax.....	70
4.2.5	Πρακτική εφαρμογή του Διεμπλοκέα ARP.....	71
4.3	Παράλληλη αποκωδικοποίηση Turbo.	72
5.	Πειραματική ανάλυση λειτουργίας Turbo codes.....	80
5.1	Περαιτέρω ανάλυση απόδοσης σφάλματος κωδικών Turbo.....	80
5.1.1	Απόδοση BER για frame size $K = 40$	81
5.1.2	Απόδοση BER για frame size $K = 100$	83
5.1.3	Ο BER ως λειτουργία του frame size K (Latency-λανθάνουσα κατάσταση)	85
5.1.4	Απόδοση BER σε κανάλι AWGN για αλγόριθμο max *.....	86
5.1.5	Απόδοση BER σε κανάλι Rayleigh για αλγόριθμο max *	89
5.1.6	Σύγκριση απόδοσης σε κανάλια Rayleigh και AWGN	90
5.1.7	Το BER σε συνάρτηση του αριθμού των επαναλήψεων.....	91
5.2	Ανάλυση εξόδου διαγράμματος.	93
5.2.1	Εξωγενείς πληροφορίες ως συνάρτηση των εκ των προτέρων πληροφοριών.....	94
5.2.2	Τροχιά αποκωδικοποίησης	96
6.1	Περίληψη.....	100
6.2	Συμπεράσματα.....	101
6.3	Βιβλιογραφία.....	102

Εισαγωγή

Δεν συμβαίνει συχνά στον κόσμο της τεχνολογικής έρευνας που μια πρόταση εκ των έσω γίνεται αντικείμενο γλεουασμού. Είναι ακόμα πιο σπάνιο όταν η πρόταση αποδεικνύεται στο τέλος να είναι πραγματικά επαναστατική. Αυτό συνέβη πριν από μία εικοσαετία και πλέον στη Διεθνή Διάσκεψη του 1993 IEEE για τις επικοινωνίες στη Γενεύη της Ελβετίας. Δύο Γάλλοι ηλεκτρολόγοι μηχανικοί, ο Claude Berrou και ο Alain Glavieux, έκαναν έναν καινοτόμο,ριζοσπαστικό όπως αναφέρθηκε,ισχυρισμό: ότι είχε εφεύρει ένα ψηφιακό σύστημα κωδικοποίησης που θα μπορούσε να παρέχει επικοινωνίες σχεδόν χωρίς λάθη σε ρυθμούς δεδομένων και βελτίωση της αποτελεσματικότητας μετάδοσης ισχύος πολύ πέρα από ό, τι θεωρούσαν ποτέ πιθανόν οι περισσότεροι ειδικοί.

Το σχήμα αυτό, υποστήριξαν οι συγγραφείς, θα μπορούσε να διπλασιάσει την απόδοση των δεδομένων για μια δεδομένη ισχύ μετάδοσης ή εναλλακτικά, να επιτύχουν ένα συγκεκριμένο ρυθμό μετάδοσης δεδομένων επικοινωνίας με το μισό της μεταδιδόμενης ενέργειας, ένα τεράστιο κέρδος που θα μπορούσε να αξίζει μια περιουσία σε εταιρείες επικοινωνιών.

Κάποιοι βετεράνοι μηχανικοί επικοινωνιών πίστευαν τα αποτελέσματα. Οι Γάλλοι, και οι δύο καθηγητές στο τμήμα ηλεκτρονικών στο Ecole Nationale Supérieure des Telecommunications de Bretagne στη Βρέστη της Γαλλία, ήταν τότε άγνωστοι στην κοινότητα πληροφορίας-θεωρίας. Πρέπει να είχαν ξεστρατίσει στους υπολογισμούς τους, μερικοί υποστήριξαν. Οι ισχυρισμοί ήταν τόσο εξωφρενικοί που πολλοί ειδικοί δεν μπόηκαν καν στον κόπο να διαβάσουν την διάλεξη που παρουσίασαν οι Berrou και Glavieux.

Όσο απίστευτο και να φαινόταν, σύντομα αποδείχθηκε αληθινό, καθώς και άλλοι ερευνητές άρχισαν να αναπαράγουν τα αποτελέσματα. Ειδικοί στην κωδικοποίηση τότε συνειδητοποίησαν τη σημασία του εν λόγω έργου. Ο Berrou και ο Glavieux είχαν δίκιο, και το σύστημα κωδικοποίησης διόρθωσης σφαλμάτων τους, το οποίο

έκτοτε ονομάστηκε κώδικες turbo(Turbo Codes), έφερε την επανάσταση στον τομέα διόρθωσης σφαλμάτων κωδικοποίησης.

Από μια τεχνολογία που εφαρμόστηκε πρώτη φορά κυρίως σε δορυφορικές συνδέσεις και σε τουλάχιστον ένα σύστημα διαστημικών επικοινωνιών, οι κωδικοί turbo είναι η επικρατέστερη τάση αυτή τη στιγμή μαζί με τις διακλαδώσεις και τις εξελιγμένες τους μορφές. Έχουν ενσωματωθεί στο σύστημα κινητής τηλεφωνίας, εκατομμύρια άνθρωποι καθημερινά χρησιμοποιούν διάφορες μορφές κωδικών Turbo. Αυτό το σύστημα κωδικοποίησης έχει αφήσει τα κινητά τηλέφωνα και άλλες φορητές συσκευές να χειρίζονται δεδομένα πολυμέσων, όπως βίντεο και εικόνες κ.α. σε πάνω από το όριο των θορυβώδη κανάλιων που ήταν σύνηθες των κυψελοειδών επικοινωνιών. Ερευνητές εξέλιξαν τη χρήση των κωδικών turbo για ψηφιακό ήχο και μετάδοση βίντεο, καθώς επίσης και για την αύξηση των ταχυτήτων μετάδοσης δεδομένων σε ενισχυμένες εκδόσεις των δικτύων Wi-Fi. Με τόσο εύρος χρήσης, οι κώδικες turbo βρίσκονται αυτή τη στιγμή στην πρώτη γραμμή της έρευνας των τηλεπικοινωνιών, με εκατοντάδες ομάδων εργασίας, σε εταιρείες και πανεπιστήμια σε όλο τον κόσμο. Ο κατάλογος περιλαμβάνει γίγαντες των τηλεπικοινωνιών, όπως η France Télécom και η NTT DoCoMo, κολοσσούς βαρόμετρα στον χώρο των τηλεπικοινωνιών όπως η Sony, η NEC, η Lucent, η Samsung, η Motorola και η Qualcomm, κατασκευαστές hardware και τσιπ όπως η Broadcom, Conexant, Comtech AHA, STMicroelectronics, Altera, Xilinx και εξειδικευμένες νεοεμφανιζόμενες εταιρίες όπως οι Turbocash, Turboconcept και iCoding.

1. TURBO CODES-Η ΑΡΧΗ

Οι Κώδικες Turbo κάνουν ένα απλό αλλά απίστευτο πράγμα: αφήνουν τους μηχανικούς σχεδιασμού συστημάτων που φτάνουν πολύ κοντά στη λεγόμενη χωρητικότητα καναλιού, την απόλυτα μέγιστη χωρητικότητα, σε bits ανά δευτερόλεπτο, από ένα κανάλι επικοινωνίας για ένα δεδομένο επίπεδο ισχύος στον πομπό. Αυτό το κατώτατο όριο για αξιόπιστες επικοινωνίες ανακαλύφθηκε από τον φημισμένο Claude Shannon, τον λαμπρό ηλεκτρολόγο μηχανικό και μαθηματικό που εργαζόταν στα εργαστήρια τηλεφώνων Bell στο Murray Hill, NJ, και είναι γνωστός ως ο πατέρας της θεωρίας της πληροφορίας (Shannon: Cracking the Channel). Σε ένα έγγραφο ορόσημο το 1948, ο Shannon, ο οποίος πέθανε το 2001, έδειξε ότι με τους σωστούς κωδικούς διόρθωσης σφαλμάτων, τα δεδομένα μπορούν να μεταδοθούν με ταχύτητα μέχρι και την χωρητικότητα του καναλιού, σχεδόν χωρίς λάθη, και με εκπληκτικά χαμηλή ισχύ εκπομπής. Πριν από την εργασία του Shannon, οι μηχανικοί πίστευαν ότι για να μειωθούν τα λάθη τον τομέα των τηλεπικοινωνιών, ήταν απαραίτητο να αυξηθεί η ισχύς μετάδοσης ή να στείλουν το ίδιο μήνυμα επανειλημμένα. Ο Shannon βασικά έδειξε ότι δεν ήταν αναγκαίο να σπαταλάμε τόσο πολύ ενέργεια και χρόνο, αν είχαμε τα σωστά συστήματα κωδικοποίησης. Μετά την ανακάλυψή του, το πεδίο της θεωρίας κωδικοποίησης άνθισε, και ερευνητές ανέπτυξαν αρκετά καλούς κώδικες. Αλλά και πάλι, πριν από τους κωδικούς turbo, ακόμη και οι καλύτερες κώδικες συνήθως απαιτούσαν περισσότερες από δύο φορές την ισχύ εκπομπής που ο νόμος του Shannon εανέφερε πως ήταν αναγκαία για να φτάσουν σε ένα συγκεκριμένο επίπεδο αξιοπιστίας, μια τεράστια σπατάλη ενέργειας. Το χάσμα ανάμεσα στο πρακτικό και το ιδανικό- το οποίο μετράται σε ντεσιμπέλ- ένα κλάσμα(αναλογία) μεταξύ του επιπέδου του σήματος και του επίπεδο θορύβου σε λογαριθμική κλίμακα, ήταν περίπου 3,5 dB. Για να φτάσουν κοντά σε αυτό, οι μηχανικοί χρειάζονταν πιο περίτεχνους κωδικούς.

Αυτός ήταν ο στόχος που διαρκούσε περισσότερο από τέσσερις δεκαετίες, μέχρι ο Berrou και ο Glavieux έκαναν την ανακάλυψή τους στις αρχές του 1990. Όταν

εισήγαγαν κώδικες turbo το 1993, έδειξαν ότι ήταν δυνατό να πλησιάσουν σε ένα εκκληκτικό νούμερο 0,5 dB από το όριο του Shannon, για λόγο δυαδικών ψηφιακών σφαλμάτων, ένα στις 100 000. Σήμερα, οι κώδικες turbo εξακολουθούν να μειώνουν αυτό το εκκληκτικό νούμερο .

Η λύση για την αντιμετώπιση του θορύβου που μαστίζει όλα τα κανάλια επικοινωνίας, σύμφωνα με την θεμελιώδη εργασία του Shannon, ήταν να διαιρέσουμε τα δεδομένα σε σειρές από bits και να προσθέσουμε σε κάθε χορδή μια σειρά από επιπλέον bits τα λεγόμενα bits ισοτιμίας, τα οποία θα βοηθήσουν στον εντοπισμό και τη διόρθωση λαθών στο τέλος της λήψης. Η προκύπτουσα ομάδα των bits-τα bits δεδομένων και τα bits ισοτιμίας -ονομάζονται κωδική λέξη, και τυπικά αντιπροσωπεύουν ένα μπλοκ από χαρακτήρες, μερικά εικονοστοιχεία μιας εικόνας, ένα δείγμα της φωνής, ή κάποιο άλλο κομμάτι των δεδομένων.

Ο Shannon έδειξε ότι με τη σωστή συλλογή των κωδικών λέξεων-με το σωστό κώδικα με άλλα λόγια, ήταν δυνατό να επιτευχθεί η χωρητικότητα του καναλιού. Στη συνέχεια όμως, ποιος κώδικας θα μπορούσε να το κάνει; "Ο Shannon άφησε αναπάντητο το ερώτημα του ποιος θαεφεύρει κώδικες" είπε ο David Forney, καθηγητής ηλεκτρολόγος-μηχανικός στο Cambridge με βάση το Τεχνολογικό Ινστιτούτο της Μασαχουσέτης (MIT) και μέλος της IEEE.Ο Shannon απέδειξε μαθηματικά ότι η κωδικοποίηση ήταν το μέσο για να φθάσει, αλλά δεν είχε δείξει ακριβώς πώς να κατασκευάσει αυτούς τους ικανούς να πλησιάσουν την χωρητικότητα, κωδικούς. Το έργο του, παρ 'όλα αυτά, περιείχε πολύτιμες ενδείξεις.Ο Shannon σκεφτηκε κωδικές λέξεις ως σημεία στο χώρο. Για παράδειγμα, η κωδική λέξη 011 μπορεί να θεωρηθεί ως σημείο σε ένα τρισδιάστατο χώρο με συντεταγμένες $x = 0$, $y = 1$, και $z = 1$. Οι κωδικές λέξεις με περισσότερα από τρία bits είναι σημεία στο υπερδιάστημα. Ο θόρυβος μπορεί να αλλάξει κομμάτια μιας κωδική λέξη και, επομένως, τις συντεταγμένες της, μετατοπίζοντας το σημείο στο χώρο. Αν δύο σημεία βρίσκονται κοντά το ένα στο άλλο και το ένα επηρεάζεται από το θόρυβο, το σημείο αυτό μπορεί να πέσει ακριβώς επάνω από το άλλο, με αποτέλεσμα κάποιο σφάλμα στην αποκωδικοποίηση. Ως εκ τούτου, όσο μεγαλύτερες οι διαφορές στις κωδικές λέξεις- μεγαλύτερη απόσταση μεταξύ τους είναι, τόσο πιο δύσκολο είναι για το θόρυβο να προκαλέσει λάθη.

Για την επίτευξη της χωρητικότητας,ο Shannon απέδειξε ότι θα πρέπει να επιλέξουμε τυχαία επιμηκής κωδικές λέξεις. Με άλλα λόγια, πηγαίνοντας πίσω στην χωρική αναλογία του, αν θα μπορούσαν να γίνουν οι κωδικές λέξεις όσο τυχαίες και όσο

μακρυνές θέλαμε, θα μπορούσαμε να θέσουμε τα σημεία αυθαίρετα μακριά το κάθε ένα από το άλλο στο χώρο. Δεν θα υπήρχε ουσιαστικά καμία πιθανότητα ένα σημείο να πέσει λανθασμένα πάνω σε άλλο. Δυστυχώς, τέτοιες μεγάλες, τυχαίες κωδικές λέξεις δεν είναι πρακτικές: πρώτον, επειδή υπάρχει ένας αστρονομικός αριθμός κωδικών λέξεων, δεύτερον επειδή αυτός ο κώδικας θα είναι εξαιρετικά αργός για να χρησιμοποιηθεί καθώς θα μεταδίδονταν πολλά bits για μία μόνο κωδική λέξη. Ακόμα, η τυχαία φύση ενός κώδικα θα αποδειχθεί ότι είναι κρίσιμης σημασίας για τους κώδικες turbo.

Οι ειδικοί στην κωδικοποίηση έβάλαν στην άκρη τους ιδανικούς τυχαίους κώδικες του Shannon, καθώς επικεντρώθηκαν στην ανάπτυξη πρακτικών κωδικών που θα μπορούσαν να εφαρμοστούν σε πραγματικά συστήματα. Σύντομα άρχισαν να αναπτύσσουν καλούς κωδικούς επιλέγοντας έξυπνα, bits ισοτιμίας που περιόρισαν τις κωδικές λέξεις σε συγκεκριμένες τιμές, κάνοντας αυτές τις κωδικές λέξεις απίθανο να μπερδευτούν με άλλες.

Για παράδειγμα, ας υποθέσουμε ότι έχουμε μια κωδική λέξη οκτώ-bit (επτά bits δεδομένων συν ένα bit ισοτιμίας). Εάν επιμήνουμε ακόμη ότι όλες οι κωδικές λέξεις έχουν ίσο αριθμό 1s, καθιστώντας το επιπλέον bit ισοτιμίας ένα 1, εάν είναι απαραίτητο για να πληροί αυτή την προϋπόθεση. Τώρα, αν κάποιος από τα οκτώ bits μεταβάλλεται από το θόρυβο, συμπεριλαμβανομένου του bit ισοτιμίας, ο δέκτης γνωρίζει πως υπήρξε ένα λάθος, επειδή ο αριθμός των ισοτιμιών δεν θα είναι σωστόι, θα υπάρχει περίεργος αριθμός 1s.

Αυτό το βασικό σύστημα μπορεί να ανιχνεύσει ένα σφάλμα, αλλά δεν μπορεί να το διορθώσει, δεν ξέρει ποιο κομμάτι έχει ζημειωθεί. Για να διορθωθούν τα λάθη, θα πρέπει να έχουμε περισσότερα bits ισοτιμίας. Οι ειδικοί στην κωδικοποίηση έχουν καταλήξει σε πολλούς και όλο και πιο περίπλοκους τρόπους δημιουργίας bits ισοτιμίας. Block κώδικες, κώδικες Hamming, κώδικες Reed-Solomon, και συνελκτικοί κώδικες (οι οποίοι μπορούν να λειτουργήσουν με Turbo codes) χρησιμοποιούνται ευρέως και επιτυγχάνουν πολύ χαμηλά ποσοστά σφάλματος.

Παρ' όλα αυτά, το πρόβλημα της υπολογιστικής πολυπλοκότητας κληρονομήσε τους ειδικούς κωδικοποίησης και μαστίζε όλους αυτούς τους τρόπους κωδικοποίησης. Το

πρόβλημα της πολυπλοκότητας αναδύεται όταν βρίσκουμε το κόστος ενός κώδικα όσον αφορά το ποσό των απαιτούμενων υπολογισμών για την αποκωδικοποίηση των δεδομένων μας. Όσο πιο κοντά μπορούμε να πάμε στο όριο του Shannon, τόσο πιο περίπλοκη γίνεται αυτή η διαδικασία, γιατί θα πρέπει να υπάρχουν περισσότερα bits ισοτιμίας και οι κωδικές λέξεις γίνονται όλο και πιο μακρυνές.

Για κωδικές λέξεις με μόλις 3 bits, για παράδειγμα, έχουμε συνολικά μόνο 23, ή 8, κωδικές λέξεις. Για να προσεγγιστεί η χωρητικότητα, ωστόσο, μπορεί να χρειαστούν κωδικές λέξεις με, ας πούμε, 1000 bits, και ως εκ τούτου ο αποκωδικοποιητής θα πρέπει να αναζητήσει μέσα από μία εξαιρετικά μεγάλη συλλογή από 21000, περίπου 10301 κωδικές λέξεις. Για σύγκριση, ο εκτιμώμενος αριθμός των ατόμων στο ορατό σύμπαν είναι περίπου 1080.

Το αποτέλεσμα ήταν ότι, εάν ξεκινούσαμε να εκμεταλλευόμαστε τους καλύτερους υφιστάμενους κωδικές εκείνης της περιόδου ως στρατηγική μας για την επίτευξη αυθαίρετα αξιόπιστων επικοινωνιών στο όριο του Shannon, θα είμασταν καταδικασμένοι να αποτύχουμε. "Η υπολογιστική πολυπλοκότητα είναι απλά αστρονομική», λέει το μέλος του IEEE, R. Michael Tanner, καθηγητής ηλεκτρικών και μηχανικών υπολογιστών και κοσμήτορας στο Πανεπιστήμιο του Ιλινόις στο Σικάγο. "Οι κωδικοί αυτοί δεν έχουν τη δυνατότητα να το κάνουν." Πώς θα μπορούσαν οι ερευνητές ξεπεράσουν αυτό το εμπόδιο; Ήταν μάταιο, κατέληξαν κάποιοι στα τέλη της δεκαετίας του 1970.

Οι Κώδικες Turbo ήρθαν για να λυθεί το πρόβλημα της πολυπλοκότητας, διαχωρίζοντας το πρόβλημα σε πιο διαχειρίσιμες συνιστώσες. Αντί ενός ενιαίου κωδικοποιητή στον μεταδότη και ένα μονό αποκωδικοποιητή στο δέκτη, οι κώδικες turbo χρησιμοποιούν δύο κωδικοποιητές στο ένα άκρο και δύο αποκωδικοποιητές στο άλλο.

Οι ερευνητές είχαν κατανοήσει στα τέλη της δεκαετίας του 1960 πως το πέρασμα των δεδομένων μέσω δύο κωδικοποιητών σε σειρά θα μπορούσε να βελτιώσει την ικανότητα αντίστασης σφραγμάτων μιας μετάδοσης- για έναν τέτοιο συνδυασμό κωδικοποιητών, το σύνολο είναι μεγαλύτερο από το άθροισμα των μερών. Οι κώδικες turbo χρησιμοποιούν δύο κωδικοποιητές που λειτουργούν σε συνεργεία-δεν λειτουργούν σε σειρά, αλλά παράλληλα.

Η διαδικασία turbo ξεκινά με τρία αντίγραφα του μπλοκ δεδομένων που πρόκειται να μεταδοθούν. Το πρώτο αντίγραφο πηγαίνει σε έναν από τους κωδικοποιητές, όπου ένας συνελκτικός κώδικας παίρνει τα bits δεδομένων και υπολογίζει δυαδικά ψηφία ισοτιμίας από αυτούς. Το δεύτερο αντίγραφο πηγαίνει στο δεύτερο κωδικοποιητή, το οποίο περιέχει έναν πανομοιότυπο συνελκτικό κώδικα. Αυτός ο δεύτερος κωδικοποιητής δεν παίρνει την αρχική σειρά των bits, αλλά μάλλον μια σειρά με τα bits σε μια άλλη σειρά, ανακατεμένα από ένα σύστημα που ονομάζεται αναδιατάκτης ή διεμπλοκέας. Αυτός ο κωδικοποιητής τότε διαβάζει αυτά τα κωδικοποιημένα δυαδικά ψηφία δεδομένων και υπολογίζει bits ισοτιμίας από αυτούς. Τέλος, ο πομπός παίρνει το τρίτο αντίγραφο των αρχικών δεδομένων και το στέλνει, μαζί με τις δύο σειρές των bits ισοτιμίας, στο κανάλι.

Η αναδιάταξη των bits στον διεμπλοκέα είναι το βασικό βήμα στην όλη διαδικασία. Βασικά, αυτή η μετάθεση φέρνει μεγαλύτερη ποικιλομορφία στις κωδικές λέξεις; στη χωρική αναλογία, ωθεί τα σημεία μακρύτερα χώρια στο χώρο. "Ο ρόλος της μετάθεσης είναι να εισαγάγει κάποια τυχαία συμπεριφορά στον κώδικα», λέει ο Βεργου. Με άλλα λόγια, ο διεμπλοκέας προσθέτει ένα τυχαίο χαρακτήρα στην μεταδιδόμενη πληροφορία, όπως θα έκαναν και οι τυχαίοι κώδικες Shannon.

Εν συνεχεία όμως οι κώδικες turbo, όπως και κάθε άλλος κώδικας με τεράστιο αριθμό κωδικών λέξεων, θα κατέληγαν επίσης στο αδιέξοδο της υπολογιστικής πολυπλοκότητας. Στην πραγματικότητα, οι κωδικοί turbo συνήθως λειτουργούν με κωδικές λέξεις που έχουν περίπου χίλια bits, ένας αρκετά μη ευέλικτος αριθμός. Θα ήταν αβοήθητου αν είχαμε έναν μόνο αποκωδικοποιητή στο δέκτη. Αλλά οι κώδικες turbo χρησιμοποιούν δύο συστατικούς αποκωδικοποιητές που συνεργάζονται για να παρακάμψουν το πρόβλημα της πολυπλοκότητας.

Ο ρόλος του κάθε αποκωδικοποιητή είναι να πάρει τα δεδομένα, τα οποία θα μπορούσαν να έχουν καταστραφεί από τον θόρυβο κατά μήκος του καναλιού, και να αποφασίσει ποια είναι η πιο πιθανή τιμή, 0 ή 1, για κάθε bit. Κατά μία έννοια, τη λήψη αποφάσεων σχετικά με την αξία του κάθε bit είναι σαν να έπρεπε να μαντέψουμε αν βρέχει ή όχι έξω. Ας υποθέσουμε ότι δεν μπορεί κάποιος να κοιτάξει έξω από ένα παράθυρο και να μην ακούει ήχους απ'έξω. Σε αυτή την περίπτωση, δεν έχει ουσιαστικά καμία ιδέα, και μπορεί απλά να το παίξει κορώνα γράμματα με ένα κέρμα και να κάνει την εικασία του. Αλλά τι γίνεται αν ελέγξει τις προβλέψεις του

καιρου πχ στην τηλεοραση και λέει για βροχή; Επίσης, τι γίνεται αν ξαφνικά ακούσει βροντή; Τα γεγονότα αυτά επηρεάζουν την εικασία του. Τώρα μπορεί να μαντέψει καλύτερα από ό, τι απλώς να ρίχνεις ένα κέρμα; θα πρέπει μάλλον να πει ότι υπάρχει μια καλή πιθανότητα να βρέξει και να πάρει ομπρέλα μαζί του.

Κάθε αποκωδικοποιητής turbo μετράει επίσης για «ενδείξεις» που τον βοηθούν να μαντέψει, εάν το λαμβανόμενο bit είναι 0 ή 1. Πρώτον, επιθεωρεί το αναλογικό επίπεδο σήματος των λαμβανόμενων bits. Ενώ πολλά συστήματα αποκωδικοποίησης μετατρέπουν το λαμβανόμενο σήμα είτε σε 0 ή 1, πετώντας πολύτιμες πληροφορίες, επειδή το αναλογικό σήμα έχει διακυμάνσεις που μπορούν να μας πουν περισσότερα για κάθε bit- ο αποκωδικοποιητής turbo μετατρέπει το σήμα σε ακέραιους αριθμούς που μετρούν πόσο σίγουροι μπορεί να είμαστε ότι ένα bit είναι ένα 0 ή ένα 1.

Επιπλέον, ο αποκωδικοποιητής κοιτάζει τα δυαδικά ψηφία ισοτιμίας του, τα οποία του λενε εάν τα ληφθέντα δεδομένα φαίνονται άθικτα ή έχουν σφάλματα.

Το αποτέλεσμα αυτής της ανάλυσης είναι ουσιαστικά μια ενημερωμένη εικασία για κάθε κομμάτι. "Το τι κανουν οι turbo κώδικες εσωτερικά είναι το να καταλήξουν σε αποφάσεις σχετικά με τα bits και την αξιοπιστία των αποφάσεων αυτών πως είναι σωστές κάπως" λέει ο David Garrett, ένας ερευνητής στο εργαστήριο ασύρματης έρευνας στα Bell Labs, μέρος της Lucent Technologies, στο Murray Hill, NJ. Αυτά τα bit αξιοπιστίας εκφράζονται ως αριθμοί, που ονομάζονται αναλογίες πιθανοφάνειας (log-likelihood ratios- LLRs), που μπορεί να ποικίλλουν, για παράδειγμα, μεταξύ -7 και +7. Μια αναλογία +7 σημαίνει ότι ο αποκωδικοποιητής είναι σχεδόν εντελώς σίγουρος το bit είναι ένα 1; ένα -5 σημαίνει ότι ο αποκωδικοποιητής σκέφτεται το bit είναι ένα 0, αλλά δεν είναι απολύτως πεπεισμένος. (Τα πραγματικά συστήματα έχουν συνήθως μεγαλύτερα χρονικά διαστήματα, όπως -127 έως 127).

Παρόλο που το επίπεδο του σήματος και οι ελέγχοι ισοτιμίας είναι χρήσιμες ενδείξεις, δεν είναι αρκετά. Ένας μονός αποκωδικοποιητής δεν μπορεί ακόμα να πάρει σωστές αποφάσεις σε όλες τις περιπτώσεις σχετικά με τα μεταδιδόμενα bits και συχνά θα καταλήξει με λάθος σειρά των bits, ο αποκωδικοποιητής χάνεται σε ένα σύμπαν κωδικών λέξεων, και η κωδική λέξη που επιλέγει ως τα αποκωδικοποιημένα δεδομένα δεν είναι πάντα η σωστή. Αυτός είναι ο λόγος για τον οποίο ένας αποκωδικοποιητής από μόνος του δεν μπορεί να κάνει τη δουλειά.

Αποδεικνύεται όμως ότι οι πληροφορίες αξιοπιστίας ενός αποκωδικοποιητή είναι χρήσιμες σε άλλον και το αντίστροφο, επειδή οι δύο σειρές των bits ισοτιμίας αναφέρονται στα ίδια ακριβώς στοιχεία; απλά τα bits διατάσσονται σε διαφορετική σειρά. Έτσι, οι δύο αποκωδικοποιητές προσπαθούν να λύσουν το ίδιο πρόβλημα, αλλά κοιτάζοντας το από διαφορετικές οπτικές γωνίες.

Οι δύο αποκωδικοποιητές, στη συνέχεια, μπορούν να ανταλλάξουν την αξιοπιστία των πληροφοριών με έναν επαναληπτικό τρόπο για να βελτιώσουν τη δική τους αποκωδικοποίηση. Το μόνο που έχουν να κάνουν, πριν από την εναλλαγή των σειρών αξιοπιστίας, είναι να οργανώσουν το περιεχόμενο των σειρών με τη σειρά που χρειάζεται κάθε αποκωδικοποιητής. Έτσι, ένα bit που έντονα ανιχνεύεται ως 1 σε ένα αποκωδικοποιητή, για παράδειγμα, επηρεάζει τη γνώμη του άλλου αποκωδικοποιητή στο αντίστοιχο bit.

Στο παράδειγμα με τη βροχή φανταστείτε να δείτε ένας συνάδελφο να βγαίνει έξω που κρατώντας μια ομπρέλα. Είναι ένα πολύτιμο πρόσθετο κομμάτι πληροφορίας που θα μπορούσε να επηρεάσει την εικασία σας. Στην περίπτωση των αποκωδικοποιητών Turbo, τώρα κάθε αποκωδικοποιητής έχει όχι μόνο τη δική του «γνώμη», έχει επίσης μια «εξωτερική γνώμη» για να τον βοηθήσει να καταλήξει σε μια απόφαση για κάθε bit. «Είναι σαν ένα τζίνι σας είχε δώσει τις πληροφορίες," λέει ο Gerhard Kramer, ένας ερευνητής στο ερευνητικό κέντρο μαθηματικών επιστημών στα Bell Labs. Αυτό το τζίνι ψιθυρίζει στο αυτί σας πόσο σίγουροι θα πρέπει να είστε για το εάν ένα bit της εισαγωγής είναι 1 ή 0, λέει, και που σας βοηθά να αποκωδικοποιήσετε αυτό το bit.

Στην καρδιά της κωδικοποίησης turbo είναι αυτή η επαναληπτική διαδικασία, κατά την οποία κάθε συνιστώμενος αποκωδικοποιητής εκμεταλλεύεται την εργασία του άλλου σε ένα προηγούμενο στάδιο αποκωδικοποίησης. Μετά από έναν ορισμένο αριθμό επαναλήψεων, συνήθως 4 έως 10, και οι δύο αποκωδικοποιητές αρχίζουν να συμφωνούν σε όλα τα bits. Αυτό σημαίνει ότι οι αποκωδικοποιητές δεν έχουν χαθεί πια σε ένα σύμπαν κωδικών λέξεων; έχουν ξεπεράσει το φράγμα πολυπλοκότητας.

«Είναι μια λύση διαίρει και βασίλευε», λέει ο Robert J. McEliece, καθηγητής ηλεκτρολογικής μηχανολογίας στο Ινστιτούτο Τεχνολογίας της Καλιφόρνια, στην Πασαντένα, και επιστημονικός συνεργάτης του IEEE. «Έσπασε το πρόβλημα σε δύο

μικρότερα κομμάτια, έλυσε τα κομμάτια, και στη συνέχεια ένωσε τα κομμάτια όλα μαζί πάλι.

Ένας άλλος τρόπος κατανόησης σχετικά με τη διαδικασία αποκωδικοποίησης turbo είναι από η ιδέα των σταυρόλεξων, όπως λέει ο Βεργου. Φανταστείτε ότι ο Γιώργος λ.χ έλυσε ένα σταυρόλεξο και ήθελε να στείλει τη λύση στον Πέτρο. Διαμέσου ενός αθόρυβου καναλιού, θα ήταν αρκετό για να στείλει τον πίνακα με τις λέξεις. Αλλά διαμέσου ενός θορυβώδους καναλιού, τα γράμματα στον πίνακα έχουν μπερδευτεί εξαιτίας του θορύβου. Όταν ο Πέτρος λαμβάνει το σταυρόλεξο, πολλές λέξεις δεν βγάζουν νόημα. Για να βοηθήσει τον Πέτρο να διορθώσει τα λάθη, ο Γιώργος μπορεί να του στείλει τα στοιχεία για τις οριζόντιες και κάθετες λέξεις. Αυτό είναι περιττή πληροφορία, δεδομένου ότι το σταυρόλεξο έχει ήδη λυθεί, αλλά ωστόσο βοηθάει τον Πέτρο, επειδή όπως γίνεται με τα δυαδικά ψηφία ισοτιμίας, επιβάλλει περιορισμούς σχετικά με τις λέξεις που μπορούν να τεθούν σε σειρά. Είναι ένα πρόβλημα με δύο διαστάσεις: η επίλυση των γραμμών συμβάλλει στην επίλυση των στηλών και το αντίστροφο, όπως ένας αποκωδικοποιητής βοηθάει τον άλλο στο σύστημα αποκωδικοποίησης turbo.

Ανατρέχοντας πίσω κάποια χρόνια, ο Βεργου περιπλανιόταν στους διαδρόμους του συνεδριακού κέντρου στη Γενεύη, κρυφοκοιτάζοντας πάνω από τους ώμους των άλλων συμμετεχόντων και βλέποντας πολλούς από αυτούς να προσπαθούν να κατανοήσουν την εργασία του. Κατά την παρουσίαση, οι νέοι διδακτορικοί φοιτητές και ένα μέρος των βετεράνων κωδικοποίησης συνοστίζονται στο αμφιθέατρο, με ανθρώπους που στέκονται δίπλα στην πόρτα. Όταν οι Βεργου και Glavieux ολοκληρώνουν, περιβάλλονται από πλήθος κόσμου, οι οποίοι ζητούν περισσότερες επεξηγήσεις ή απλά επιθυμούν να σφίξουν το χέρι τους.

Ακόμα, για να πεισθούν όσοι είχαν αμφιβολίες ότι το έργο δεν είχε γιγαντιαία περιθώρια λάθους πήρε αρκετό χρόνο. «Επειδή το θεμέλιο των ψηφιακών επικοινωνιών έχει βασιστεί σε ισχυρές μαθηματικές εκτιμήσεις», είπε ο Βεργου αργότερα, «οι κώδικες διόρθωσης σφαλμάτων υπήρχε η πεποίθηση ότι ανήκουν αποκλειστικά στον κόσμο των μαθηματικών».

Αυτό που οδήγησε τον Βεργου και τον Glavieux στη σημαντική εξέλιξη τους δεν ήταν κάποιο απόκρυφα θεώρημα, αλλά ο αγώνας για την επίλυση πραγματικών προβλημάτων στον τομέα των τηλεπικοινωνιών. Στα τέλη της δεκαετίας του 1980,

όταν άρχισαν να εργάζονται σε συστήματα κωδικοποίησης, ήταν έκπληξη για αυτούς το γεγονός ότι μια σημαντική έννοια στην ηλεκτρονική- η ανατροφοδότηση (feedback) - δεν είχε χρησιμοποιηθεί στους ψηφιακούς δέκτες.

Στους ενισχυτές, ένα δείγμα του σήματος εξόδου συνήθως ανατροφοδοτείται στην είσοδο για να εξασφαλιστεί η σταθερή απόδοση. Οι Berrou και Glavieux αναρωτήθηκαν, γιατί δεν πρέπει να χρησιμοποιηθεί και στην κωδικοποίηση;

Έτρεξαν τα πρώτα πειράματα με το νέο σύστημα κωδικοποίησης τους το 1991, χρησιμοποιώντας προσομοιώσεις σε υπολογιστή, και όταν τα αποτελέσματα βγήκαν, έμειναν άφωνοι. «Κάθε μέρα ρωτούσα τον εαυτό μου για τα πιθανά σφάλματα στο πρόγραμμα», λέει ο Berrou.

Το πρώτο πράγμα που έκαναν οι δυο τους μετά την επιβεβαίωση ότι τα αποτελέσματά τους ήταν σωστά ήταν να πατεντάρουν την εφεύρεση στη Γαλλία, την Ευρώπη και τις Ηνωμένες Πολιτείες. Εκείνη την εποχή, η France Télécom ήταν ο κύριος χορηγός του έργου τους, έτσι η γαλλική εταιρεία πήρε την κατοχή των διπλωμάτων ευρεσιτεχνίας των turbo κώδικων. Οι εφευρέτες και ιδρυμά τους, ωστόσο, μοιράζονται μέρος των κερδών αδειοδότησης. (οι Κωδικοί Turbo δεν είχαν κατοχυρωθεί με δίπλωμα ευρεσιτεχνίας στην Ασία, όπου μπορούν, συνεπώς, να χρησιμοποιηθούν δωρεάν.)

Ήταν η France Télécom που ζήτησε από τον Berrou να καταλήξει σε ένα εμπορικό όνομα για την εφεύρεση. Βρήκε το όνομα όταν μια μέρα, βλέποντας ένα αγώνα αυτοκινήτων στην τηλεόραση, παρατήρησε ότι ο νεοσύστατος κώδικας χρησιμοποιούσε τα αποτελέσματα των αποκωδικοποιητών για να βελτιώσει τη διαδικασία αποκωδικοποίησης, όπως και ένας υπερσυμπιεστής χρησιμοποιεί την εξάτμιση του για να συμπιέσει τον αέρα στον κινητήρα για την ενίσχυση της κάυσης. Και εγένετο οι «κώδικες Turbo»!

Οι κώδικες Turbo είναι ήδη σε χρήση στην Ιαπωνία, όπου έχουν ενσωματωθεί στα πρότυπα για τα τρίτης (και τέταρτης, πλέον) γενιάς συστήματα κινητής τηλεφωνίας, επίσημα γνωστό ως το Παγκόσμιο Σύστημα Κινητών Τηλεπικοινωνιών (UMTS). Έχουν χρησιμοποιηθεί για εικόνες, βίντεο και μεταδόσεις ηλεκτρονικών ταχυδρομείων, ανέφερε ο Hirohito Suda, διευθυντής του Εργαστηρίου Επεξεργασίας Σήματος Ραδιοφώνου στο NTT DoCoMo, στην Yokosuka, Ιαπωνία. Για την

μετάδοση φωνής, ωστόσο, οι συνελκτικοί απλοί κώδικες χρησιμοποιούνται ακόμα, λόγω του ότι οι καθυστερήσεις στην αποκωδικοποίηση τους είναι μικρότερες από εκείνες των κωδικών turbo.

Στην πραγματικότητα, η καθυστέρηση αποκωδικοποίησης, ο χρόνος που χρειάζεται για να γίνει αποκωδικοποίηση των δεδομένων, είναι ένα σημαντικό μειονέκτημα για τους κώδικες Turbo. Οι πολλές επαναλήψεις που απαιτούνται από την αποκωδικοποίηση turbo καθιστούν την καθυστέρηση μη αποδεκτή για επικοινωνίες φωνής σε πραγματικό χρόνο και άλλες εφαρμογές που απαιτούν άμεση επεξεργασία των δεδομένων, όπως η αποθήκευση στο σκληρό δίσκο και οπτική μετάδοση.

Για τα συστήματα που μπορούν να ανεχτούν καθυστερήσεις αποκωδικοποίησης, όπως οι δορυφορικές επικοινωνίες, οι κώδικες Turbo έχουν γίνει μια ελκυστική επιλογή. Στην πραγματικότητα, ήδη από τον Σεπτέμβριο του 2003, η Ευρωπαϊκή Υπηρεσία Διαστήματος, με έδρα το Παρίσι, Γαλλία έβαλε σε λειτουργία το SMART-1, η πρώτη αποστολή (πύραυλος) που πήγε στο διάστημα με τη μετάδοση των δεδομένων να γίνεται από κώδικες turbo. Από τότε έχουν χρησιμοποιηθεί κατά κόρον σε αποστολές εξερεύνησης πλανητών και παρατήρησης κομητηών, συλλογής φωτογραφιών και επέκτασεις δικτύου τηλεπικοινωνιών ανά τον κόσμο. Χρησιμοποιήθηκαν από όλες τις υπηρεσίες του τομέα και φυσικά από την NASA.

Πέρα από τη διόρθωση των σφαλμάτων, οι κωδικοί turbo βοήθησαν τις κινητές συσκευές επικοινωνίας να επιτύχουν καλύτερη λήψη. Ψηφιακές εκπομπές ήχου, που μας δίνουν τις επιλογές των ραδιοφωνικών προγραμμάτων ποιότητας DVD, CD κ.τ.λ. και δορυφορικών συνδέσεων, όπως το Παγκόσμιο Δίκτυο της Inmarsat Ltd, στο Λονδίνο, έχουν ενσωματώσει κώδικες turbo στα συστήματά τους από την προηγούμενη δεκαετία.

Και πέρα από τη διόρθωση των σφαλμάτων, οι κωδικοί turbo ή η λεγόμενη αρχή turbo βοηθούν επίσης να λυθούν μια σειρά από προβλήματα επικοινωνίας. "Η ιδέα turbo κωδικοποίησης πυροδότησε πολλές άλλες ιδέες», λέει ο Lajos Hanzo, καθηγητής στη Σχολή Ηλεκτρονικών και Επιστήμης Υπολογιστών στο Πανεπιστήμιο του Σαουθάμπτον, Ηνωμένο Βασίλειο, και επιστημονικός συνεργάτης του IEEE. Ένα παράδειγμα παρουσιάζεται στην προσπάθεια μετριάσμου των επιπτώσεων των πολλαπλών διαδρομών διάδοσης, που είναι, η παραμόρφωση του σήματος που εμφανίζεται όταν ληφθούν πολλαπλά αντίγραφα ενός σήματος που αναπήδησε από

διαφορετικές επιφάνειες. Οι κώδικες Turbo μπορούν τελικά να βοηθήσουν τις φορητές συσκευές να λύσουν αυτό το σημαντικό περιορισμό της κινητής τηλεφωνίας.

Τέλος, μια άλλη σημαντική επίδραση των κωδίκων Turbo είναι πως έκαναν τους ερευνητές να συνειδητοποιήσουν ότι υπήρχαν άλλοι κωδικοί που πλησίαζαν το όριο χωρητικότητας-ικανότητας. Στην πραγματικότητα, μια εναλλακτική λύση που της έχει δοθεί νέα ώθηση στις τηλεπικοινωνίες είναι οι κώδικες χαμηλής πυκνότητας ελέγχου ισοτιμίας (Low Density Parity Check-LDPC), οι οποίοι εφευρέθηκαν στις αρχές της δεκαετίας του 1960 από τον Robert Gallager στο MIT, αλλά σε μεγάλο βαθμό είχαν ξεχαστεί από τότε. «Στη δεκαετία του 1960 και του 1970, υπήρξε ένας πολύ καλός λόγος για τον οποίο κανείς δεν έδωσε προσοχή στους κώδικες LDPC» λέει ο Forney του MIT. "Ήταν σαφώς πολύ περίπλοκοι για την τεχνολογία της εποχής."

Όπως κώδικες turbo, οι LDPC πετύχαιναν την χωρητικότητα μέσω μιας επαναληπτικής διαδικασίας αποκωδικοποίησης, αλλά αυτοί οι κώδικες είναι σημαντικά διαφορετικοί από τους κωδικούς turbo. Τώρα οι ερευνητές έχουν εφαρμόσει τους LDPC κώδικες, ώστε στην πραγματικότητα να ξεπεράσουν τους κώδικες turbo και να φτάσουν ακόμη πιο κοντά στο όριο του Shannon. Πράγματι, θα μπορούσαν να αναδειχθούν σε ένα σοβαρό ανταγωνιστή για τους turbo κώδικες, ειδικά για ασύρματα πρότυπα δίκτυο επόμενης γενιάς, όπως το IEEE 802.11 και το IEEE 802.16. «Οι κώδικες LDPC χρησιμοποιούν πολλές από τις ίδιες γενικές ιδέες [κώδικες turbo], » λέει ο McEliece. «Αλλά με συγκεκριμένους τρόπους, είναι ακόμα πιο εύκολοι να αναλυθούν και ευκολότεροι στην εφαρμογή». Ένα άλλο πλεονέκτημα, ίσως το μεγαλύτερο απ' όλα, είναι ότι οι πατέντες LDPC έχουν λήξει, έτσι ώστε οι εταιρείες μπορούν να τους χρησιμοποιήσουν χωρίς να χρειάζεται να πληρώνουν για τα δικαιώματα πνευματικής ιδιοκτησίας.

Οι κώδικες Turbo έθεσαν τέλος σε μια αναζήτηση που κράτησε για περισσότερο από 40 χρόνια. «Είναι αξιοσημείωτο, επειδή υπάρχει αυτή η επανάσταση, αν σήμερα πλέον δεν μπορείτε να πλησιάσετε το όριο του Shannon, τι τρέχει με εσάς; » λέει ο Tanner από το Πανεπιστήμιο του Ιλινόις. «Ο καθένας μπορεί να πλησιάσει στην όριο χωρητικότητας του Shannon, αλλά ας αναρωτηθούμε για το πόσο γρηγορότερα πηγαίνει ο κώδικας σας ... και αν είστε 0,1 dB από τον Shannon ή 0.001 dB. »

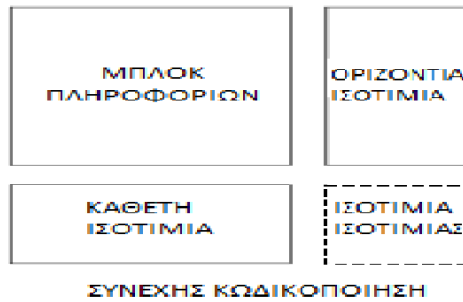
Ήταν η διορατικότητα και η αφέλεια των «αουτσάιντερ» που βοήθησαν τους Berrou και Glavieux να συνειδητοποιήσουν τι αγνοούσε η επιστημονική κοινότητα της θεωρίας αποκωδικοποίησης. « Οι ξώδικες Turbo είναι το αποτέλεσμα μιας εμπειρικής, επίπονης κατασκευής ενός παγκόσμιου συστήματος κωδικοποίησης / αποκωδικοποίησης, χρησιμοποιώντας τα υπάρχοντα τούβλα-κομμάτια τα οποία δεν είχαν ποτέ συναρμολογηθεί με αυτόν τον τρόπο πριν» , έγραψαν πριν από λίγα χρόνια.

Ο Berrou δήλωσε ότι το έργο τους είναι απόδειξη ότι δεν είναι πάντοτε απαραίτητο να ξέρετε για τα θεωρητικά όρια ώστε να είστε σε θέση να τα φτάσετε.

Καταλαβαίνουμε λοιπόν, πως η θεωρία των κωδικών διόρθωσης σφάλματος έχει παρουσιάσει ένα μεγάλο αριθμό κατασκευών κώδικα με αντίστοιχους αλγόριθμους αποκωδικοποίησης. Ωστόσο, για εφαρμογές όπου απαιτείται πολύ ισχυρή διόρθωση σφάλματος, τα χαρακτηριστικά που χρειάζεται να διαθέτουν αυτές οι κατασκευές έχουν ως αποτέλεσμα πάρα πολύ σύνθετες λύσεις αποκωδικοποιητή. Ο τρόπος για την καταπολέμηση αυτό είναι να χρησιμοποιηθεί συνεχής κωδικοποίηση, όπου δύο (ή και περισσότεροι) κωδικοί χρησιμοποιούνται ο ένας μετά από τον άλλο ή παράλληλα - συνήθως με κάποιο είδος διεμπλοκής.

Οι κώδικες που περιέχονται (συστατικοί κώδικες) αποκωδικοποιούνται με τους αντίστοιχους αποκωδικοποιητές, αλλά το τελικό αποκωδικοποιημένο αποτέλεσμα είναι συνήθως υποδεέστερο του αναμενομένου-βέλτιστου. Αυτό σημαίνει ότι καλύτερα αποτελέσματα θα μπορούσαν να επιτευχθούν με ένα πιο περίπλοκο αλγόριθμο αποκωδικοποίησης - όπως η εξαναγκασμένη προσπάθεια δοκιμής όλων των πιθανών κωδικών λέξεων. Ωστόσο, οι συστατικοί κώδικες προσφέρουν την δυνατότητα εναλλαγής μεταξύ διόρθωσης λαθών και την πολυπλοκότητα του αποκωδικοποιητή.

Παράδειγμα συνεχούς κωδικοποίησης απεικονίζεται παρακάτω. Εδώ βλέπουμε το πλαίσιο πληροφοριών να απεικονίζεται ως ένα τετράγωνο – εάν υποθέσουμε μπλοκ διεμπλοκής- και βλέπουμε την ισοτιμία από την κάθετη κωδικοποίηση και την ισοτιμία από την οριζόντια κωδικοποίηση. Για σειριακή αλληλουχία τα bits ισοτιμίας από έναν από τους συστατικούς κώδικες είναι κωδικοποιημένα με δεύτερο κώδικα και έχουμε ισοτιμία ισοτιμίας.

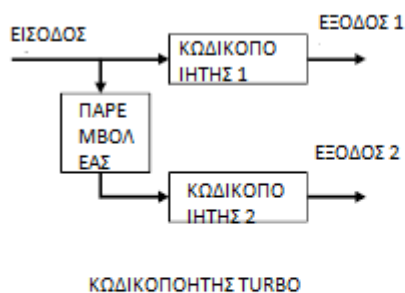


Αν οι κώδικες λειτουργούσαν παράλληλα, δεν θα είχαμε αυτή την πρόσθετη ισοτιμία.

Η ιδέα της συνεχούς κωδικοποίησης ταιριάζει καλά με την κωδικοποίηση καναλιού της θεωρίας του Shannon, δηλώνοντας ότι όσο μένουμε στη σωστή πλευρά της χωρητικότητας του καναλιού μπορούμε να διορθώσουμε τα πάντα, εάν φυσικά ο κώδικας είναι αρκετά μεγάλος. Αυτό σημαίνει επίσης ότι εάν ο κώδικας είναι πολύ μεγάλος, δεν χρειάζεται να είναι ο βέλτιστος. Το μήκος από μόνο του δίνει καλές δυνατότητες διόρθωσης λαθών, και η συνεχής κωδικοποίηση είναι απλά ένας τρόπος κατασκευής - και ιδιαίτερα αποκωδικοποίησης - κώδικων μεγάλου μήκους.

1.1 Κωδικοποίηση

Η βασική ιδέα των κωδικών Turbo είναι να χρησιμοποιήσετε δύο συνελκτικούς κώδικες παράλληλα με κάποιο είδος διεμπλοκής μεταξύ τους. Συνελκτικοί κώδικες μπορούν να χρησιμοποιηθούν για να κωδικοποιήσουμε μια συνεχής ροή δεδομένων, υποθέτοντας σε αυτή την περίπτωση ότι τα δεδομένα είναι διαμορφωμένα σε πεπερασμένα μπλοκ – αντιστοιχώντας το μέγεθος του παρεμβολέα. Τα πλαίσια μπορούν να καταργηθούν - δηλαδή οι κωδικοποιητές εξαναγκάζονται σε μια γνωστή κατάσταση μετά το μπλοκ πληροφοριών. Η ουρά τερματισμού προστίθεται μετέπειτα στην κωδικοποιημένη πληροφορία και χρησιμοποιείται στον αποκωδικοποιητή. Το σύστημα απεικονίζεται παρακάτω.



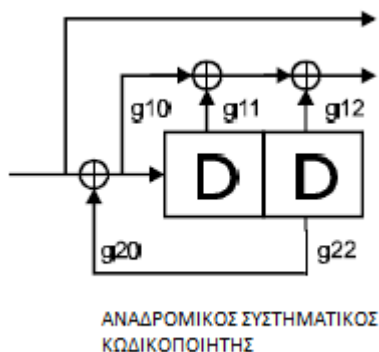
Μπορούμε να θεωρήσουμε τον κώδικα turbo και ένα μεγάλο μπλοκ κώδικα. Η απόδοση εξαρτάται από την κατανομή βάρους - όχι μόνο την ελάχιστη απόσταση, αλλά τον αριθμό των λέξεων με χαμηλό βάρος. Ως εκ τούτου, θέλουμε πρότυπα εισόδου που να δίνουν λέξεις χαμηλού βάρους από τον πρώτο κωδικοποιητή για να διαπλέκονται με τα πρότυπα δίνοντας λέξεις με μεγάλο βάρος για το δεύτερο κωδικοποιητή.

Συνελκτικοί κώδικες έχουν αρκετά συχνά κωδικοποιηθεί σε μορφή «τροφοδοσίας προς τα εμπρός», όπως $(G1, G2) = (1 + D2, 1 + D + D2)$. Ωστόσο, για αυτούς τους κωδικούς ένα ενιαίο 1, δηλαδή την ακολουθία ... 0001000 ..., θα δώσει μια κωδική λέξη η οποία είναι ακριβώς τα πρωταρχικά διανύσματα και το βάρος αυτής της κωδικής λέξης θα είναι σε γενικές γραμμές πολύ χαμηλό. Είναι σαφές ότι ένα απλό 1

θα μεταδίδεται μέσα σε οποιοδήποτε παρεμβολέα ως ένα μόνο 1, οπότε το συμπέρασμα είναι ότι αν χρησιμοποιήσουμε τους κώδικες με μορφή «τροφοδοσίας προς τα εμπρός» στο σύστημα turbo ο κώδικας που θα προκύψει θα έχει ένα μεγάλο αριθμό κωδικών λέξεων με πολύ χαμηλό βάρος.

Το κόλπο είναι να χρησιμοποιήσετε τους κωδικούς στο αναδρομική συστηματική μορφή τους, όπου διαιρούμε με ένα από τα πρωταρχικούς φορείς. Το παράδειγμά μας δίνει $(1, G_2 / G_1) = (1, (1 + D + D^2) / (1 + D^2))$. Αυτή η λειτουργία δεν αλλάζει το σύνολο των κωδικοποιημένων ακολουθιών, αλλά η χαρτογράφηση των ακολουθιών εισόδου από τις ακολουθίες εξόδου είναι διαφορετική. Λέμε ότι ο κώδικας είναι ο ίδιος, πράγμα που σημαίνει ότι τα στοιχεία της απόστασης παραμένουν αμετάβλητα, αλλά η κωδικοποίηση είναι διαφορετική.

Στο σχήμα που ακολουθεί απεικονίζεται ένας κωδικοποιητής οσον αφορά την αναδρομική συστηματική μορφή. Η ακολουθία εξόδου που πήραμε από τον κωδικοποιητή «τροφοδοσίας προς τα εμπρός» με ένα και μόνο 1 λαμβάνεται τώρα με την είσοδο $1 + D^2 = G_1$. Πιο σημαντικό είναι το γεγονός ότι ένα και μόνο 1 δίνει μια κωδική λέξη ημι-άπειρου βάρους, έτσι με τους αναδρομικούς συστηματικούς κωδικοποιητές μπορεί να έχουμε μια ευκαιρία να βρούμε έναν παρεμβολέα στον οποίο τα μοτίβα πληροφοριών που δίνουν λέξεις χαμηλού βάρους από το πρώτο κωδικοποιητή να παρεμβάλλονται με μοτίβα από το δεύτερο κωδικοποιητή δίνοντας λέξεις με υψηλό βάρος. Τα περισσότερα κρίσιμα πρότυπα εισόδου είναι τώρα τα πρότυπα του βάρους 2. Για τον κώδικα του παραδείγματος, η ακολουθία της πληροφορίας ... 01010 ... θα δώσει μια έξοδο με βάρος 5.



Σημειώνουμε το γεγονός ότι οι κώδικες είναι συστηματικοί, είναι απλά μια σύμπτωση, αν αποδεικνύεται ότι είναι πολύ βολικό για αρκετούς λόγους. Ένας από

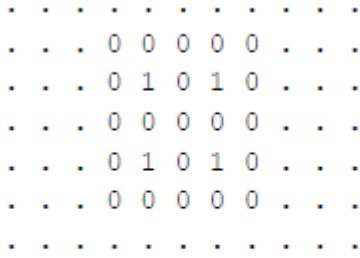
αυτους είναι ότι το ποσοστό λάθους δυαδικών ψηφίων (bit error rate - BER), μετά την αποκωδικοποίηση ενός συστηματικού κώδικα δεν μπορεί να υπερβαίνει το BER στο κανάλι.

Φανταστείτε ότι τα ληφθέντα σύμβολα ισοτιμίας ήταν εντελώς τυχαία, τότε ο αποκωδικοποιητής θα μπορούσε φυσικά να κολλήσει στην εισερχόμενη εκδοχή των πληροφοριών. Αν τα σύμβολα ισοτιμίας τουλάχιστον έχουν κάποιο νόημα θα αποκτήσουμε πληροφορίες σχετικά με τον μέσο όρο και το BER μετά την αποκωδικοποίηση θα είναι κάτω από το BER στο κανάλι.

Ένα πράγμα που είναι σημαντικό όσον αφορά τη συστηματική διαδικασία είναι πως αν μεταδώσουμε το συστηματικό μέρος και από τους δύο κωδικοποιητές, αυτό θα είναι απλώς μια επανάληψη, και γνωρίζουμε ότι μπορούμε να κατασκευάσουμε καλύτερους κωδικούς από τους απλούς κώδικες επανάληψης. Το τμήμα πληροφοριών θα πρέπει να διαβιβάζεται μόνο από έναν από τους συστατικούς κώδικες, έτσι ώστε αν χρησιμοποιήσουμε συστατικούς κώδικες με ρυθμό $1/2$, το τελικό ποσοστό του κώδικα turbo γίνεται $1/3$. Εάν απαιτείται μεγαλύτερη επάρκεια, θα πρέπει να επιλέξουμε τους συστατικούς κώδικες με τα χαμηλότερα ποσοστά. Επίσης μπορούμε να χρησιμοποιήσουμε μεθόδους διάτρησης μετά τους συστατικούς κωδικοποιητές για να αυξηθεί το ποσοστό των κωδικών turbo.

Τώρα έρχεται το ερώτημα της διεμπλοκής. Η πρώτη επιλογή θα ήταν ένα απλό μπλοκ παρεμβολέα, δηλαδή να γράφει από τη γραμμή και να διαβάζει από τη στήλη. Ωστόσο, δύο λέξεις εισόδου της χαμηλού βάρους θα δώσουν κάποια πολύ ατυχές μοτίβα σε αυτό το παρεμβολέα. Το μοτίβο φαίνεται στο σχήμα που θα ακολουθήσει στο παράδειγμα του κώδικα που έχουμε ήδη δώσει. Βλέπουμε ότι αυτό είναι ακριβώς δύο φορές η κρίσιμη λέξη δύο εισόδων για τον οριζόντιο κωδικοποιητή και δύο φορές το κρίσιμο πρότυπο δύο εισόδων για τον κατακόρυφο

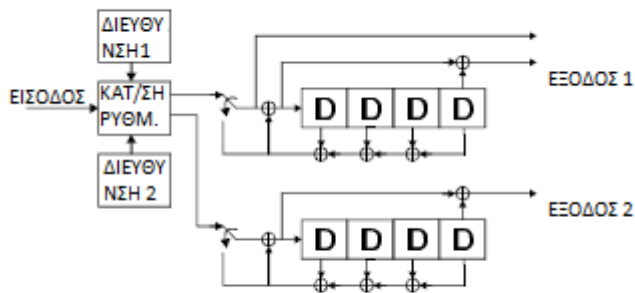
κωδικοποιητή επίσης. Το αποτέλεσμα είναι μια λέξη κώδικα χαμηλού βάρους (16 για τον κώδικα του παραδείγματος μας), όχι η χαμηλότερη δυνατή, αλλά δεδομένου ότι το μοτίβο εμφανίζεται σε κάθε θέση στον παρεμβολέα εμείς θα έχουμε ένα μεγάλο αριθμό από αυτές τις λέξεις.



ΚΡΙΣΙΜΟ ΜΟΤΙΒΟ ΣΕ ΜΠΛΟΚ
ΠΑΡΕΜΒΟΛΕΑ

Αυτή τη φορά το τέχνασμα είναι να χρησιμοποιήσετε ένα ψευδο-τυχαίο παρεμβολέα, δηλαδή να διαβάσει τα bits πληροφοριών στο δεύτερο κωδικοποιητή σε μια τυχαία (αλλά σταθερή) σειρά. Το μοτίβο από το παραπάνω σχήμα μπορεί να εξακολουθήσει να εμφανίζεται, αλλά όχι τόσο συχνά. Από την άλλη πλευρά, έχουμε τώρα τη πιθανότητα ότι ένα κρίσιμο μοτίβο δύο εισόδων να είναι παρεμβαλλόμενο σε ένα άλλο κρίσιμο μοτίβο δύο εισόδων. Η πιθανότητα ότι ένα συγκεκριμένο μοτίβο δύο εισόδων παρεμβάλλεται σε ένα άλλο (ή το ίδιο) ειδικό πρότυπο δύο εισόδων είναι $2 / N$, όπου N είναι το μέγεθος του διεμπλοκέα. Δεδομένου ότι το πρώτο μοτίβο θα μπορούσε να εμφανιστεί σε οποιαδήποτε από τις θέσεις N στο μπλοκ, πρέπει να περιμένουμε αυτό το ατυχές ταίριασμα να εμφανίζεται δύο φορές σε ένα ψευδο-τυχαίο παρεμβολέα οποιουδήποτε μήκους. Ακόμα ο ψευδοτυχαίος παρεμβολέας είναι ανώτερος από τον παρεμβολέα μπλοκ, και η ψευδοτυχαία διαπλοκή είναι πρότυπο για τους κώδικες Turbo.

Θα ολοκληρώσουμε αυτό το τμήμα, δείχνοντας ένα πιο λεπτομερές σχέδιο μιας κωδικοποίησης turbo.



ΠΑΡΑΔΕΙΓΜΑ ΚΩΔΙΚΟΠΟΗΤΗ TURBO

Βλέπουμε λοιπόν τους δύο αναδρομικούς συστηματικούς κωδικοποιητές, αυτή τη φορά για τον κώδικα $(1, (1 + D^4) / (1 + D + D^2 + D^3 + D^4))$. Παρατηρούμε ότι το συστηματικό κομμάτι (bit) αφαιρείται από έναν από αυτούς. Στην είσοδο των συστατικών κωδικοποιητών βλέπουμε ένα διακόπτη. Αυτό χρησιμοποιείται για να αναγκάσει τους κωδικοποιητές να εισέλθουν σε πλήρη μηδενική κατάσταση, δηλαδή να τερματίσει η όλη δικτυακή διαδικασία. Το πλήρες εισερχόμενο πλαίσιο διατηρείται σε μια ρυθμιστική κατάσταση από όπου διαβάζεται με δύο διαφορετικά σύνολα διευθύνσεων, ένα για την αρχική αλληλουχία και ένα για την αλληλουχία εξ παρεμβολής. Με αυτό το τρόπο η έξοδος 1 και η έξοδος 2 αντιστοιχούν με το ίδιο πλαίσιο και μπορούν να συγχωνευτούν πριν από τη μετάδοση.

1.2 Πρώτη Αποκωδικοποίηση

Η αποκωδικοποίηση των κώδικων διόρθωσης σφάλματος είναι βασικά μια σύγκριση των πιθανοτήτων για τις διαφορετικές κωδικές λέξεις - ή με τους συνελκτικούς κώδικες, διαφορετικές διαδρομές στα δικτυακή διαδικασία. Όταν μιλάμε για πιθανότητες, είναι πάντα η πιθανότητα κάποιου γεγονότος έχοντας μια ορισμένη ποσότητα πληροφοριών σχετικά με αυτό το γεγονός. Αυτό είναι ιδιαίτερα σαφές όταν μιλάμε για πιθανότητες όσον αφορά κάτι που έχει ήδη συμβεί, η οποία είναι πάντα η περίπτωση στην θεωρία κωδικοποίησης. Τι εννοούμε όταν μιλάμε για την πιθανότητα ότι x στάλθηκε, $p(x)$ είναι η πιθανότητα ότι ο X στάλθηκε έχοντας μια δεδομένη ποσότητα πληροφοριών που έχουμε σχετικά με το συμβάν. Συνήθως αυτή είναι μόνο η λαμβανόμενη θορυβώδης εκδοχή του X - και φυσικά τη γνώση του καθεστώτος κωδικοποίησης, του σύνδεσμου μετάδοσης κ.λπ.

Σε ορισμένες περιπτώσεις έχουμε κάποια γνώση του εκπεμπόμενου σήματος πριν αποκωδικοποιήσουμε αυτό που λαμβάνουμε. Αυτό μπορεί να είναι πληροφορίες ότι ορισμένα μηνύματα είναι πιο πιθανό να συμβούν από άλλα ή πληροφορίες που μεταδίδονται από άλλες αλληλουχίες. Καλούμε αυτές τις πληροφορίες, πληροφορίες προτεραιότητας και έχουν τις αντίστοιχες προτεραιότητες όσον αφορά τις πιθανότητες τους. Παρόμοια, μπορούμε να μιλήσουμε για τις πιθανότητες «εκ των υστέρων» όταν έχουμε ήδη συμπεριλάβει τόσο τις πιθανότητες πληροφοριών με προτεραιότητα, αλλά και τις πληροφορίες που έχουν αποκτηθεί από την αποκωδικοποίηση.

Για τους κωδικούς των turbo έχουμε δύο κωδικοποιημένες ακολουθίες. Είναι σαφές ότι πρέπει να ξεκινήσουμε από την αποκωδικοποίηση ενός από τις δύο για να πάρουμε μια πρώτη εκτίμηση της ακολουθίας πληροφοριών. Θα πρέπει στη συνέχεια να χρησιμοποιηθεί αυτή η εκτίμηση ως μια εκ των πληροφοριών με προτεραιότητες στην αποκωδικοποίηση της δεύτερης κωδικοποιημένης αλληλουχίας. Αυτό προϋποθέτει ότι ο αποκωδικοποιητής είναι σε θέση να χρησιμοποιήσει μία ήπιας μορφής απόφαση-σύστημα εισόδου και να παράγει κάποιας ήπιας μορφής έξοδο.

Η αποκωδικοποίηση παρουσιάζεται στο παρακάτω σχήμα.



Ο αποκωδικοποιητής πρότυπο για τους κώδικες turbo είναι ο αποκωδικοποιητής πιθανοτήτων «εκ των υστέρων»- A Posteriori Probability decoder (APP) ο οποίος μερικές φορές αναφέρεται ως μέγιστος «εκ των Υστέρων» αλγόριθμος αποκωδικοποίησης- Maximum A Posteriori decoding algorithm (MAP). Ο αποκωδικοποιητής APP πράγματι υπολογίζει τις πιθανότητες των υστερων για κάθε bits πληροφορίας.

Εμείς θα αναπαραστήσουμε τη ήπια μορφή εισόδου / εξόδου, όπως είναι οι αναλογίες πιθανοφάνειας, δηλαδή λ.χ έναν καταγεγραμμένο αριθμό όπου οι αρνητικοί αριθμοί δείχνουν ότι το μηδέν είναι η πιο πιθανή τιμή του δυαδικού ψηφίου. Όπως φαίνεται από τον παρακάτω τύπο ο λόγος πιθανοφάνειας στις πιθανότητες «εκ των υστέρων» μπορεί εύκολα να διαχωριστεί σε δύο κομμάτια - η αναλογία πιθανοφάνειας των προτέρων πιθανοτήτων του δυαδικού ψηφίου d και οι t πληροφορίες που αποκτήθηκαν από την τρέχουσα εκείνη τη στιγμή παρατήρηση. Αυτό σημαίνει ότι όταν έχουμε αποκτήσει πρόσθετες πληροφορίες σχετικά με τα bits πληροφοριών, όπως και με το δεύτερη αποκωδικοποίηση, απλά προσθέτουμε ένα (αρνητικό ή θετικό) στοιχείο στον λόγο πιθανοφάνειας.

$$\begin{aligned} \Lambda(d_t) &= \log \frac{\Pr\{d_t=1, \text{ παρατήρηση}\}}{\Pr\{d_t=0, \text{ παρατήρηση}\}} \\ &= \log \frac{\Pr_{ap}\{d_t=1\}}{\Pr_{ap}\{d_t=0\}} + \log \frac{\Pr\{\text{ παρατήρηση} | d_t=1\}}{\Pr\{\text{ παρατήρηση} | d_t=0\}} \\ &= \log \frac{\Pr_{ap}\{d_t=1\}}{\Pr_{ap}\{d_t=0\}} + \Lambda'(d_t) \end{aligned}$$

1.3 Τι σημαίνει turbo στους κώδικες Turbo.

Όταν έχουμε μια εξίσωση ισοτιμίας, περιλαμβάνει μια σειρά από bits πληροφοριών. Ας δούμε μία από τις απλούστερες εξισώσεις ισοτιμίας, ένα άθροισμα δύο bits πληροφοριών: $P = I_1 + I_2$. Είναι σαφές ότι εάν και το P και το I_2 είναι αξιόπιστα έχουμε μια αξιόπιστη εκτίμηση του I_1 , από την άλλη πλευρά αν το I_2 είναι πολύ αναξιόπιστο δεν έχουμε πολλές πληροφορίες σχετικά με το I_1 . Αν τώρα υποθέσουμε ότι τόσο το I_1 όσο και το I_2 είναι αναξιόπιστα όταν αποκωδικοποιήσαμε την πρώτη ακολουθία, αλλά ότι το I_2 εμπλέκεται σε ορισμένες εξισώσεις ισοτιμίας με πολύ αξιόπιστα bits στην δεύτερη ακολουθία κωδικοποίησης, τότε θα μπορούσαμε να επιστρέψουμε στις εξισώσεις ισοτιμίας από την πρώτη αλληλουχία για μία δεύτερη επανάληψη με αυτή τη νέα και πολύ πιο αξιόπιστη εκτίμηση του I_2 . Με αυτό τον τρόπο θα μπορούσαμε να συνεχίσουμε να από-κωδικοποιήσουμε τις δύο κωδικοποιημένες ακολουθίες και κινηθούμε προς την τελική απόφαση.

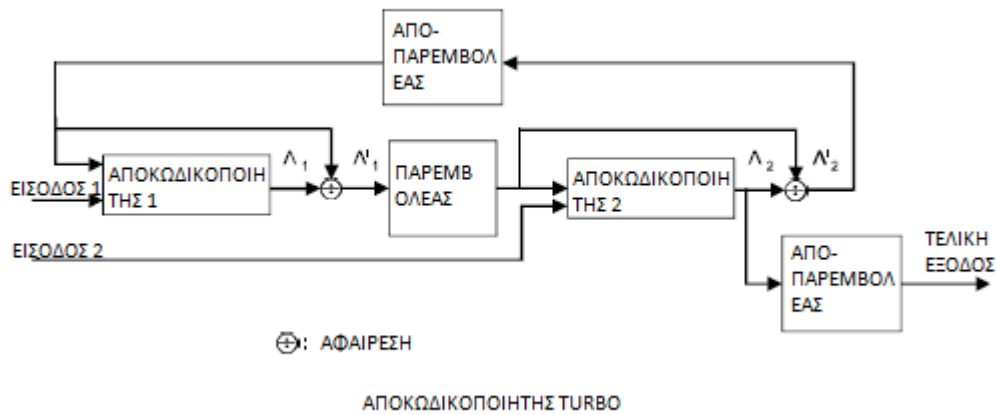
Ωστόσο, αυτό δεν είναι τόσο εύκολο, δεδομένου ότι πρέπει να είμαστε πολύ προσεκτικοί να μην χρησιμοποιήσουμε τις πληροφορίες μας περισσότερες από μία φορά. Ευτυχώς βλέπουμε από τον τύπο που είχαμε αναφέρει, ότι να είναι εύκολο να αφαιρέσουμε την εκ των προτέρων πληροφορία (η οποία ήρθε από τον άλλο αποκωδικοποιητή), από την έξοδο του αποκωδικοποιητή. Αυτό θα αποτρέψει το μεγαλύτερο μέρος της ανεπιθύμητης ανατροφοδότησης. Μπορούμε ακόμα να έχουμε βρόχους στη διαδικασία λήψης αποφάσεων, όπως δηλαδή θα μπορούσαμε να δούμε ότι οι I_1 επηρεάζει το I_2 στον πρώτο αποκωδικοποιητή, ότι το I_2 επηρεάζει το I_3 στο δεύτερο αποκωδικοποιητή και τέλος, ότι το I_3 επηρεάζει το I_1 στην επόμενη επανάληψη του πρώτου αποκωδικοποιητή. Με αυτό τον τρόπο η νέα βελτιωμένη εκτίμηση του I_1 θα πρέπει να βασίζεται σε πληροφορίες που προέρχονται από το I_1 εξ' αρχής.

Η χρήση του συστήματος στην πράξη έχει δείξει ότι εάν αφαιρέσουμε τον λόγο πιθανοφάνειας του ενός εκ των προτέρων πληροφοριών μετά από κάθε συστατικό αποκωδικοποιητή και κάνουμε μια σειρά από επαναλήψεις αποκωδικοποίησης έχουμε ένα σύστημα που λειτουργεί εντυπωσιακά καλά και το οποίο μάλιστα για πολλές εφαρμογές υπερτερεί στην πραγματικότητα από τα προηγουμένως γνωστά συστήματα. Ακόμα όμως, πρέπει να καταλήξουμε ότι το τελικό αποτέλεσμα μετά την

ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΠΛΗΡΟΦΟΡΙΚΗΣ

αποκωδικοποίηση turbo είναι μια υπο-βέλτιστη αποκωδικοποίηση λόγω των βρόχων στη διαδικασία λήψης αποφάσεων. Για χαμηλές αναλογίες σήματος προς θόρυβο μπορεί ακόμη και να δούμε ότι η αποκωδικοποίηση δεν συγκλίνει σε τίποτα κοντά στην μεταδιδόμενη κωδική λέξη.

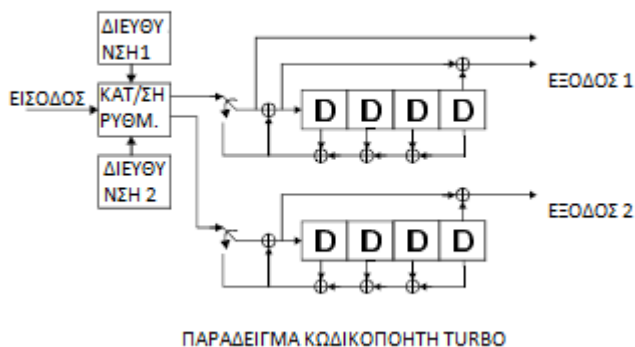
Ο αποκωδικοποιητής turbo απεικονίζεται παρακάτω.



1.4 Παράδειγμα απόδοσης

Θα δείξουμε ένα παράδειγμα των επιδόσεων με τους κώδικες Turbo.

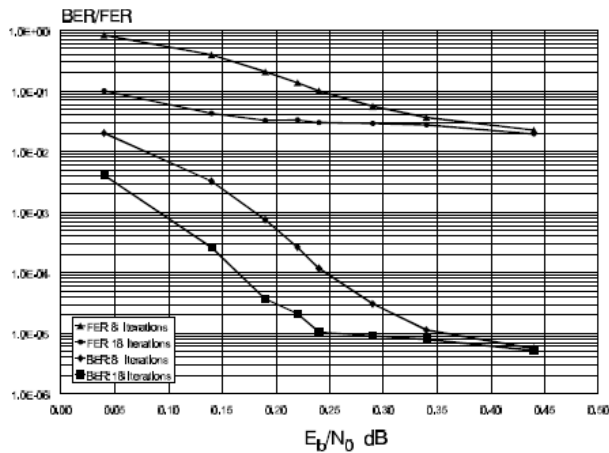
Χρησιμοποιούμε το σύστημα που αναφέραμε ως παράδειγμα και παραπάνω, δηλαδή τον κωδικό $(1, (1 + D^4) / (1 + D + D^2 + D^3 + D^4))$ και για τους δύο κωδικοποιητές αλλά η ακολουθία πληροφοριών μεταδίδεται μόνο από τον πρώτο.



Αυτό σημαίνει ότι το συνολικό ποσοστό αναλογίας είναι $1/3$. Το μήκος του μπλοκ είναι 10384 bits και χρησιμοποιούμε ένα ψευδο-τυχαίο παρεμβολέα. Μετά από κάθε σχηματισμό οι κωδικοποιητές αναγκάζονται να μεταβούν σε μηδενική κατάσταση. Η αντίστοιχη ουρά τερματισμού, 4 bits πληροφοριών και 4 δυαδικά ψηφία ισοτιμίας για κάθε κωδικοποιητή, συνολικά 16 bits προσαρτώνται στο μεταδιδόμενο σχηματισμό και χρησιμοποιούνται στον αποκωδικοποιητή. Κατά σειρά ο τερματισμός μειώνει τον ρυθμό, αλλά για μεγάλους σχηματισμούς αυτό δεν έχει καμία πρακτική επίδραση. Στην περίπτωση αυτή, ο συντελεστής μειώνεται από 0,3333 σε 0.3332.

Οι καμπύλες απόδοσης για Bit Error Rate (BER) και Frame Error Rate (FER) φαίνονται σε παρακάτω απεικόνιση. Λόγω της υπο-βέλτιστης αποκωδικοποίησης οι καμπύλες απόδοσης αποτελούνται από δύο μέρη. Σε περίπτωση χαμηλής αναλογίας σηματος/θορύβου το κύριο πρόβλημα είναι η έλλειψη σύγκλισης στην επαναληπτική διαδικασία αποκωδικοποίησης, με αποτέλεσμα τους σχηματισμούς με μεγάλο αριθμό σφαλμάτων μέσα τους. Στην περιοχή αυτή απέχουμε αρκετά από τη βέλτιστη αποκωδικοποίηση. Αυτό σημαίνει ότι μπορούμε να επωφεληθούμε από περισσότερες επαναλήψεις. Όπως βλέπουμε από το σχεδιάγραμμα υπάρχει σημαντικό κέρδος

πηγαίνοντας από 8 στις 18 επαναλήψεις, και με περισσότερες επαναλήψεις η απόδοση μπορεί να είναι ακόμη καλύτερη.



Για σήματα με υψηλή αναλογία σηματος/θορύβου η αποκωδικοποίηση είναι σχεδόν η βέλτιστη, και το κύριο πρόβλημα είναι οι κωδικές λέξεις χαμηλού βάρους. Αυτή η περιοχή αναφέρεται συνήθως ως σφάλμα πατώματος(κατώτατου ορίου), δεδομένου ότι η βελτίωση για την αύξηση της αναλογίας σηματος προς θόρυβο είναι πολύ μικρή. Παρά το όνομα δεν είναι ένα πραγματικό πάτωμα, καθώς τα Bit Error Rate (BER) και Frame Error Rate (FER) μειώνονται συνεχώς, αν και όχι τόσο γρήγορα όσο για τα σήματα με χαμηλή αναλογία σηματος/θορύβου. Παρατηρήστε ότι όταν ο λόγος σηματος προς θόρυβο είναι υψηλός ένας μικρός αριθμός επαναλήψεων είναι επαρκής.

1.5 Αποκωδικοποίηση APP

Ο αλγόριθμος του αποκωδικοποιητή πιθανοτήτων «εκ των υστέρων»- A Posteriori Probability decoder (APP) πράγματι υπολογίζει τις εκ των υστέρων πιθανότητες των μεταδιδόμενων δυαδικών ψηφίων πληροφορίας για ένα συνελκτικό κώδικα. Σε αυτήν την παρουσίαση θα περιοριστούμε στους συνελκτικούς κώδικες με ρυθμό $1/n$. Ένας συνελκτικός κωδικοποιητής με μνήμη M (όπως εκείνος που έχουμε παρουσιάσει στο σχήμα Αναδρομικού Συστηματικού Αποκωδικοποιητή) μπορεί να θεωρηθεί ως μία πηγή Markov όπου το $2M$ δηλώνεται S_t , η είσοδος D_t και η έξοδος

X_t . Η έξοδος X_t και η νέα κατάσταση S_t είναι συναρτήσεις της εισόδου d_t και της προηγούμενης κατάστασης S_{t-1} .

Αν η έξοδος X_t μεταδίδεται μέσω ενός διακριτού καναλιού δίχως μνήμη με λευκό θόρυβο Gaussian, η πιθανότητα λήψης του Y_t όταν το απεσταλμένο είναι X_t είναι

$$\Pr\{Y_t | X_t\} = \prod_{j=1}^n \frac{1}{\sqrt{2\pi}\sigma} \cdot e^{-\frac{(y_{tj} - x_{tj})^2}{2\sigma^2}}$$

όπου x είναι το j -οστό bit της λέξης X_t που μεταδίδεται και Y_{tj} η αντίστοιχη που λαμβάνεται. Ο λόγος σήματος προς θόρυβο είναι $E_s / N_0 = 1/2\sigma^2$. Κατά σειρά, η γνώση της αναλογίας του σήματος προς του θόρυβο είναι απαραίτητη για τον αλγόριθμο APP. Ωστόσο, μπορεί να καθοριστεί σε μια σταθερή τιμή, ανάλογα με το σημείο λειτουργίας του συστήματος, με μόνο μια μικρή υποβάθμιση της απόδοσης.

Ας υποθέσουμε ότι έχουμε λάβει την ακολουθία $Y_1^L = Y_1, Y_2, \dots, Y_L$. Οι εκ των υστέρων πιθανότητες των μεταβάσεων (π.χ των υποσυστημάτων) βρίσκονται έτσι:

$$\Pr\{S_{t-1}=m', S_t=m | Y_1^L\} = \frac{\Pr\{S_{t-1}=m', S_t=m, Y_1^L\}}{\Pr\{Y_1^L\}}, \quad t=1 \dots L$$

Το $\Pr\{Y_1^L\}$ είναι μια σταθερά για μία δεδομένη λαμβανόμενη ακολουθία και αφού υπολογίζουμε τους κώδικες με ρυθμό $1/n$, υπάρχει ένα συγκεκριμένο κομμάτι πληροφοριών που σχετίζονται με κάθε μετάβαση κατάστασης. Εμείς ως εκ τούτου,

$$\sigma_t(i, m') = \Pr\{d_t=i, S_{t-1}=m', Y_1^L\}$$

καθορίζουμε

Ο τελικός λόγος πιθανοφάνειας γίνεται

$$\Lambda(d_t) = \log \frac{\Pr\{d_t=1, \text{observation}\}}{\Pr\{d_t=0, \text{observation}\}} = \frac{\sum_{m'} \sigma_t(1, m')}{\sum_{m'} \sigma_t(0, m')}$$

Για να υπολογιστεί το $\sigma(i, m')$, ορίζουμε τις ακόλουθες λειτουργίες πιθανοτήτων $\alpha_t(m)$, $\beta_t(m)$, και $\gamma_t(i, m')$ ως

$$\alpha_t(m) = \Pr\{S_t = m, \mathbf{Y}_1^t\}$$

$$\beta_t(m) = \Pr\{\mathbf{Y}_{t+1}^L | S_t = m\}$$

$$\gamma_t(i, m') = \Pr\{d_t = i, \mathbf{Y}_t | S_{t-1} = m'\}$$

Σε σύγκριση με τον αλγόριθμο Viterbi το $\alpha_t(m)$ αντιστοιχεί στις μετρήσεις του συστήματος, ενώ το $\gamma_t(i, m')$ αντιστοιχεί στις μετρήσεις του υποσυστήματος. Το $\beta_t(m)$ μπορεί να θεωρηθεί ως υποανάπτυκτη μέτρηση συστήματος. Για το συμβολισμό-σημειογραφία θα χρειαστούμε επίσης μια λειτουργία, δίνοντας στο νέο κωδικοποιητή συστήματος, S_t όταν $S_{t-1} = m'$ και $d_t = I$ άρα

νέο συστημα (i, m') `newstate(i, m')`

Και δίνοντας στον παλιό κωδικοποιητή συστήματος S_{t-1} όταν $S_t = m$ και $d_t = I$ άρα

παλαιό σύστημα (I, m') `oldstate(i, m')`

Δεδομένου ότι ο κωδικοποιητής είναι μια διαδικασία Markov και το κανάλι είναι χωρίς μνήμη, έχουμε

$$\Pr\{\mathbf{Y}_{t+1}^L | S_t = m, \mathbf{Y}_1^t\} = \Pr\{\mathbf{Y}_{t+1}^L | S_t = m\}$$

και

$$\begin{aligned} \sigma_t(i, m') &= \Pr\{S_{t-1} = m', \mathbf{Y}_1^{t-1}\} \cdot \Pr\{d_t = i, \mathbf{Y}_t | S_{t-1} = m'\} \\ &\quad \cdot \Pr\{\mathbf{Y}_{t+1}^L | S_t = \text{newstate}(i, m')\} \\ &= \alpha_{t-1}(m') \cdot \gamma_t(i, m') \cdot \beta_t(\text{newstate}(i, m')) \end{aligned}$$

Αν υποθέσουμε ότι οι σχηματισμοί τερματίζονται σε κατάσταση 0, έχουμε $\alpha_0(0) = 1$, και $\alpha_0(m) = 0$, $m = 1, 2, \dots, 2^M - 1$. Μπορούμε να υπολογίσουμε το α με τη μέθοδο της αναδρομής προς τα εμπρός

$$\begin{aligned} \alpha_t(m) &= \sum_{i=0,1} \Pr\{d_t=i, S_{t-1}=\text{oldstate}(i,m), \mathbf{Y}_1^t\} \\ &= \sum_{i=0,1} \Pr\{S_{t-1}=\text{oldstate}(i,m), \mathbf{Y}_1^{t-1}\} \cdot \Pr\{d_t=i, \mathbf{Y}_t | S_{t-1}=\text{oldstate}(i,m)\} \\ &= \sum_{i=0,1} \alpha_{t-1}(\text{oldstate}(i,m)) \cdot \gamma_t(i, \text{oldstate}(i,m)) \end{aligned}$$

Στο τέλος του σχηματισμού έχουμε $\beta_L(0) = 1$, και $\beta_L(m) = 0$, $m = 1, 2, \dots, 2^M - 1$.

Μπορούμε να υπολογίσουμε το β με τη μέθοδο της αναδρομής προς τα πίσω

$$\begin{aligned} \beta_t(m) &= \sum_{i=0,1} \Pr\{d_{t+1}=i, \mathbf{Y}_{t+1}^L | S_t=m\} \\ &= \sum_{i=0,1} \Pr\{d_{t+1}=i, \mathbf{Y}_{t+1}^L | S_t=m\} \cdot \Pr\{\mathbf{Y}_{t+2}^L | S_{t+1}=\text{newstate}(i,m)\} \\ &= \sum_{i=0,1} \gamma_{t+1}(i,m) \cdot \beta_{t+1}(\text{newstate}(i,m)) \end{aligned}$$

Αν οι σχηματισμοί δεν έχουν τερματιστεί δεν έχουμε γνώση των αρχικών και τελικών συστημάτων. Σε αυτή την περίπτωση πρέπει να χρησιμοποιήσετε $\alpha_0(m) = \beta_L(m) = 2^{-M}$

Επειδή το $\alpha_t(m) = \Pr\{S_t=m, \mathbf{Y}_1^t\}$ γίνεται πολύ μικρό με την αύξηση του t πρέπει να χρησιμοποιήσουμε κάποια επανακλιμάκωση. Κατά σειρά η λειτουργία $\alpha'_t(m)$ πρέπει

$$\alpha'_t(m) = \Pr\{S_t=m | \mathbf{Y}_1^t\} = \frac{\Pr\{S_t=m, \mathbf{Y}_1^t\}}{\Pr\{\mathbf{Y}_1^t\}}$$

να χρησιμοποιηθεί

Όπου το $\Pr\{\mathbf{Y}_1^t\}$ καθορίζεται ως το άθροισμα του $\alpha_t(m)$ από όλες τις εξισώσεις, πράγμα που σημαίνει ότι οι τιμές του $\alpha'_t(m)$ πάντα καταλήγουν σε μία. Ωστόσο, δεδομένου ότι η έξοδος είναι η αναλογία πιθανοφάνειας, η πραγματική ανακλιμάκωση δεν είναι σημαντική όσο αποφεύγονται οι υπορροές. Το $\beta_t(m)$ χρειάζεται παρομοίως ανακλιμάκωση.

Ο αλγόριθμος που αναπαριστάται εδώ προϋποθέτει το $\alpha_t(m)$ αποθηκεύεται για τον ολοκληρωμένο σχηματισμό από τη στιγμή που πρέπει να περιμένουμε το τέλος του σχηματισμού του αλγόριθμου για να μπορέσουμε να υπολογίσουμε το $\beta_t(m)$. Εμείς

αντί αυτού μπορούμε να χρησιμοποιήσουμε μια προσέγγιση με τη μέθοδο ενός συρόμενου παράθυρου με περίοδο T και περίοδο κατάρτισης T_r . Καταρχάς το $a_t(m)$ υπολογίζεται και αποθηκεύονται για $t = 0$ έως $T-1$. Ο υπολογισμός του $\beta_t(m)$ ξεκινά τη χρονική στιγμή $t=T + T_r-1$ με το αρχικούς όρους $\beta_{T+T_r-1}(m) = 2^{-M}$. Οι πρώτες τιμές T_r $\beta_t(m)$ απορρίπτονται αλλά μετά την περίοδο της κατάρτισης, δηλαδή για $t = T-1$ μέχρι το 0 , υποθέτουμε ότι το $\beta_t(m)$ είναι σωστό και έτοιμο για τον υπολογισμό του $\sigma_t(i, m')$. Μετά το πρώτο παράθυρο θα συνεχίσουμε με το επόμενο μέχρι να φτάσουμε στο τέλος του σχηματισμού-αλγόριθμου όπου χρησιμοποιούμε τις πραγματικούς τελικούς όρους για $\beta_t(m)$.

Φυσικά, αυτή η προσέγγιση είναι προσεγγιστική, αλλά αν η περίοδος κατάρτισης επιλεγεί προσεκτικά η υποβάθμιση των επιδόσεων μπορεί να είναι πολύ μικρή.

Δεδομένου ότι έχουμε μόνο μία έξοδο που σχετίζεται με κάθε μετάβαση, μπορούμε να υπολογίσουμε το $\gamma_t(i, m')$ ως

$$\gamma_t(i, m') = \Pr_{\text{apriori}}\{d_t = i\} \cdot \Pr\{\mathbf{Y}_t \mid d_t = i, S_{t-1} = m'\}$$

Για τους turbo κώδικες, ο «εκ των προτέρων» πληροφορίες γενικώς φτάνουν με καλό λόγο πιθανοφάνειας. Ευτυχώς όπως βλέπουμε από τον υπολογισμό του $a_t(m)$ και $\beta_t(m)$, το $\gamma_t(i, m')$ χρησιμοποιείται πάντα σε ζεύγη $-\gamma_t(0, m')$ και $\gamma_t(1, m')$. Αυτό σημαίνει ότι μπορούμε να πολλαπλασιάσουμε το $\gamma_t(i, m')$ με μια σταθερά

$$k_t = \frac{1}{\Pr_{\text{apriori}}\{d_t = 0\}}$$

και να πάρουμε :

$$\gamma'_t(1, m') = \frac{\Pr_{\text{apriori}}\{d_t=1\}}{\Pr_{\text{apriori}}\{d_t=0\}} \cdot \Pr\{\mathbf{Y}_t \mid d_t=1, S_{t-1}=m'\}$$

$$\gamma'_t(0, m') = \Pr\{\mathbf{Y}_t \mid d_t=0, S_{t-1}=m'\}$$

Για μια πραγματική εφαρμογή οι τιμές του $\alpha_t(m)$, $\beta_t(m)$ και $\gamma_t(i, m')$ μπορεί να παρασταθεί ως ο αρνητικός λογάριθμος ως προς τις πραγματικές πιθανότητες. Αυτή είναι επίσης κοινή πρακτική για τους αποκωδικοποιητές Viterbi όπου οι μετρήσεις του υποσυστήματος και οι μετρήσεις των συστημάτων είναι λόγος σε σχέση με τις αντίστοιχες πιθανότητες.

Με την λογαριθμική παρουσίαση, ο πολλαπλασιασμός προστίθενται και η προσθήκη πραγματοποιείται ως μια λειτουργία, την «E» όπου

$$x \text{ E } y = -\log(e^{-x} + e^{-y}) = \min(x, y) - \log(1 + e^{-|y-x|})$$

Αυτή η λειτουργία μπορεί να περιοριστεί στην εύρεση του ελάχιστου και να βελτιωθεί προσθέτοντας ένα μικρό συντελεστή διόρθωσης.

2. Συστήματα επικοινωνίας 3G

Ο όρος 3G είναι ένας γενικός όρος που καλύπτει ένα φάσμα τεχνολογιών ασύρματων δικτύων, συμπεριλαμβανομένων των WCDMA, CDMA2000, UMTS και EDGE. Το UMTS είναι μια τεχνολογία μετάδοσης ομιλίας και δεδομένων υψηλής ταχύτητας που είναι μέρος της IMT-2000 τρίτης γενιάς (3G) ασύρματων προτύπων της Διεθνούς Ένωσης Τηλεπικοινωνιών (ITU). Το Wideband CDMA (WCDMA) είναι η τεχνολογία που χρησιμοποιείται στο UMTS και όσον αφορά τις επικοινωνίες 3G, είναι το πρότυπο διασύνδεσης τηλεπικοινωνιών και δεδομένων. Ως αποτέλεσμα, οι όροι UMTS και WCDMA συχνά χρησιμοποιούνταν ο ένας αντί του άλλου, έτσι ώστε σήμερα στον κόσμο των τηλεπικοινωνιών το WCDMA να ταυτίζεται με το UMTS (του οποίου αποτελεί μέρος). Το EDGE - Enhanced Data for Global Evolution, είναι ένα πρότυπο επικοινωνιών το οποίο έδινε στο σύστημα επικοινωνιών GSM τη δυνατότητα να μπορεί να ανταπεξέλθει στα στάνταρ των δικτύων 3G όσον αφορά τον τομέα ταχύτητας μετάδοσης δεδομένων. Παρόλα αυτά δεν καταφερε να υπάρξει ανταγωνιστικό στον τομέα των τηλεπικοινωνιών και συχνά αναφέρεται ως δίκτυο 2.75G.

2.1 Turbo κωδικές σε συστήματα 3G

Ενώ στα συστήματα ασύρματης επικοινωνίας 3G χρησιμοποιήθηκαν κατά κόρον οι τυπικοί αναδιατάκτες με τις μεθόδους τυπικής κωδικοποίησης και αποκωδικοποίησης, σύντομα παρουσιάστηκε ένα πιο συγκεκριμένα εξελιγμένο σύστημα κωδικοποίησης turbo. Παρακάτω θα αναλύσουμε τους Code Matched Interleavers (CMI), οι οποίοι παρεμβολείς-αναδιατάκτες όπως φανερωνει και το όνομα τους είναι προσαρμοσμένοι στα χαρακτηριστικά των εκάστοτε συνελκτικών κωδικοποιητών, δηλαδή του εκάστοτε κώδικα. Στη περίπτωση μας θα ερευνήσουμε πως οι παρεμβολείς CMI σε συστήματα turbo κώδικων για τα δύο 3G συστήματα επικοινωνίας, WCDMA και CDMA-2000, θα αντικαταστήσουν τους απλούς αλλά πολύπλοκους στην χρήση, απλούς παρεμβολείς 3G. Ο σχεδιασμός των «προσαρμοσμένων στον κώδικα αναδιατακτών» και στα δύο συστήματα 3G

επιτυγχάνεται μελετώντας σχολαστικά το ειδικό βάρος κάθε συστήματος. Οι κώδικες αυτοί χρησιμοποιήθηκαν πρώτη φορά το 2002 και προτιμούνται πλέον για τις επικοινωνίες 3G καθώς συνδυάζοντας ψευδοτυχαίους παρεμβολείς και S-τυχαίους παρεμβολείς, εξαλείφουν τις λέξεις χαμηλού βάρους που έχει ο κώδικας ή την μείωση οσως δεν μπορούν να εξαλειφθούν. Τα αποτελέσματα προσομοίωσης για το 3G σύστημα κωδικών turbo που χρησιμοποιεί προσαρμοσμένους κώδικες παρεμβολέων για $N = 400$ (WCDMA) και $N = 378$ (CDMA2000), με ρυθμό κώδικα 1/3 με αποκωδικοποιητή Log-MAP πάνω σε επιπρόσθετο «Λευκό θόρυβο Gaussian» (Additive White Gaussian Noise – AWGN) και κανάλι εξασθένισης Rayleigh λαμβάνονται και συγκρίνονται με συστήματα κωδικοποίησης turbo με απλούς τυπικούς παρεμβολείς 3G. Από τα αποτελέσματα των επιδόσεων, παρατηρείται ότι το 3G turbo σύστημα κωδικοποίησης το οποίο χρησιμοποιεί CMI αποδίδει καλύτερα από τον τυπικό κώδικα turbo 3G με κέρδος κωδικοποίησης 0,15-0,35 dB.

Κωδικοί Turbo στο σύστημα WCDMA:

Το WCDMA είναι μια εξέλιξη του GSM και χρησιμοποιεί ένα ευρύτερο φάσμα και η σηματοδότηση είναι με βάση το σύστημα GSM. Το πρότυπο CDMA (3GPP TS 25.2123G) επιτρέπει ένα εύρος μεγέθους πλαισίου 40-5114 bits με ρυθμό κώδικα 1/3 και 1/2. Ο κωδικοποιητής WCDMA χρησιμοποιεί 8 όμοιους RSC κωδικοποιητές με έναν εσωτερικό παρεμβολέα (ETSI, 2000). Ο εσωτερικός παρεμβολέας αποτελείται από bits εισόδου σε μια ορθογώνια μήτρα με γέμισμα, ενδο-σειρά και μεταλλαγές μεταξύ της ροής του ορθογώνιου πλέγματος και bits εξόδου από τον ορθογώνιο πλέγμα με περικοπή. Ο πολυωνυμικός κώδικας που χρησιμοποιείται στο σύστημα είναι (13/11) και ο αλγόριθμος Log-MAP χρησιμοποιείται στη διαδικασία αποκωδικοποίησης.

Κωδικός Turbo στο σύστημα cdma2000:

Ο CDMA2000 turbo κωδικοποιητής αποτελείται από ένα ζεύγος ρυθμού 1/3 συστηματικούς συνελκτικούς κωδικοποιητές με μήκος περιορισμού $K = 4$ που

χωρίζονται από έναν παρεμβολέα (3GPP2, 2002). Ο παρεμβολέας αποτελείται από μπλοκ παρεμβολέα, δείκτη ποιότητας πλαισίου και δεσμευμένα bits εισόδου. Το σύστημα CDMA2000 χρησιμοποιεί RSC κωδικοποιητή με πολυώνυμικό κώδικα $[(13:15) / 11]$. Το μέγεθος πλαισίου του συστήματος αυτού (καθώς και του διεμπλοκέα) πρέπει να είναι μία από τις ακόλουθες ειδικές τιμές: 378, 570, 762, 1146, 1530, 2398, 3066, 4602, 6138, 9210, 12282 ή 20730 bits με ρυθμό κώδικα από 1/2, 1/3 και 1/4. Ο Log-MAP χρησιμοποιείται ως αλγόριθμος αποκωδικοποίησης.

2.2 Προτεινόμενος κώδικας για συνδυασμό με το σχεδιασμό του αναδιατάκτη

Ένας καλός σχεδιασμός αναδιατάκτη για έναν κώδικα turbo είναι αυτός που παράγει υψηλή απόδοση βάρους (Αντερσεν και Zylyabov, 1997, Divsalar και Pollara, 1995). Ο προτεινόμενος προσαρμοσμένος κώδικας αναδιατακτών (CMI) θα μπορούσε να αντικαταστήσει τους υπάρχοντες εσωτερικούς αναδιατάκτες 3G. Ο CMI αναδιατάκτης χωρίζεται σε δύο μέρη: (1) τον S-τυχαίο αναδιατάκτη και (2) τον ψευδοτυχαίο

αναδιατάκτη. Η αρχή πίσω από τον S-τυχαίο αναδιατάκτη είναι να διαχωρίσει τη θέση των στοιχείων μέσα σε ένα παράθυρο όσο το δυνατόν περισσότερο, έτσι ώστε κάθε δύο στοιχεία εντός ενός παραθύρου μεγέθους S δεν θα εντοπιστούν σε ένα παράθυρο μεγέθους S στην ίδια στιγμή. Η ιδέα στην οποία βασίζεται ο ψευδοτυχαίος σχεδιασμός αναδιατάκτη είναι να αφαιρέσει κάποια φαινομενική σειρά έτσι ώστε οι ακολουθίες εξόδου ισοτιμίας χαμηλού βάρους δεν θα συμβαίνουν και στους δύο άνω και κάτω συστατικούς κωδικοποιητές την ίδια στιγμή. Ο σχεδιασμός του αναδιατάκτη CMI μπορεί να είναι ιδιαιτερωθεί ως εξής (Feng et al., 2002): Εξάλειψη των λέξεων κώδικα χαμηλού βάρους με αποτέλεσμα τη βελτίωση των ποσοστών σφαλμάτων, μείωση των λέξεων χαμηλού βάρους του κώδικα οι οποίες δεν μπορούν να εξλειφθούν.

Η εξάλειψη μιας συγκεκριμένης λέξης κώδικα μπορεί να γίνει με το διαχωρισμό της μορφής εισόδου που δημιουργεί τη συγκεκριμένη λέξη κώδικα. Ένας αναδιατάκτης με S-τυχαίο περιορισμό μπορεί είτε να διακόψει μικρού βάρους, ακολουθίες 3

εισόδων με μήκη έως $S + 1$ ή να το επεκτείνει σε ένα μεγαλύτερο με μήκη πάνω από 2 ($S + 1$), όπου $S < 2$. Ως εκ τούτου, εξετάζουμε τον διαχωρισμό των βάρους-2 και βάρους-4 πρότυπων εισόδου στο σχεδιασμό του αναδιατάκτη μας. Ο αλγόριθμος για να υπολογιστεί η ελεύθερη απόσταση του κώδικα turbo βασίζεται στους επιμέρους κώδικες 3G turbo, δηλαδή, ένα υποσύνολο ενός κώδικα που ορίζεται μέσω περιορισμών στις άκρες των δικτυακών διαγραμμάτων του και επιτρέπει τον υπολογισμό μεγάλων αποστάσεων για μεγάλους αναδιατάκτες χωρίς περιορισμό ως προς το βάρος ακολουθίας εισόδου. Το φάσμα βάρους των συστημάτων 3G με $N = 400$ (WCDMA) και $N = 378$ (cdma2000), με ρυθμό κώδικα $1/3$ φαίνονται στον Πίνακα 1 και 2.

Από τον Πίνακα 1 και 2, εντοπίσαμε τις κωδικές λέξεις που θα πρέπει να σπάσουν είναι εκείνες με βάρος 12, 14, 16, 18 και 20. Εντοπίσαμε ότι οι λέξεις κώδικα με βάρη 12 και 16 δημιουργούνται από πρότυπα εισόδου με βάρη 3, 4, 5, 6, 7, 8, 9 για WCDMA και 3, 4, 5, 6, 7 για cdma2000. Ο κώδικας με βάρη 14, 18 και 20 δημιουργείται από μοτίβα εισόδου με βάρος 2, 3, 4, 5, 6, 7, 8 για WCDMA και 2, 3, 4, 5, 6 και 9 για cdma2000.

ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΠΛΗΡΟΦΟΡΙΚΗΣ

Πίνακας 1. Το φάσμα βάρους του συστήματος WCDMA με και χωρίς αναδιατάκτη

<u>Βάρος</u>	<u>Αριθμός λέξεων με αναδιατάκτη</u>	<u>Αριθμός λέξεων χωρίς αναδιατάκτη</u>
8	-	13
10	-	7
12	1	19
14	1	67
16	3	109
18	6	72
20	7	191
22	21	401
24	30	346
26	50	612
28	76	868
30	130	968
32	264	843
34	478	1320
36	800	1640
38	1400	1312
40	2307	1213

ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΠΛΗΡΟΦΟΡΙΚΗΣ

Πίνακας 2. Το φάσμα βάρους του συστήματος CDMA2000 με και χωρίς αναδιατάκτη

<u>Βάρος</u>	<u>Αριθμός λέξεων με αναδιατάκτη</u>	<u>Αριθμός λέξεων χωρίς αναδιατάκτη</u>
8	-	11
10	-	7
12	1	21
14	1	63
16	3	109
18	5	67
20	6	170
22	19	387
24	21	324
26	48	600
28	65	868
30	124	947
32	198	840
34	438	1205
36	708	1402
38	1297	1201
40	2088	1103

Και για τα δύο συστήματα 3G, τα μοτίβα εισόδου βάρους 2 τα οποία παράγουν χαμηλότερο βάρος ισοτιμίας ελέγχου της ακολουθίας C_2 είναι $X_2=(00..010000100...)$, όπου η απόσταση μεταξύ δύο 1 είναι $\mu = 5$. Ο έλεγχος της ισοτιμίας $C_2=(00..010000100...)$, όπου $z_{min} = 4$. Αν ο αναδιατάκτης αντιστοιχίζει την είσοδο ακολουθίας βάρους 2 με σειρά μ σε μια σειρά με το ίδιο βάρος και μ , το προκύπτον βάρος συνολικά της κωδικής λέξης είναι:

$$D = 2 + 2 z_{min} \quad (1)$$

Μία ακολουθία εισόδου βάρους 2 που παράγει ένα πεπερασμένη ακολουθία ελέγχου βάρους της ισοτιμίας μπορεί να αναλυθεί σύμφωνα με τον Divsalar και Pollara (1995):

ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΠΛΗΡΟΦΟΡΙΚΗΣ

$$X_2(D) = (1 + D^{\mu k_1}) D^{\tau_1} \quad (2)$$

όπου, $k_1=1, 2, 3, \dots$ και τ_1 είναι η χρονική καθυστέρηση. Ο έλεγχος της ισοτιμίας του βάρους της εξίσωσης 2 δίνεται από:

$$k_1 (z_{\min} - 2) + 2 \quad (3)$$

Εάν οι χάρτες αναδιατακτών (2) για $X'_2(D) = (1 + D^{\mu k_2}) D^{\tau_2}$, το συνολικό βάρος της δημιουργημένης κωδικής λέξης δίνεται από:

$$d = w + w(y_1) + w(y_2) \quad (4)$$

όπου, y_1 και y_2 είναι η πρώτη και η δεύτερη έξοδος κωδικοποιητή, αντίστοιχα. Στη συνέχεια, το συνολικό βάρος της κωδικής λέξης που παράγεται από ακολουθία εισόδου βάρους 2 είναι:

$$d = 6 + (k_1 + k_2) (z_{\min} - 2) \quad (5)$$

Εξετάζουμε το ενδεχόμενο να σπάσουμε τα πρότυπα εισόδου που δημιουργούν λέξεις κώδικα με όχι περισσότερο από 20 μόνο βάρος. Έτσι, έχουμε

$$d = 6 + (k_1 + k_2) (z_{\min} - 2) \leq d^2_{\max} \quad (6)$$

το οποίο ισοδυναμεί με

$$k_1 + k_2 \leq d^2_{\max} - 6 / z_{\min} - 2 \quad (7)$$

Ας είναι οι j_1 και j_2 οι θέσεις των 1's στην βάρους-2

ακολουθία εισόδου, όπου j_1, j_2 ανήκουν στο A και $a(j_1)$ και $a(j_2)$ ανήκουν στο A . Οι ακόλουθες προϋποθέσεις χαρτογράφησης θα πρέπει να πληρούνται για να αποφευχθεί η χαρτογράφηση της ακολουθίας εισόδου σε άλλη ακολουθία εισόδου με βάρους-2 που δημιουργεί ένα πεπερασμένο μήκος λέξης κώδικα:

$$| a(j_1) - a(j_2) | \bmod \mu \neq 0 \text{ όπου } | j_1 - j_2 | \bmod \mu = 0 \quad (8)$$

όπου, μ είναι η ελάχιστη απόσταση μεταξύ δύο 1's στο πρότυπο εισόδου με βάρος-2 που δημιουργεί το πεπερασμένο βάρος της λέξης κώδικα. Εξετάζουμε ακολουθίες εισόδου βάρους-4 ως ένωση δύο μονών ακολουθιών βάρους-2:

$$X_4(D) = (1 + D^{\mu k'1}) D^{\tau1} + (1 + D^{\mu k'2}) D^{\tau2} \quad (9)$$

Η είσοδος στο δεύτερο κωδικοποιητή,

$$X'_4(D) = (1 + D^{\mu k'3}) D^{\tau3} + (1 + D^{\mu k'4}) D^{\tau4}$$

όπου, $t_2 > t_1 + \mu k'1$, $t_4 > t_3 + \mu k'1$. Το συνολικό βάρος της δημιουργημένης λέξης κώδικα μπορεί να υπολογιστεί από την εξίσωση 4:

$$d = 4 + y1(4) + y2(4) / (z_{\min} - 2) = 12 + (k'_1 + k'_2 + k'_3 + k'_4),$$

η οποία είναι αντιστοιχεί σε

$$k'_1 + k'_2 + k'_3 + k'_4 \leq d_{\max}^4 - 1 / (z_{\min} - 2) \quad (10)$$

Ας είναι τα j_1, j_2, j_3 και j_4 οι θέσεις των 1s στην βάρους-4 ακολουθία εισόδου, όπου j_1, j_2, j_3 και $j_4 \in A$ και $j_1 < j_2 < j_3 < j_4$. Έστω $\alpha(j_1), \alpha(j_2), \alpha(j_3)$ και $\alpha(j_4)$ είναι οι θέσεις των 1's στο αναδιάτακτη ακολουθία εισόδου $\alpha(j_1), \alpha(j_2), \alpha(j_3)$ και $\alpha(j_4) \in A$. Οι ακόλουθοι όροι χαρτογράφησης θα πρέπει να πληρούνται για να αποφευχθεί η χαρτογράφηση της ακολουθίας εισόδου που περιλαμβάνει δύο πρότυπα εισόδου βάρους-2, η οποία δημιουργεί πεπερασμένου μήκους λέξης κώδικα.

$$| \alpha(j_1) - \alpha(j_3) | \bmod \mu \neq 0 \quad \text{και} \quad | \alpha(j_2) - \alpha(j_4) | \bmod \mu \neq 0$$

$$\text{όταν} \quad | (j_1) - (j_2) | \bmod \mu = 0 \quad \text{και} \quad | (j_3) - (j_4) | \bmod \mu = 0 \quad \text{ή,}$$

$$| \alpha(j_1) - \alpha(j_4) | \bmod \mu \neq 0 \quad \text{και} \quad | \alpha(j_2) - \alpha(j_3) | \bmod \mu \neq 0$$

$$\text{όταν} \quad | (j_1) - (j_2) | \bmod \mu = 0 \quad \text{και} \quad | (j_3) - (j_4) | \bmod \mu = 0.$$

Ωστόσο, και στις δύο περιπτώσεις για τα πρότυπα εισόδου, βάρους-2 και βάρους-4, μόνο εκείνα τα πρότυπα εισόδου που παράγουν χαμηλού βάρος κωδικές λέξεις με μεγάλη συμβολή στις ανάγκες των επιδόσεων πρέπει να χωρίζονται. Έτσι, για αμφότερες τις περιπτώσεις, έχουμε να εξετάσουμε το πρότυπο εισόδου που παράγει κωδικές λέξεις με βάρος όχι μεγαλύτερο από 20 και μόνο. Η επίδραση των πρότυπων

εισόδου με βάρος μεγαλύτερο από 4 στην απόδοση του κώδικα είναι μικρή και τυπικά μπορούν να διαχωριστούν από τον S-τυχαιο περιορισμό. Ο αλγόριθμος που χρησιμοποιείται (Yuan,1999) για τον σχεδιασμό αναδιατάκτη περιγράφεται ως εξής:

- Δημιουργούμε έναν τυχίο ακέραιο από το σύνολο $(1, 2, 3, \dots, N)$, όπου N είναι το μέγεθος του αναδιατάκτη. Ο επιλεγμένος ακέραιος αντιπροσωπεύει τη θέση αναδιατάκτη.
- Κάθε τυχαία επιλεγμένος ακέραιος συγκρίνεται με τους S προηγούμενους επιλεγμένους ακεραίους. Αν η απόλυτη τιμή της διαφοράς μεταξύ του τρέχων επιλεγμένου ακεραίου και οποιουδήποτε από τους S προηγούμενους επιλεγμένους ακεραίους είναι μικρότερη από το S , τότε να απορρίπτει τον τρέχων επιλεγμένο ακέραιο και να πάει στο βήμα 1.
- Έλεγχος αν το αποτέλεσμα του τρέχων αναδιατάκτη πληροί τις προϋποθέσεις χαρτογράφησης που αναλύονται στις Εξισώσεις 7, 8, 10 και 11, εάν όχι, να πάει ξανά στο βήμα 1.
- Εάν δεν υπάρχει ακέραιος αριθμός από το σύνολο που να μπορεί να ικανοποιήσει και τα δύο βήματα 2 και 3 ταυτοχρόνως, για ένα μεγάλο αριθμό επαναλήψεων, τότε να μειώσει το S κατά 1 και να ξεκινήσει την αναζήτηση πάλι.
- Αποθήκευση του τρέχων ακεραίου. Η διαδικασία επαναλαμβάνεται μέχρι να επιλεγούν όλοι οι ακέραιοι N .
- Έξοδος.

2.3 ΠΡΟΣΟΜΟΙΩΣΗ

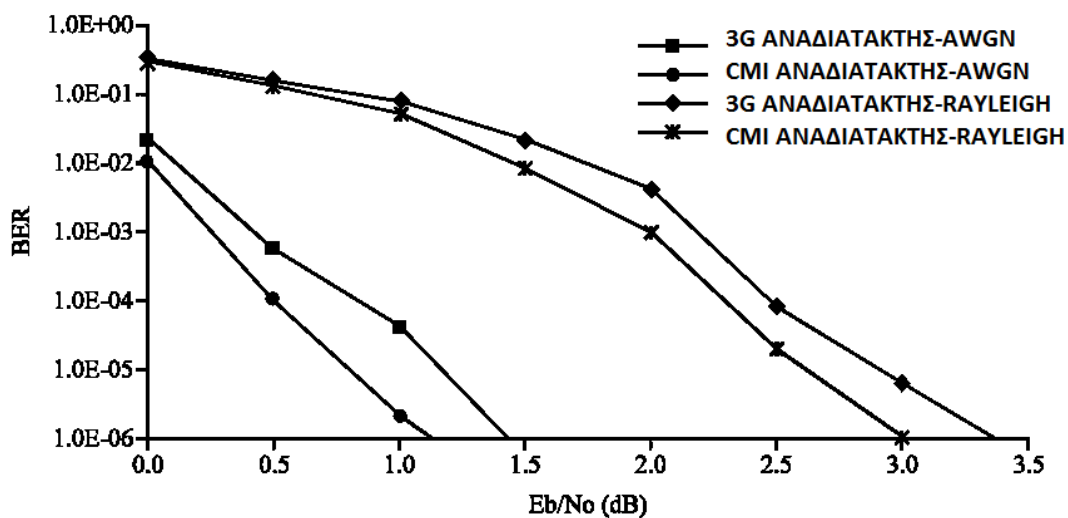
Για να βεβαιωθούμε για την καταλληλότητα του CMI-Code Matched Interleaver και για τα δύο πρότυπα ασύρματης επικοινωνίας 3G, κάνουμε την προσομοίωση της απόδοσης των δύο 3G συστημάτων κωδικών turbo, χρησιμοποιώντας 3G αναδιατάκτες και τους προτεινόμενους CMI. Όσον αφορά το σύστημα WCDMA, οι

προσομοιώσεις πραγματοποιήθηκαν για μήκος μπλοκ πληροφοριών, $N = 5114$, ρυθμό κώδικα 1/3 χρησιμοποιώντας αποκωδικοποιητή Log-MAP με μέγιστο αριθμό επαναλήψεων αποκωδικοποίησης να είναι 8 και 14 επί θορύβου AWGN και ασυσχέτιστου καναλιού εξασθένησης Rayleigh. Στο σύστημα cdma2000, προσομοιώσεις εκτελέστηκαν για μήκος μπλοκ πληροφοριών, $N = 6138$, ρυθμό κώδικα 1/3 χρησιμοποιώντας αποκωδικοποιητή Log-MAP με μέγιστο αριθμό επαναλήψεων αποκωδικοποίησης να είναι 9 για το κανάλι AWGN και 12 για ασυσχέτιστο κανάλι εξασθένησης Rayleigh. Πλάτος εξασθένησης $\alpha = 1$ για το κανάλι AWGN και 0,8662 για το κανάλι Rayleigh. Τα αποτελέσματα των δύο προσομοιώσεων φαινονται στο σχήμα 3 και 4 .

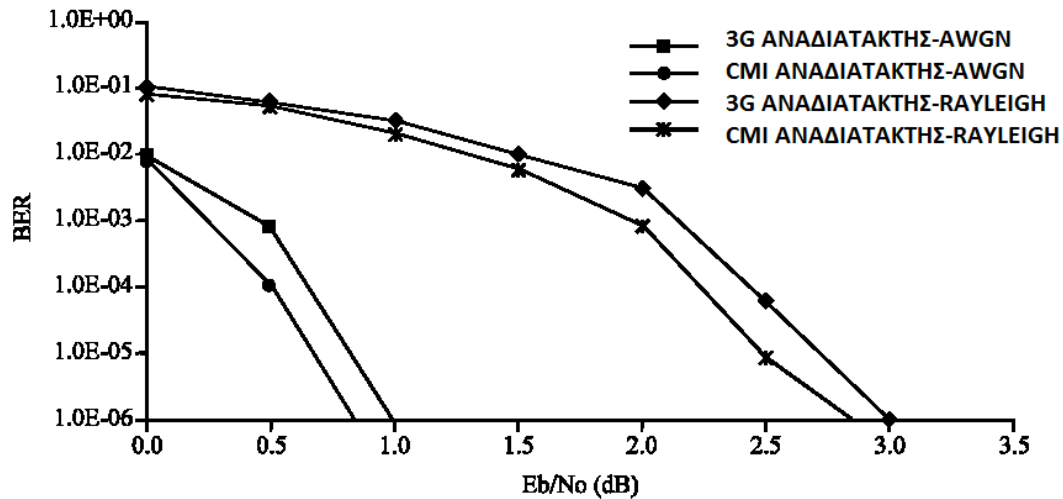
Από το σχήμα 3, παρατηρείται ότι το σύστημα με τους CMI αναδιατάκτες, αποδίδει καλύτερα από ό, τι το σύστημα με τους τυπικούς αναδιατάκτες 3G και στις δύο περιπτώσεις με κέρδος κωδικοποίησης περίπου 0.2-0.35 dB.

Στο σχήμα 4, διαπιστώνεται ότι το σύστημα με τους CMI αναδιατάκτες, αποδίδει καλύτερα από ό, τι το σύστημα με τους τυπικούς αναδιατάκτες 3G και στις δύο περιπτώσεις με κέρδος κωδικοποίησης περίπου 0.15-0.3 dB.

Σχήμα 3: Αποτελέσματα προσομοίωσης για το σύστημα WCDMA.



Σχήμα 4: Αποτελέσματα προσομοίωσης για το σύστημα CDMA2000



Έχοντας λοιπόν ερευνήσει την χρήση του προσαρμοσμένου στον κώδικα αναδιατάκτη στο 3G σύστημα κωδικοποίησης turbo, που ταιριάζει και στα δύο Πρότυπα Επικοινωνίας 3G, WCDMA και cdma2000. Ο αναδιατάκτης CMI αναλύθηκε και μελετήθηκε σχολαστικά στα φάσματα βάρους και των δύο 3G συστημάτων turbo. Τα αποτελέσματα της προσομοίωσης δείχνουν την απόδοση 3G συστημάτων κωδικοποίησης turbo με CMI αναδιατάκτες να αποδίδουν καλύτερα από ό, τι τα συστήματα κωδικοποίησης turbo με τυπικούς 3G αναδιατάκτες. Το κέρδος κωδικοποίησης του συστήματος είναι από 0,15 έως 0,35 dB για τις διάφορες συνθήκες του εκάστοτε καναλιού.

3. Συστήματα επικοινωνίας 4G

Τα συστήματα επικοινωνίας 4G ακολουθούν τις τρεις προηγούμενες γενιές της κινητής τηλεφωνίας οι οποίες περιλαμβάνουν τα 1G, 2G και τα συστήματα 3G που αναλύσαμε. 4G ονομάζεται το ασύρματο πρότυπο επικοινωνίας 4^{ης} γενιάς (4th generation wireless communication standard) το οποίο περιλαμβάνει τις τεχνολογίες

LTE, WiMAX, και UMB. Σε σχέση με τα συστήματα 3^{ης} γενιάς, τα συστήματα 4G παρέχουν ταχύτερες ταχύτητες αποστολής και λήψης πακέτων για VoIP (Voice over IP), μεγάλες ποσότητες δεδομένων και λειτουργούν σωστότερα σε εφαρμογές συνεχούς ροής δεδομένων (stream).

LTE - Long Term Evolution

Το LTE έχει σχεδιαστεί για να αντικαταστήσουν τα παλαιότερα κινητά ασύρματα δίκτυα με μια νέα και βελτιωμένη ταχύτητα για τους χρήστες κινητών με ασύρματη σύνδεση στο Internet. Το πλεονέκτημα του LTE είναι ότι έχει τη δυνατότητα να αυξήσει τη χωρητικότητα του δικτύου επιτρέποντας περισσότερα δεδομένα να διαβιβάζονται με το ίδιο εύρος ζώνης που χρησιμοποιείται από τις κινητές συσκευές 3G.

Το LTE λειτουργεί μέσω ραδιοκυμάτων και είναι σε θέση να συνδεθεί στα τρέχον δίκτυα για να παρέχει μια εύκολη μετάβαση από το 3G. Επιτρέπει επίσης στις τρέχουσες εφαρμογές στους τομείς των τηλεπικοινωνιών να τρέχουν πιο γρήγορα και ανοίγει την πόρτα για την ραγδαία ανάπτυξη εφαρμογών κινητής επικοινωνίας για το μέλλον. Για τους λόγους αυτούς και περισσότερους έχει επικρατήσει πλέον και αναφέρεται ως ασύρματο Internet 4G.

WiMAX - Worldwide Interoperability for Microwave Access

Το WiMAX, αντιπροσωπεύει στην ουσία ένα συνασπισμός που βοηθά στην καθιέρωση προτύπων για τα ευρυζωνικά δίκτυα ασύρματης πρόσβασης. Είναι διαθέσιμο περισσότερο καιρό από το LTE και προσφέρει υπηρεσίες ασύρματου Internet 4G για το σπίτι αλλά και εκτός. Αυτό το είδος της σύνδεσης 4G στο Internet είναι κυρίως διαδεδομένο στις Ηνωμένες Πολιτείες, όπου το LTE είναι κάπως περιορισμένο, ιδιαίτερα σε σχέση με άλλες χώρες.

Το WiMAX είναι ικανό να υποστηρίξει τόσο κινητή όσο και σταθερή ασύρματη συνδεσιμότητα, μορφή επικοινωνίας η οποία είναι ελάχιστα διαδεδομένη στην Ελλάδα.

UMB - Ultra Mobile Broadband

Το UMB είναι ένα ακόμα πρότυπο επικοινωνίας το οποίο πληρούσε τα στοιχεία για να θεωρείται 4^{ης} γενιάς. Αυτό το είδος ασύρματου Internet 4G χρησιμοποιούσε μεγάλο εύρος ζώνης και χαμηλή λανθάνουσα κατάσταση μέσω ενός δικτύου TCP / IP

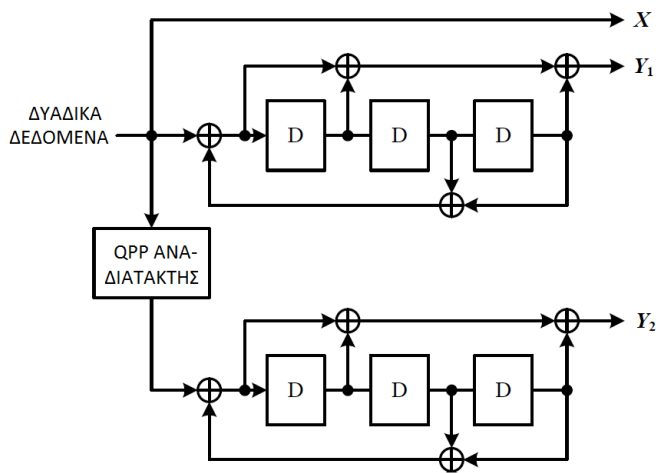
που παρείχε προηγμένες υπηρεσίες για το σκοπό της συνεχούς ροής βίντεο και online gaming για την τεχνολογία των κινητών συσκευών. Θα ήταν η συνέχεια του προτύπου CDMA-2000, εξαλείφοντας πολλά από τα μειονεκτήματα του χρησιμοποιώντας τον πολυπλέκτη OFDMA. Αναφερόμαστε σε αυτό σε παρατατικό χρόνο γιατί από τον Νοέμβριο του 2008, η Qualcomm, κύρια εταιρεία εξέλιξης του UMB, ανακοίνωσε ότι περατώνει την ανάπτυξη της τεχνολογίας και στράφηκε και αυτή στην ανάπτυξη του LTE. Έτσι ενώ στα πρότυπα επικοινωνίας 3^{ης} γενιάς υπάρχει εξέλιξη για χώρες ανά τον κόσμο, το πρότυπο 4^{ης} γενιάς UMB δεν θεωρείται ενεργό καθώς καμία χώρα από όσες έχουν δίκτυα 4G δεν το χρησιμοποιεί. Περαιτέρω, οι εταιρείες κινητών τηλεπικοινωνιών μεγάλων χωρών που στα δίκτυα 3^{ης} γενιάς τους χρησιμοποιούσαν CDMA-2000 όπως οι Αυστραλία, ΗΠΑ, Καναδάς, Κίνα, Ιαπωνία και Νότια Κορέα έχουν ήδη χρησιμοποιήσει ή θα χρησιμοποιήσουν πρότυπα LTE ή WiMAX ως την τεχνολογία 4G τους.

3.1 Turbo Κώδικες σε συστήματα 4G

Οι κώδικες Turbo όπως έχουμε δει χρησιμοποιούνται κατά κόρον στις τηλεπικοινωνίες 3G, είτε με τη μορφή FEC- FORWARD ERROR CORRECTION είτε με πιο απλές μεθόδους κωδικο-αποκωδικοποίησης. Στα πρότυπα τηλεπικοινωνίας 4G χρησιμοποιούνται συνενωμένοι κώδικες που είναι μια πιο προχωρημένη μορφή διόρθωσης σφαλμάτων για την επίτευξη μετάδοσης δεδομένων υψηλής ταχύτητας. Πιο συγκεκριμένα χρησιμοποιούνται αλγόριθμοι Log-MAP και Max-Log-MAP που είχαμε αναφέρει στο δεύτερο κεφάλαιο καθώς μειώνουν την αριθμητική πολυπλοκότητα ενώ διατηρούν καλά επίπεδα αποκωδικοποίησης. Επίσης, γίνεται μεγάλη χρήση Δυαδικών κωδικών-Binary codes (LTE) και Διπλών Δυαδικών κωδικών- Double Binary codes(Wimax) οι οποίοι θα αναλυθούν παρακάτω.

3.2 Δυαδικός Κώδικας Turbo σε πρότυπο 3GPP LTE

Το σύστημα κωδικοποίησης Turbo στο πρότυπο 3GPP LTE (3GPP TS 36.212 , 2008) είναι ένας παράλληλος συνενωμένος συνελκτικός κώδικας –parallel concatenated convolutional code(PCCC)- με δύο οκταδικούς συστατικούς κωδικοποιητές και έναν αντιμεταθετικό πολυωνυμικό αναδιατάκτη δευτέρου βαθμού-quadratic permutation polynomial (QPP) interleaver. Ο ρυθμός κωδικοποίησης του κώδικα Turbo είναι 1/3. Η δομή του κωδικοποιητή Turbo παρουσιάζεται στο παρακάτω Σχήμα 1.



Σχ.1: Δομή του Turbo κωδικοποιητή με ρυθμό 1/3 στο πρότυπο LTE

Όπως φαίνεται στο σχήμα, ένας κωδικοποιητής Turbo αποτελείται από δύο δυαδικούς συνελκτικούς κωδικοποιητές που συνδέονται με ένα αναδιατάκτη. Ο βασικός ρυθμός κωδικοποίησης είναι 1/3 που σημαίνει πως N αριθμός bit δεδομένων θα πρέπει να κωδικοποιούνται σε 3N αριθμό bits δεδομένων. Ο συνάρτηση μεταφοράς συστατικού κώδικα 8 σημείων για PCCC είναι:

$$G(D) = \begin{bmatrix} 1, & g_1(D) \\ & g_0(D) \end{bmatrix},$$

Όπου

$$g_0(D) = 1 + D^2 + D^3,$$

$$g_1(D) = 1 + D + D^3.$$

Η αρχική τιμή των καταχωρητών μετατόπισης των οκταδικών συστατικών κωδικοποιητών πρέπει να είναι όλα τα μηδενικά, όταν αρχίζουν να κωδικοποιούν τα δυαδικά ψηφία εισόδου. Ο τερματισμός του χωροδικτύωματος γίνεται με τη λήψη των bits ουράς από τη ανατροφοδότηση του καταχωρητή μετατόπισης εφόσον όλα τα bits πληροφοριών έχουν κωδικοποιηθεί. Τα bits ουράς υπερπλήρη μετά την κωδικοποίηση των bits πληροφοριών.

Η λειτουργία του αναδιατάκτη είναι να λάβει κάθε εισερχόμενο μπλοκ δεδομένων N bits και να τα ανακατέψει με ένα ψευδο-τυχαίο τρόπο. Ένα από τα νέα χαρακτηριστικά του κωδικοποιητή 3GPP LTE Turbo είναι ο δευτέρου βαθμού αντιμεταθετικός πολωνυμικός εσωτερικός αναδιατάκτης (QPP). Θα δούμε αργότερα ότι αυτός ο QPP αναδιατάκτης είναι το στοιχείο κλειδί που θα επιτρέψει την παράλληλη αποκωδικοποίηση των κωδίκων Turbo.

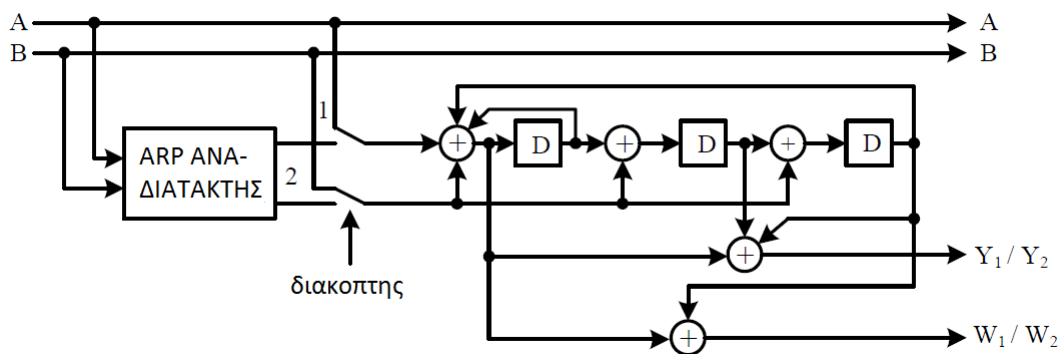
Διπλός δυαδικός κώδικας Turbo στο πρότυπο WiMax - IEEE 802.16e

Ο συνελκτικός κωδικοποιητής Turbo για το IEEE 802.16e πρότυπο (IEEE Std 802.16, 2004) – WiMax απεικονίζεται στο σχήμα 2. Χρησιμοποιεί ένα διπλό δυαδικό κυκλικό αναδρομικό συστηματικό συνελκτικό κώδικα. Ζευγάρια δεδομένων (A, B), αντί για μία ακολουθία bits, τροφοδοτούνται στον κυκλικό αναδρομικό συστηματικό συνελκτικό κωδικοποιητή δύο φορές, και τέσσερα bits ισοτιμίας (Y1, W1) και (Y2, W2) δημιουργούνται στη φυσική ροή και στην παρεμβαλλόμενη ροή, αντίστοιχα. Τα πολυώνυμα του κωδικοποιητή περιγράφονται σε δυαδική μορφή συμβόλων ως εξής:

- Για τον κλάδο ανατροφοδότησης: $1 + D + D^3$,
- Για το bit ισοτιμίας Y : $1 + D^2 + D^3$,
- Για το bit ισοτιμίας W : $1 + D^3$.

Η εφαρμογή του λεγόμενου μέτρου εξόδου δηγμάτων της ουράς του χωροδικτύωματος χρησιμοποιείται σε αντιδιαστολή με την προσθήκη επιπλέον bits ουράς. Σε αυτό

σύστημα τερματισμού, η αρχική κατάσταση του χωροδιχτύματος ισούται με την τελική του κατάσταση. Ως εκ τούτου, πρέπει να εκτελείται για διεργασία πριν την κωδικοποίηση για να καθορίσει την αρχική κατάσταση. Αυτό δεν είναι ένα σύνθετο πρόβλημα επειδή η διαδικασία κωδικοποίησης μπορεί να πραγματοποιηθεί σε ένα πολύ υψηλότερο ποσοστό. Ένας σύμβολιζόμενος, σαν κανονικός αντιμεταθετικός, (ARP) αναδιατάκτης χρησιμοποιείται στο πρότυπο WiMax, ο οποίος μπορεί να επιτρέψει παράλληλη αποκωδικοποίηση των διπλών δυαδικών κώδικων Turbo.

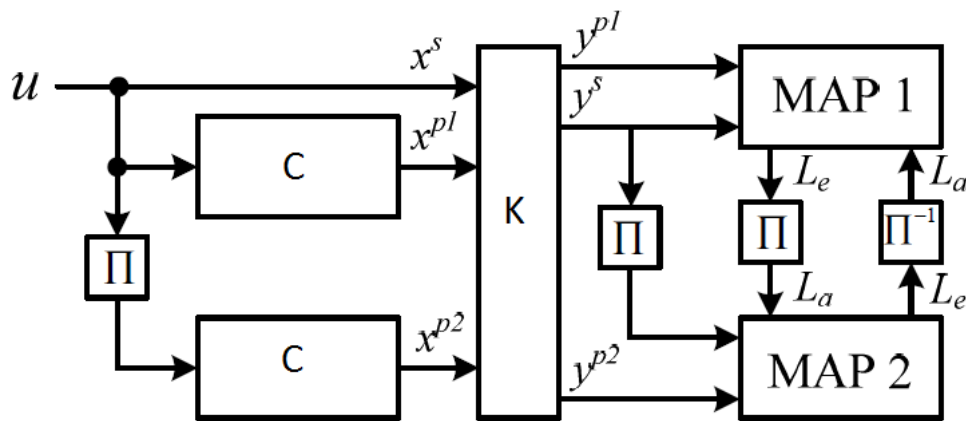


Σχήμα 2. Δομή διπλού δυαδικού κωδικοποιητή Turbo με ρυθμό 1/3 στο WiMax

3.3 Αλγόριθμος Αποκωδικοποίησης

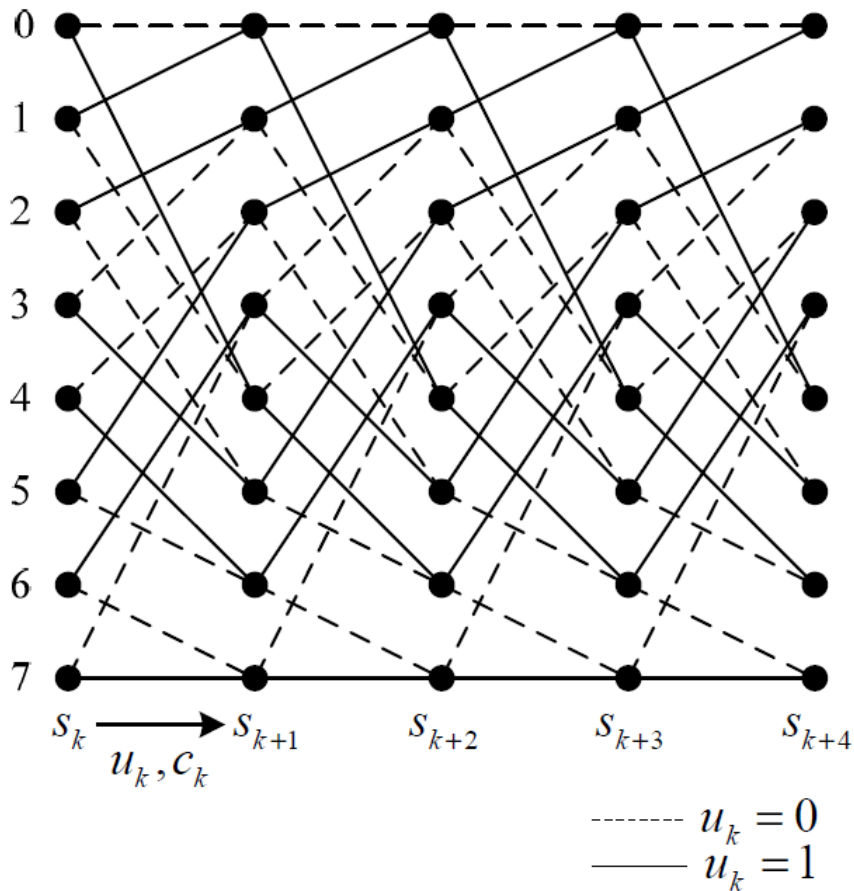
Ο αλγόριθμος αποκωδικοποίησης που χρησιμοποιείται στους αποκωδικοποιητές Turbo και στο συγκεκριμένο σύστημα είναι του μέγιστου εκ των υστέρων αλγοριθμος, ο MAP που έχει προταθεί από τον Bahl το 1974 και ονομάζεται επίσης αλγόριθμος BCJR. Η υψηλή πολυπλοκότητα και μεγάλη καθυστέρηση του αρχικού αλγορίθμου MAP έκανε τις υψηλών ταχυτήτων VLSI εφαρμογές, να είναι εξαιρετικά δύσκολο να υλοποιηθούν. Ευτυχώς, πολλές απλοποιήσεις έχουν εφαρμοστεί στον πρωτότυπο αλγόριθμο MAP προκειμένου να μειωθεί η πολυπλοκότητα στην εφαρμογή του. Ο όρος της αποκωδικοποίησης Turbo λειτουργικά απεικονίζεται στο Σχήμα 3. Όπως αναφέρθηκε προηγουμένως, η αποκωδικοποίηση βασίζεται στον αλγόριθμο MAP και υπολογίζεται συνήθως στον τομέα log (Robertson 1995) για να αποφευχθούν πολλαπλασιασμοί και διαιρέσεις. Κατά τη διαδικασία της αποκωδικοποίησης, κάθε αποκωδικοποιητής SISO (single input

single output ή soft input soft output, γενικά η τεχνολογίες που χρησιμοποιούν ένα στοιχείο εισόδου, ένα εξόδου ή τα συστήματα επικοινωνίας που χρησιμοποιούν ένα πομπό και έναν δεκτη) λαμβάνει τις εγγενείς αναλογίες λογαριθμικής πιθανοφάνειας, log-likelihood ratios (LLRs) από το κανάλι και τα εξωγενείς LLRs από τον άλλο συστατικό αποκωδικοποιητή SISO μέσω διεμπλοκής-αναδιάταξης (Π) ή αποδιεμπλοκής (Π-1). Ο κύριος ρόλος του Turbo εσωτερικού αναδιατάκτη είναι να δημιουργήσει μια αντιμετάθεση στην ακολουθία των δεδομένων εισόδου που είναι τόσο ασυσχέτιστη είναι δυνατόν. Η τυχειότητα του διεμπλοκέα δεν επηρεάζει μόνο την απόδοση αποκωδικοποίησης, αλλά οδηγεί επίσης σε καθυστέρηση της αποκωδικοποίησης, επειδή ο ένας αποκωδικοποιητής SISO πρέπει να περιμένει τον άλλο αποκωδικοποιητή SISO να τελειώσει την αποκωδικοποίηση του για να μπορέσει να ξεκινήσει την επόμενη επανάληψη.



Σχήμα 3. Βασική δομή κωδικοποιητή και αποκωδικοποιητή Turbo, όπου C είναι συνελκτικοί κωδικοποιητές και K κανάλι θορύβου.

Μια αποτελεσματική αναπαράσταση της διαδικασίας αποκωδικοποίησης Turbo είναι το χωροδικτύωμα το οποίο περιγράφει όλες τις πιθανές μεταβάσεις κατάστασης μέσω μίας γραφικής απεικόνισης. Το Σχήμα 4 δείχνει ένα τμήμα του χωροδικτυώματος για ένα δυαδικό κώδικα Turbo 8 σημείων, όπου οι διακεκομμένες ακμές αντιστοιχούν σε bit εισόδου $u_k = 0$, και οι συνεχείς ακμές αντιστοιχούν σε $u_k = 1$.



Σχήμα 4. Χωροδικτύωμα για έναν δυαδικό κώδικα Turbo 8 σημείων

Ο αλγόριθμος MAP είναι ένα αλγόριθμος αποκωδικοποίησης βέλτιστου σύμβολου που ελαχιστοποιεί την πιθανότητα σφάλματος κάποιου σύμβολο. Υπολογίζει τις εκ των υστέρων πιθανότητες (APPs) των bits πληροφοριών από την ληφθείσα αλληλουχία. Ο αλγόριθμος μπορεί να συνοψιστεί ως εξής:

$$L(\hat{u}_k) = \log \frac{P(u_k = +1 | y)}{P(u_k = -1 | y)} = \log \frac{\sum_{u_k=+1} P(s_{k-1} = s', s_k = s, y)}{\sum_{u_k=-1} P(s_{k-1} = s', s_k = s, y)}$$

Για να υπολογίσουμε το $(s_{k-1}=s', s_k=s, y)$, μπορούμε να χρησιμοποιήσουμε τον BCJR αλγόριθμο του Bahl:

$$P(s_{k-1} = s', s_k = s, y) = \alpha_{k-1}(s') \cdot \gamma_k(s', s) \cdot \beta_k(s),$$

όπου α_k και β_k αναφέρονται στις εμπρός πίσω μετρήσεις και υπολογίζονται ως εξής:

$$\alpha_k(s) = \sum_{s'} \gamma_k(s', s) \cdot \alpha_{k-1}(s')$$

$$\beta_k(s') = \sum_s \gamma_k(s', s) \cdot \beta_{k+1}(s)$$

Στις παραπάνω εξισώσεις, το γ είναι η πιθανότητα μεταβατικής κατάστασης και υπολογίζεται ως εξής:

$$\gamma_k(s', s) = P(s | s')P(y_k | s', s) = P(u_k)P(y_k | u_k)$$

$$= C_k \exp\left\{\frac{1}{2}u_k \left(L(u_k) + L_c y_k^s\right) + \frac{1}{2}L_c y_k^p x_k^p\right\},$$

όπου το C_k είναι μια σταθερά και δεν θα επηρεάσει τον υπολογισμό του $L(u_k)$. $L_c = 4E_s/N_0$. $L(u_k)$ είναι η αναλογία λογαριθμικής πιθανοφάνειας u_k που ορίζεται ως:

$$L(u_k) = \log \frac{P(u_k = +1)}{P(u_k = -1)}.$$

Έτσι, η αναλογία πιθανότητα εκ των υστέρων (APP) λογαριθμικής πιθανοφάνειας (LLR) των bits πληροφορίας μπορεί να εκφραστεί ως:

$$L(\hat{u}_k) = \log \frac{P(u_k = +1 | y)}{P(u_k = -1 | y)} = \log \frac{\sum_{u_k=+1} \alpha_{k-1}(s') \cdot \gamma_k(s', s) \cdot \beta_k(s)}{\sum_{u_k=-1} \alpha_{k-1}(s') \cdot \gamma_k(s', s) \cdot \beta_k(s)}.$$

4. Αρχιτεκτονική αποκωδικοποιητών Turbo πολλαπλών προτύπων

4.1 Σχετική έρευνα για ευέλικτες αρχιτεκτονικές Turbo αποκωδικοποιητών

Η ικανότητα για την υποστήριξη πολύτροπης Turbo αποκωδικοποίησης είναι απαραίτητη για έναν πολλαπλών χρήσεων φυσικών βαθμίδων (PHY) δέκτη. Δεδομένου ότι ορισμένα συστήματα 3G / 4G χρησιμοποιούν διαφορετικούς τύπους Turbo κωδικοποίησης (π.χ. δυαδικούς κώδικες CDMA, UMTS, HSDPA, και 3GPP LTE και διπλούς δυαδικούς κώδικες στο WiMax), μια γενική λύση για την υποστήριξη πολλαπλών τύπων κώδικα είναι να χρησιμοποιούμε προγραμματιζόμενους επεξεργαστές. Για παράδειγμα, ένας 2 Mbps Turbo αποκωδικοποιητής προτείνεται από τον Lin να υλοποιείται σε έναν επεξεργαστή DSP (2007). Επίσης, οι Shin & Park (2007) και Muller (2006) αναπτύξαν έναν αποκωδικοποιητή Turbo πολλαπλών χρήσεων που βασίζεται σε επεξεργαστές SIMD, όπου ένας ρυθμός δεδομένων 5,48 Mbps επιτυγχάνεται (Shin & Park) και ένα ρυθμός δεδομένων 100 Mbps επιτυγχάνεται σύμφωνα με τον Muller με κόστος 16 επεξεργαστές. Ενώ αυτοί οι προγραμματιζόμενοι επεξεργαστές SIMD/VLIW προσφέρουν μεγάλες δυνατότητες ευελιξίας, έχουν πολλά μειονεκτήματα, κυρίως υψηλότερη κατανάλωση ενέργειας και χαμηλότερη απόδοση από ό,τι οι λύσεις ASIC. Ένας αποκωδικοποιητής Turbo είναι συνήθως ένα από τα πλέον εντατικά υπολογιστικά τμήματα σε ένα δέκτη 4G.Ως εκ τούτου, είναι απαραίτητο να σχεδιάσουμε έναν, περιοχής και δύναμης, αποτελεσματικό και ευέλικτο αποκωδικοποιητή Turbo σε ASIC.

Λόγω των πολλών ομοιοτήτων μεταξύ των διαφόρων συστημάτων κωδικοποίησης Turbo που χρησιμοποιούνται στα ασύρματα πρότυπα τηλεπικοινωνιών 4G, θα παρουσιάσουμε μια ρυθμιζόμενη αρχιτεκτονική VLSI για πολλαπλών προτύπων Turbo αποκωδικοποιήσεις. Αυτή η αρχιτεκτονική μπορεί να ρυθμιστεί ώστε να υποστηρίξει και τους απλούς και τους διπλούς δυαδικούς κώδικες Turbo με έως και οκτώ στοιχεία. Το πρόβλημα σύγκρουσης μνήμης αντιμετωπίζεται με την εφαρμογή μη δεσμευμένων αντιμεταθετικών παράλληλων διεμπλοκέων. Στον αλγόριθμο MAP, ο αποκωδικοποιητής, η δομή της μνήμης και η δρομολόγηση του δικτύου έχουν σχεδιαστεί για να λειτουργούν σε πλήρη ταχύτητα με τον παράλληλο διεμπλοκέα. Η

προτεινόμενη αρχιτεκτονική συναντά την πρόκληση της πολλαπλών προτύπων Turbo αποκωδικοποίησης σε πολύ υψηλούς ρυθμούς δεδομένων.

4.1.1 Αποκωδικοποίηση των Δυαδικών Κωδικών Turbo στο τομέα καταγραφής

Ο αρχικός αλγόριθμος MAP είναι υπερβολικά πολύπλοκος για εφαρμογή σε ένα πρακτικό σύστημα. Για να αποφευχθούν οι περίπλοκοι πολλαπλασιασμοί και να λυθούν τα προβλήματα αριθμητικής αστάθειας, μπορεί κανείς να υπολογίσει το MAP αλγόριθμο στον τομέα καταγραφής. Για να εξηγήσουμε τον αλγόριθμο αποκωδικοποίησης Log-MAP, πρέπει πρώτα να εισαγάγουμε τη λειτουργία \max^* η οποία ορίζεται ως:

$$\max^*(a, b) = \log \{e^a + e^b\} = \max(a, b) + \log(1 + e^{-|a-b|}).$$

Ας εξετάσουμε τη διαδικασία αποκωδικοποίησης ενός απλού δυαδικού κώδικα Turbo, όπου s_k είναι η κατάσταση του χωροδικτυώματος στο χρόνο k , τότε ο αποκωδικοποιητής MAP υπολογίζει τον LLR των υστέρων πιθανότητα (APP) κάθε bit πληροφοριών u_k κατα

$$\Lambda(\hat{u}_k) = \max_{u_k=1}^* \{ \alpha_{k-1}(s_{k-1}) + \gamma_k(s_{k-1}, s_k) + \beta_k(s_k) \} - \max_{u_k=0}^* \{ \alpha_{k-1}(s_{k-1}) + \gamma_k(s_{k-1}, s_k) + \beta_k(s_k) \},$$

Όπου α_k και β_k αναφέρονται στις εμπρός πίσω μετρήσεις και υπολογίζονται ως εξής:

$$\alpha_k(s_k) = \max_{s_{k-1}}^* \{ \alpha_{k-1}(s_{k-1}) + \gamma_k(s_{k-1}, s_k) \},$$

$$\beta_k(s_k) = \max_{s_{k+1}}^* \{ \beta_{k+1}(s_{k+1}) + \gamma_k(s_k, s_{k+1}) \},$$

όπου γ_k είναι η διακλαδική μεταβατική πιθανότητα που εισήχθη νωρίτερα και συνήθως αναφέρεται ως μια διακλάδωση μετρικού συστήματος (BM). Για να εξαγάγουμε τις εξωγενείς πληροφορίες, ο $\Lambda(u_k)$ μπορεί να χωριστεί σε τρεις περιόδους όρους: τον εξωγενή LLR $L_e(u_k)$, τον εκ των προτέρων LLR $L_a(u_k)$ και τον συστηματικό αλγόριθμο αναλογίας πιθανοφάνειας $L_c(u_k^s)$:

$$\Lambda(\hat{u}_k) = L_e(u_k) + L_a(u_k) + L_c(y_k^s).$$

4.1.2 Αποκωδικοποίηση μέσω μονοεπίπεδου προβλέψιμου μετασχηματισμού (τετραγωνική ρίζα)

Για τους δυαδικούς κώδικες Turbo, οι κύκλοι χωροδικτυωμάτων μπορούν να μειωθούν κατά 50% με την εφαρμογή ενός μονοεπίπεδου προβλέψιμου όπως απεικονίζεται στο Σχήμα 5. Επειδή δύο στάδια του χωροδικτυώματος μπορούν να υποστούν επεξεργασία σε κάθε χρονικό βήμα, αυτή η διαδικασία αναφέρεται ως μετασχηματισμός ρίζας στην τετάρτη. για παράδειγμα, η αναδρομή της τετραγωνικής ρίζας μπορεί να εκφραστεί ως:

$$\alpha_k(s_k) = \max_{s_{k-1}}^* \left\{ \max_{s_{k-2}}^* \{ \alpha_{k-2}(s_{k-2}) + \gamma_{k-1}(s_{k-2}, s_{k-1}) \} + \gamma_k(s_{k-1}, s_k) \right\}$$

$$= \max_{s_{k-2}, s_{k-1}}^* \{ \alpha_{k-2}(s_{k-2}) + \gamma_k(s_{k-2}, s_k) \},$$

όπου $\gamma_k(s_{k-2}, s_k)$ είναι η συγχωνευμένη μετρική διακλάδωση για το διάνυσμα δύο-bit $\{u_{k-1}, u_k\}$ που συνδέει τα στοιχεία S_{k-2} και S_k :

$$\gamma_k(s_{k-2}, s_k) = \gamma_{k-1}(s_{k-2}, s_{k-1}) + \gamma_k(s_{k-1}, s_k).$$

Παρομοίως, ο μετασχηματισμός τετραγωνικής ρίζας μπορεί να εφαρμοστεί στη β αναδρομή:

$$\beta_k(s_k) = \max_{s_{k+2}, s_{k+1}}^* \{ \beta_{k+2}(s_{k+2}) + \gamma_k(s_k, s_{k+2}) \}.$$

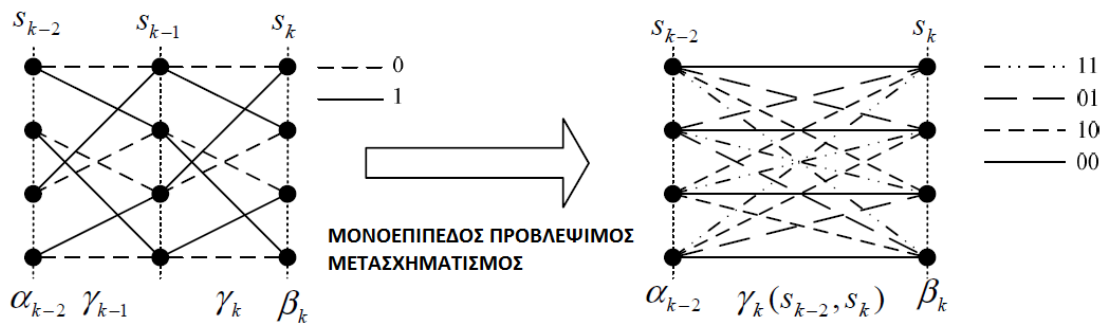
Επειδή αυτός ο αλγόριθμος τετραγωνικής ρίζας βασίζεται στο επίπεδο συμβόλων, πρέπει να καθορίσουμε την αξιοπιστία του συμβόλου, όπως:

$$L(\varphi_{ij}) = \max_{s_{k-2}, s_k}^* \{ \alpha_{k-2}(s_{k-2}) + \gamma_k^{ij} + \beta_k(s_k) \},$$

όπου το γ_k^{ij} είναι το πιθανότητα διακλαδωτικής μετάβασης του συμβόλου με $u_{k-1}=i$ και $u_k=j$. Μετά που μαθαίνουμε τις πιθανοτήτων του συμβόλου, τα LLRs bit μπορούν να υπολογιστούν ως εξής:

$$\Lambda(\hat{u}_{k-1}) = \max_{s_{k-2}, s_k}^* \{ L(\varphi_{10}), L(\varphi_{11}) \} - \max_{s_{k-2}, s_k}^* \{ L(\varphi_{00}), L(\varphi_{01}) \}$$

$$\Lambda(\hat{u}_k) = \max_{s_{k-2}, s_k}^* \{ L(\varphi_{01}), L(\varphi_{11}) \} - \max_{s_{k-2}, s_k}^* \{ L(\varphi_{00}), L(\varphi_{10}) \}.$$



Σχήμα 5. Ένα παράδειγμα ενός μονοεπίπεδου προβλέψιμου μετασχηματισμού ενός χωροδικτύωματος 4 στοιχείων

4.1.3 Αποκωδικοποίηση διπλών δυαδικών κωδικών Turbo (τετραγωνική ρίζα)

Οι διπλοί δυαδικοί κώδικες Turbo υιοθετήθηκαν στο IEEE 802.16e πρότυπο WiMax εξαιτίας της καλύτερης απόδοσης διόρθωσης σφάλματος από τους απλούς δυαδικούς κώδικες. Για τους διπλούς δυαδικούς κώδικες, η κύρια διαφορά με τους απλούς είναι το σύστημα τερματισμού του χωροδικτύωματος και το συμβολικό καθεστώς αποκωδικοποίησης του. Το διπλό δυαδικό χωροδικτύωμα είναι κλειστό σαν κύκλος με την αρχική κατάσταση να ισούται με την τελική κατάσταση. Αυτό επίσης αναφέρεται ως ένα σύστημα τερματισμού των Bits της ουράς, όπως φαίνεται και στο Σχήμα 6. Ο συμβολικός MAP αλγόριθμος εφαρμόζεται με μια αριστερόστροφη διαδικασία για α μετρική ενημέρωση στοιχείου και δεξιόστροφα για την β μετρική ενημέρωση στοιχείου. Όπως φαίνεται στο Σχήμα 6, οι τέσσερις μεταβατικές διακλαδώσεις συνδέονται με κάθε μετρική ενημέρωση των α / β στοιχείων. Ο αλγόριθμος αποκωδικοποίησης για τους διπλούς δυαδικούς κώδικες είναι εγγενώς με βάση τον αλγόριθμο τετραγωνικής ρίζας, ως εκ τούτου η ίδια τετραγωνική ρίζα (Radix-4) οι α, β και L(φ) λειτουργικές μονάδες που χρησιμοποιούνται στους δυαδικούς κώδικες μπορούν να εφαρμοστούν και στους

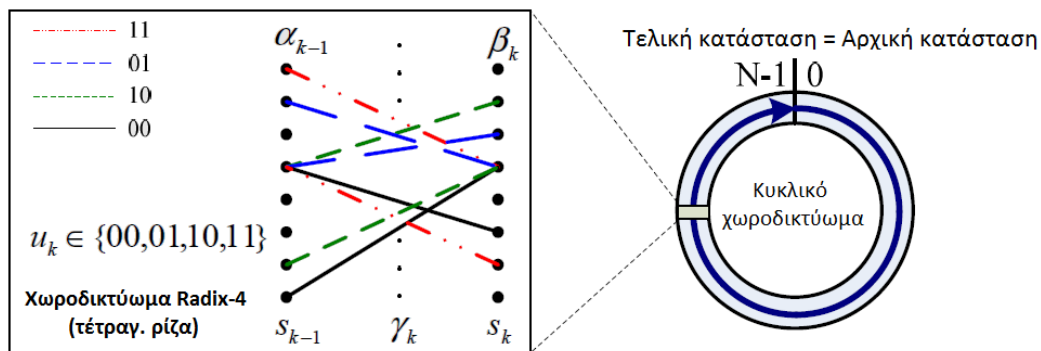
διπλούς δυαδικούς κώδικες με έναν απλό άμεσο τρόπο. Τα μόνα διαφορετικά μέρη είναι οι υπολογισμοί του μετρικού κλάδου γ^j

και το σύστημα τερματισμού χωροδικτύωματος των bit ουράς . Τρεις αλγόριθμοι LLRs πρέπει να υπολογίζονται για τους διπλούς δυαδικούς κώδικες:

$$\Lambda^1(\hat{u}_k) = L_k(\varphi_{01}) - L_k(\varphi_{00})$$

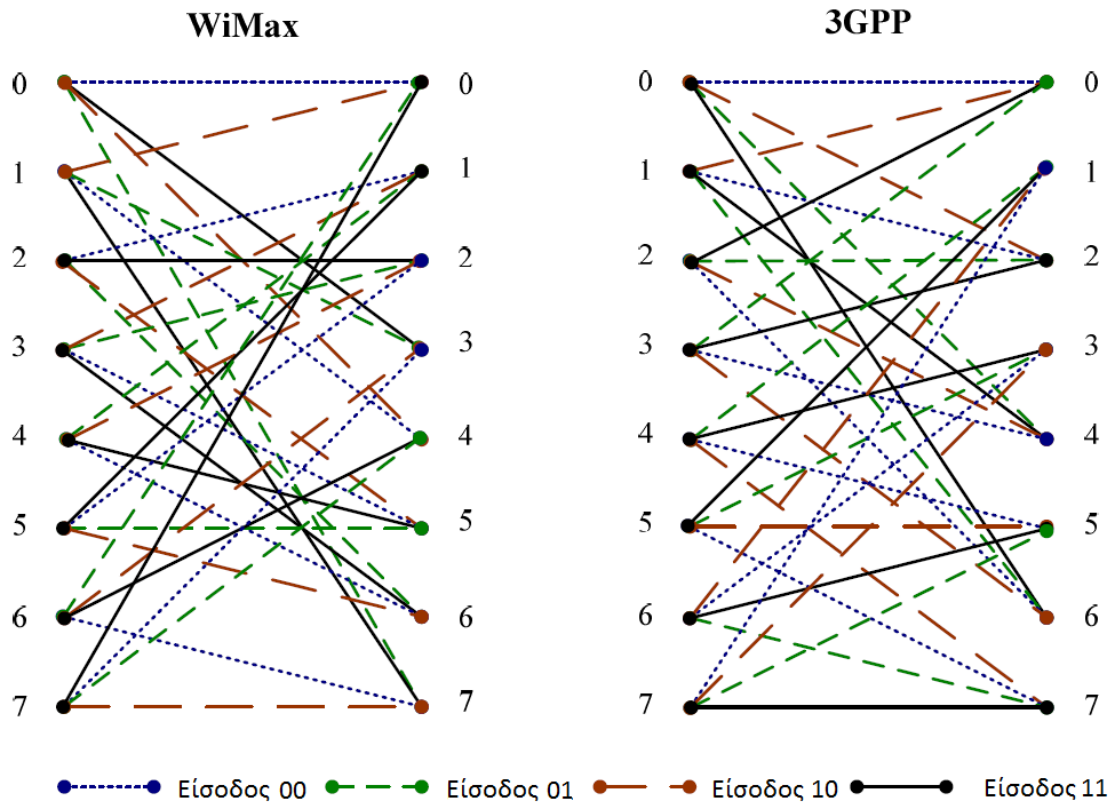
$$\Lambda^2(\hat{u}_k) = L_k(\varphi_{10}) - L_k(\varphi_{00})$$

$$\Lambda^3(\hat{u}_k) = L_k(\varphi_{11}) - L_k(\varphi_{00}).$$



Σχήμα 6. Κυκλικό χωροδικτύωμα για διπλό δυαδικό κώδικα Turbo

Παρακάτω, το σχήμα 7 συγκρίνει τις δομές χωροδικτυωμάτων (στη μορφή τετραγωνικής ρίζας-Radix 4) του IEEE 802.16e WiMAX και του 3GPP LTE προτύπου τηλεπικοινωνιών. Όπως μπορεί να φανεί, 25% της δομής του χωροδικτύωματος είναι πανομοιότυπο. Οι ομοιότητες μεταξύ αυτών των δύο χωροδικτυωμάτων σημαίνει ότι ένας γενικότερος αποκωδικοποιητής Turbo μπορεί να σχεδιαστεί έτσι ώστε να υποστηρίζει πολλαπλά ασύρματα πρότυπα με χαμηλή επιβάρυνση του υλικού(hardware).



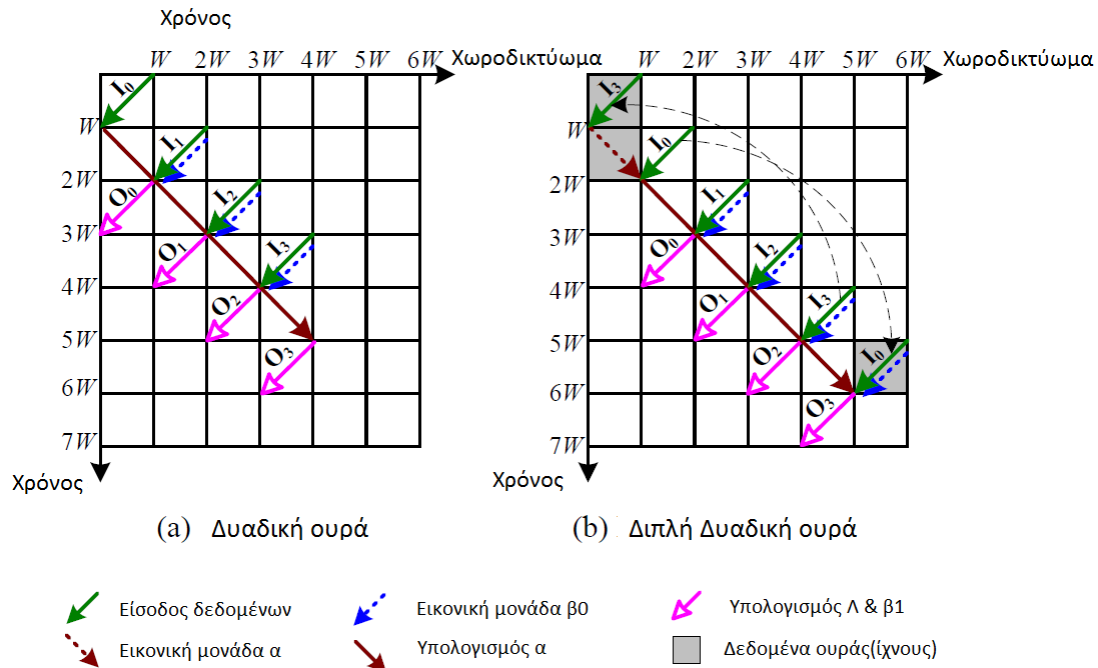
Σχήμα 7. Δομές χωροδικτυωμάτων των IEEE 802.16e WiMax και LTE 3GPP Κωδικών Turbo

4.1.4 Ενιαία Αρχιτεκτονική του Αποκωδικοποιητή Log-MAP

Με βάση την παρατήρηση ότι και οι δυαδικοί και οι διπλοί δυαδικοί κώδικες μπορούν να αποκωδικοποιηθούν σε ένα ενιαίο τρόπο, εισάγουμε έναν αποκωδικοποιητή Log-MAP με ευέλικτη αρχιτεκτονική Radix-4 για την υποστήριξη και των δύο λειτουργικών τύπων αποκωδικοποίησης. Για την αποτελεσματική υλοποίηση του αλγορίθμου Log-MAP στο υλικό (hardware), υιοθετείται η τεχνική συρόμενου παράθυρου (sliding window technique).

Οι δύο τύποι των λειτουργιών αποκωδικοποίησης μπορούν να γενικευθούν σε μία ενιαία ροή η οποία παρουσιάζεται στο Σχήμα 8. Ας χρησιμοποιήσουμε πρώτα τους δυαδικούς κώδικες Turbo ως παράδειγμα για να εξηγήσουμε τη διαδικασία αποκωδικοποίησης. Στο Σχήμα 8 (α), ας υποθέσουμε ότι μία ακολουθία δεδομένων χωρίζεται σε συρόμενα μπλοκ με ένα συρόμενο παράθυρο μήκους W . Κατά το πρώτο περιθώριο χρόνου, το πρώτο συρόμενο μπλοκ I_0 τροφοδοτείται μέσα στον

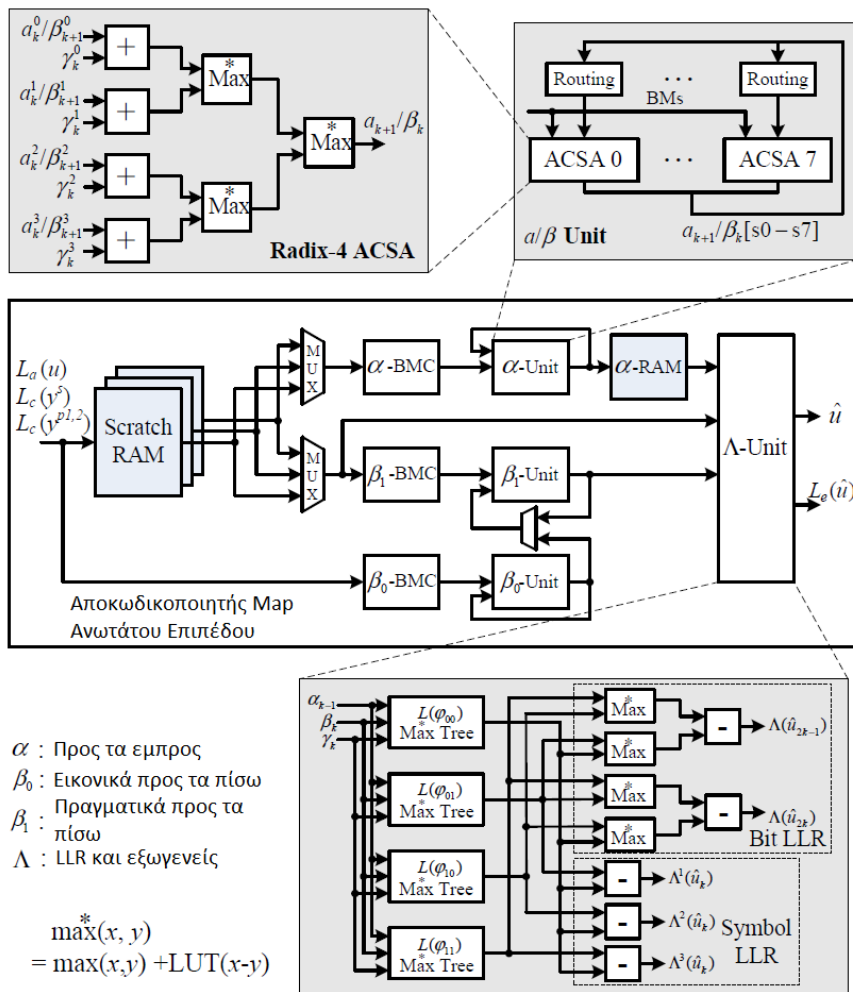
αποκωδικοποιητή και αποθηκεύεται σε προσωρινή μνήμη RAM μηδέν 0. Κατά τη δεύτερη χρονοθυρίδα, το συρόμενο μπλοκ I_1 αποθηκεύεται σε προσωρινή μνήμη RAM 1-αναδρομή ομοιώματος β_0 εκτελείται στο συρόμενο μπλοκ I_1 - συγχρόνως α αναδρομή εκτελείται στο συρόμενο μπλοκ I_0 και τα αποτελέσματα αποθηκεύονται σε α -RAM (LIFO). Ξεκινώντας από την τρίτη χρονοθυρίδα, α μονάδα (που εργάζονται στη ροή προς τα εμπρός), εικονική μονάδα β_0 (στην αντίστροφη ροή), αποτελεσματική μονάδα β_1 (στην αντίστροφη ροή) και μονάδα Λ (στην αντίστροφη ροή) λειτουργούν παράλληλα για να παρέχουν αποκωδικοποίηση σε πραγματικό χρόνο, με καθυστέρηση $2W$. Αυτή η λειτουργία αποκωδικοποίησης βασίζεται σε τρεις μονάδες αναδρομής, δύο χρησιμοποιούνται για τις αναδρομές προς τα πίσω (εικονική β_0 και αποτελεσματική β_1), και μία για αναδρομή προς τα εμπρός (α). Κάθε αναδρομική μονάδα περιέχει πλήρη παράλληλους ACSA (Add-Compare-Select-Add) διαχειριστές. Για να μειωθεί η καθυστέρηση της αποκωδικοποίησης, δεδομένα σε ένα συρόμενο παράθυρο τροφοδοτούνται μέσα στον αποκωδικοποιητή με την αντίστροφη ροή- μονάδα α εργάζεται στην ροή προς τα εμπρός – Εικονική μονάδα β_0 και αποτελεσματικές μονάδες β_1 και Λ , λειτουργούν σε αντίστροφη ροή, όπως παρουσιάζεται στο Σχήμα 8. Αυτό οδηγεί σε μια καθυστέρηση αποκωδικοποίησης βαθμού $2W$ για τους δυαδικούς κώδικες και $3W$ για τους διπλούς δυαδικούς κώδικες. Οι διπλοί δυαδικοί κώδικες έχουν ένα επιπλέον W καθυστέρησης επειδή μια πρόσθετο απόκτημα είναι απαραίτητο για να ξανα λάβουμε τις αρχικές μετρικές καταστάσεις α .



Σχήμα 8. Διάγραμμα τετραγώνων συρόμενου παράθυρου για απλούς και διπλούς δυαδικούς κώδικες Turbo

Το Σχήμα 9 παρακάτω δείχνει ένα πολύτροπο Radix-4 (τετραγωνικής ρίζας) Log-MAP αποκωδικοποιητή με αρχιτεκτονική ASIC. Τρεις προσωρινές RAM (με βάθος W) ήταν αναγκαίες για να ρυθμίσουν τους συστηματικής εισόδου, ισοτιμίας και εκ των προτέρων (a priori) αλγόριθμους LLRs. Τρεις μετρικές κλαδικές υπολογιστικές μονάδες του μετρικού συστήματος (branch metric calculation - BMC) χρησιμοποιούνται για τον υπολογισμό των μετρικών στοιχείων κλάδου για τις λειτουργικές μονάδες α , β_0 και β_1 . Για την υποστήριξη πολλαπλών κωδικών Turbo, ο αποκωδικοποιητής χρησιμοποιεί ρυθμιζόμενες BMCs και ρυθμιζόμενες μονάδες λειτουργίας α και β που μπορούν να υποστηρίξουν πολλαπλές λειτουργίες μεταφοράς διαμορφώνοντας τα μπλοκ δρομολόγησης. Το μπλοκ δρομολόγησης μπορεί να ρυθμιστεί ώστε να υποστηρίζει διαφορετικά πολώνυμα κωδικοποιητή. Κάθε μονάδα α και β αποτελείται από πλήρως παράλληλες μονάδες ACSA, ώστε η αρχιτεκτονική να μπορεί να υποστηρίξει μέχρι και αποκωδικοποίηση Turbo 8 καταστάσεων. Η τετραγωνικής ρίζας (Radix 4) μονάδα ACSA υλοποιείται με τέσσερις παράλληλες αθροιστές που ακολουθούνται από τρεις μονάδες max*. Για να δημιουργηθούν αλγόριθμοι LLRs τόσο για τους δυαδικούς όσο και για τους διπλούς δυαδικούς κώδικες, η εξωγενής μονάδα λειτουργίας Λ υλοποιεί τόσο αλγόριθμους LLR ψηφίων(bit) όσο και αλγόριθμους παραγωγής συμβόλων. Για να αποθηκεύσουμε τον

συλλογισμό, τέσσερις \max * διακλαδώσεις διαχωρίστηκαν και από των δύο ειδών τις λειτουργίες. Η μονάδα Λ μπορεί να παράγει soft bit αλγόριθμους LLRs και συμβολικούς LLRs σε πραγματικό χρόνο με μία σταθερό καθυστέρηση των $2W$ για δυαδικούς κώδικες ή $3W$ για τους διπλούς δυαδικούς κώδικες.



Σχήμα 9. Αρχιτεκτονική Ενιαίου Αποκωδικοποιητή Log-MAP

Σε αυτή την αρχιτεκτονική, πολλά μπλοκ μπορεί να μοιραστούν μεταξύ των δύο λειτουργιών αποκωδικοποίησης. Για παράδειγμα, οι μονάδες α , β και $L(\varphi)$ οι α RAMs, και προσωρινές μνήμες RAMs (scratch RAMs) μπορούν εύκολα να μοιραστούν μεταξύ δύο λειτουργιών αποκωδικοποίησης. Ο πίνακας 1 συγκρίνει τη χρήση των πόρων για την αρχιτεκτονική αποκωδικοποιητή πολλαπλών λειτουργιών (multi-mode) με την αρχιτεκτονική αποκωδικοποιητή μίας

λειτουργίας(single-mode). Στον πίνακα 1, το M είναι ο αριθμός των καταστάσεων του χωροδικτύωματος, το W είναι το μήκος συρόμενου παραθύρου, τα B_m , B_b , B_c και B_e είναι οι διευκρινίσεις των μετρικών καταστάσεων, μετρικών διακλαδώσεων, αλγόριθμων LLRs καναλιού και εξωγενείς LLRs, αντίστοιχα. Από τον πίνακα 1, μπορούμε να δούμε ότι η επιβάρυνση για την προσθήκη ευελιξίας είναι πολύ μικρή, το ποσοστό της οποίας είναι μόνο περίπου 7%. Αυτή η επιβάρυνση προέρχεται κυρίως από τους πολυπλέκτες που χρησιμοποιήθηκαν στην δρομολόγηση των δικτύων(routing) στις αναδρομικές λειτουργικές μονάδες. Ο συγκεκριμένος πίνακας μας δείχνει επίσης ότι μια ρυθμιζόμενη VLSI αρχιτεκτονική είναι μια πολλά υποσχόμενη λύση για τον αποκωδικοποιητή Turbo πολλαπλών προτύπων. Ως απόδειξη όλων αυτών, ο αποκωδικοποιητής έχει συσταθεί για μια τεχνολογία CMOS 65nm. Ο πίνακας 2 συνοψίζει τα αποτελέσματα της σύνθεσης σε ένα ρολόι συχνότητας 200MHz, η οποία είναι μια τυπική ταχύτητα ρολογιού σε ένα σχεδιασμό ASIC σύμφωνα με τον Yang Sun όπως τα παρουσίασε σε συνέδριο συστημάτων,αρχιτεκτονικών και εφαρμογών το 2008.

Πίνακας 1. Σύγκριση πολυπλοκότητας

	Multi-mode	Single-mode
Storage (bits)	$(9B_e+12B_c+MB_m)W$	$(9B_e+12B_c+MB_m)W$
B_m -bit max*	$(25/2)M+4$	$(25/2)M$
1-bit adder	$16MB_m+10MB_b$	$16MB_m+10MB_b$
1-bit flip-flop	$5MB_m+2MB_b$	$5MB_m+2MB_b$
1-bit mux	$16MB_m+16MB_b$	$3MB_m$
Normalized area	1.0	0.93

1 τεσσάρων εισόδων max4* υπολογίζεται ως 3 δύο εισόδων max*

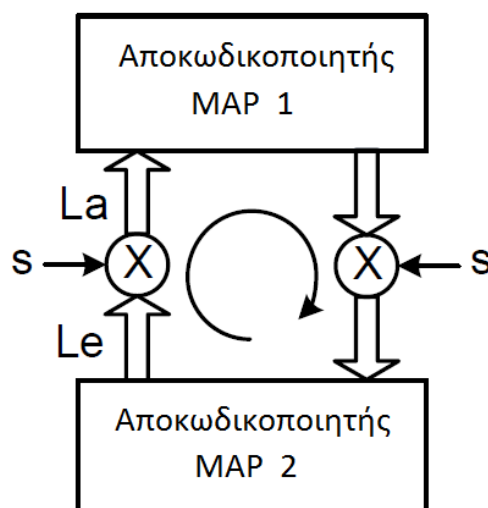
1 οχτώ εισόδων max8* υπολογίζεται ως 7 δύο εισόδων max*

Πίνακας 2. Κατανομή Εύρους

Blocks	Gate count
α unit (including α BMU)	30.8K gates
β unit x 2 (including β BMUs)	66.2K gates
Λ unit	37.3K gates
α RAM	2560 bits
Scratch RAMs x 3	4224 bits
Control logic	13.4K gates

4.1.5 Βελτιστοποίηση Χώρου

Αν και ο αποκωδικοποιητής MAP πολλαπλών λειτουργιών (multi-mode) σχεδιάστηκε με περιορισμένο προϋπολογισμό, ερευνήσαμε πρόσθετες τεχνικές για να επιτευχθεί περαιτέρω εξοικονόμηση ζώνης. Όταν ο πυρήνας της MAP αποκωδικοποίησης έχει σχεδιαστεί, υπάρχουν αρκετές επιλογές για την εφαρμογή της $\log(e^a + e^b)$ λειτουργίας. Υποθέτουμε δύο επιλογές: Log-MAP όπου $\log(e^a + e^b) \sim \max(a, b) + C$, όπου C είναι ένας συντελεστής διόρθωσης, και Max-Log-MAP, όπου $\log(e^a + e^b) \sim \max(a, b)$. Εάν χρησιμοποιήσουμε max-Log-MAP, η απώλεια απόδοσης σε σύγκριση με το Log-MAP είναι περίπου 0,3 dB με εξοικονόμηση περίπου 15% λογικής περιοχής. Για να μειωθεί το χάσμα των επιδόσεων με τον Log-MAP αποκωδικοποιητή, εισαγάγαμε ένα παράγοντα κλιμάκωσης που εφαρμόζεται στις εξωγενείς τιμές LLR όπως φαίνεται στο Σχήμα 10 με αποτέλεσμα την απώλεια 0,1 dB, με εξοικονόμηση 15% λογικής περιοχής. Με αυτή τη βελτιστοποίηση για τον multi-mode MAP αποκωδικοποιητή που χρησιμοποιεί max-Log-MAP, η ζώνη του πυριτίου είναι συγκρίσιμη με τον Log-MAP single-mode MAP αποκωδικοποιητή.



Σχήμα 10. Μέθοδος κλιμάκωσης εξωγενούς λογαριθμικής πιθανοφάνειας (LLR)

4.2 Εσωτερική αρχιτεκτονική διεμπλοκία Turbo

Η διεμπλοκή ή αλλιώς αναδιάταξη (interleaving) έχει χρησιμοποιηθεί συχνά σε μία ποικιλία συστημάτων επικοινωνίας. Γενικά, ένας διεμπλοκίας χρησιμοποιούνται για την τυχαία τοποθέτηση των θέσεων με σφάλμα για την καταπολέμηση της εξασθένιση ή του ξεσπάσματος καναλιών σφαλμάτων. Ο εσωτερικός διεμπλοκίας Turbo είναι μια συσκευή που λαμβάνει ακολουθία εισόδου δυαδικών ψηφίων (bits) και παράγει μια ακολουθία εξόδου που είναι όσο το δυνατόν ασυσχέτιστη γίνεται. Δεδομένου ότι αυτή η τυχαιότητα επηρεάζει άμεσα την απόδοση της αποκωδικοποίησης, η καλύτερη επιλογή θα είναι ο τυχαίος διεμπλοκίας. Ωστόσο, ο τυχαίος αυτός αναδιατάκτης δεν είναι μόνο δύσκολο να εφαρμοστεί, αλλά επίσης είναι ένα εμπόδιο στην παράλληλη αποκωδικοποίηση Turbo εξαιτίας του προβλήματος σύγκρουσης στην πρόσβαση της μνήμης. Συνεπώς, η αναζήτηση για έναν δομημένο διεμπλοκίας, ειδικά έναν αναδιατάκτη κοινούς αποδοχής, παραμένει ένα ενεργό θέμα στην ερευνητική κοινότητα που ασχολείται με την κωδικοποίηση στις Τηλεπικοινωνίες και όχι μόνο. Παραδοσιακά, οι συγκρούσεις μνήμης οι οποίες συμβαίνουν λόγω διεμπλοκίας, λύνονται με την προσθήκη διαφόρων προσωρινών μνημών εγγραφής. Πρόσφατα, νέοι μη αμφισβητούμενοι (Contention-Free) διεμπλοκείς έχουν υιοθετηθεί για τα ασύρματα πρότυπα επόμενης γενιάς, όπως ο τετραγωνικός πολυώνυμικός μεταθετικός (quadratic polynomial permutation - QPP) διεμπλοκίας στο πρότυπο 3GPP LTE και ο σχεδόν τακτικής μετάθεσης (almost regular permutation - ARP) διεμπλοκίας στο IEEE 802.16e πρότυπο WiMax σύμφωνα με τους Y.Sun & Takeshita και τον Berrou αντίστοιχα.

4.2.1 Contention-free Διεμπλοκείς

Σύμφωνα με τον Nimbalker, ένας διεμπλοκίας $\pi(i)$, $0 \leq i < K$, λέγεται ότι είναι contention-free (άνευ διαμαχης/αμφισβήτησης) για ένα μέγεθος παραθύρου W , αν και μόνο αν

πληρεί τις εξής προϋποθέσεις τόσο για $\psi = \pi$ (διεμπλοκέας) και $\psi = \pi^{-1}$ (αποδιεμπλοκέας)

$$\left\lfloor \frac{\psi(j + tW)}{W} \right\rfloor \neq \left\lfloor \frac{\psi(j + vW)}{W} \right\rfloor$$

Όπου $0 \leq j < W$, $0 \leq t, v < M (=K/W)$ και t διάφορο του v . Οι όροι στην παραπάνω εξίσωση είναι ουσιαστικά οι δείκτες μνήμης (memory indices) που είναι ταυτόχρονα προσβάσιμοι από τους επεξεργαστές M . Εάν αυτές οι διευθύνσεις μνήμης είναι όλες μοναδικές κατά τη διάρκεια κάθε διαδικασίας ανάγνωσης και εγγραφής (read and write operation), τότε δεν υπάρχουν “διαμάχες” όσον αφορά την πρόσβαση στη μνήμη.

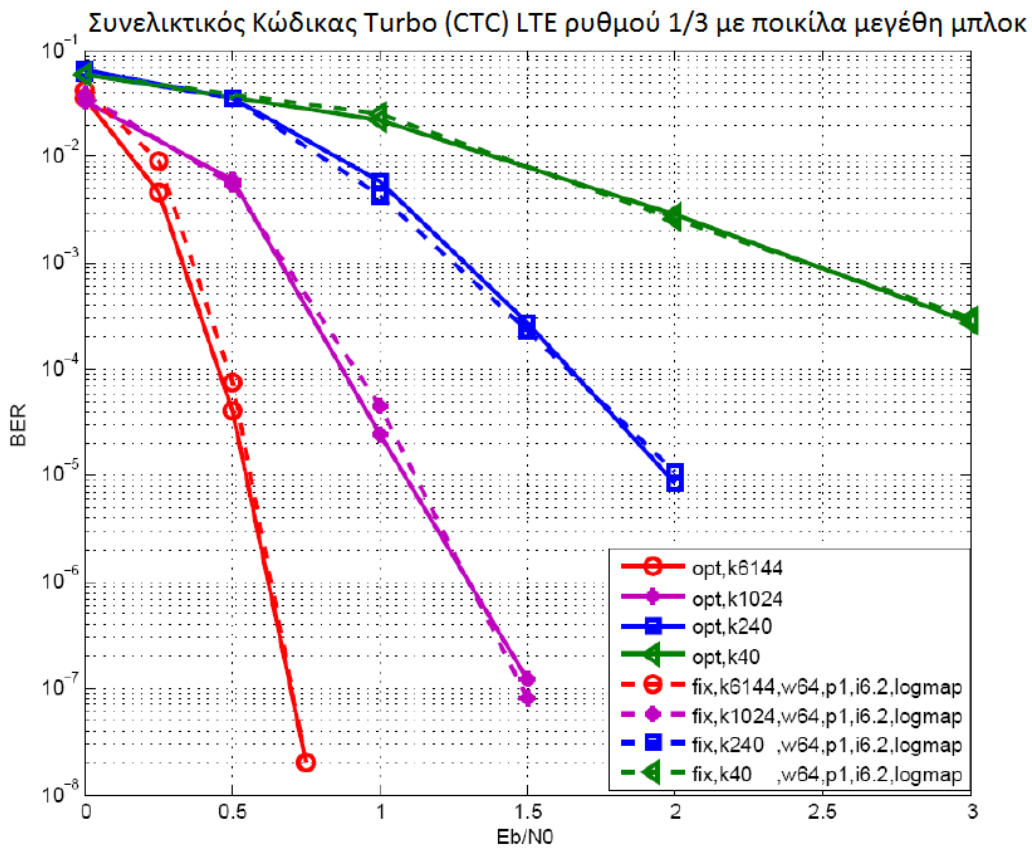
4.2.2 Τετραγωνικός πολυωνυμικός μεταθετικός (Quadratic Polynomial Permutation - QPP) διεμπλοκέας στο πρότυπο 3GPP LTE

Σύμφωνα με την επιστημονικά αποδεχόμενη (Ασύρματα Τηλεπικοινωνιακά Πρότυπα, 2008) εξίσωση

$$\Pi(x) = (f_2 x^2 + f_1 x) \bmod N,$$

και λαμβάνοντας υπόψη ένα μπλοκ πληροφοριών μήκους N , η x -οστή αναδιεταγμένη θέση εξόδου όπου όπου f_1 και f_2 είναι ακέραιοι και εξαρτώνται από το μπλοκ μήκους N ($0 \leq x, f_1, f_2 < N$). Το μπλοκ μήκους N έχει καθοριστεί στο πρότυπο 3GPP LTE και κυμαίνεται από 40 έως 6144. Το Σχήμα 11 απεικονίζει το αποτέλεσμα της προσομοίωσης του ποσοστό σφάλματος δυαδικού ψηφίου (bit error rate – BER) του κώδικα Turbo 3GPP LTE για μπλοκ μήκους 40, 240, 1024, και 6144. Στο Σχήμα 11,

τόσο μη σταθερά (floating) που αντιστοιχεί στις τιμές “opt” στο σχήμα, όσο και σταθερά (fixed) σημεία απεικονίζονται στα αποτελέσματα της προσομοίωσης. Οι παράμετροι που χρησιμοποιήθηκαν στα σταθερά σημεία της προσομοίωσης έχουν ως εξής: ακρίβεια δυαδικών ψηφίων εισόδου καναλιού LLR = 6 (με 2 ψυφία κλασματικού δυαδικού αριθμού), ακρίβεια δυαδικών ψηφίων των εσωτερικών μετρικών = 12, μήκος παραθύρου ολίσθησης = 64, παράλληλα συρόμενο παράθυρο = 1, αλγόριθμος αποκωδικοποίησης MAP = τεσσάρων (4) εισόδων πίνακας αναζήτησης βασισμένος Log-MAP, και μέγιστη επανάληψη = 6.



Σχήμα 11. Αποτελέσματα προσομοίωσης κωδικών Turbo σε πρότυπο 3GPP LTE για σταθερό και μη σταθερό σημείο

4.2.3 Πρακτική εφαρμογή του Τετραγωνικού Πολυωνομικού Μεταθετικού (Quadratic Polynomial Permutation - QPP) διεμπλοκέα

Ο άμεσος υπολογισμός της διεμπλοκής QPP είναι δύσκολος λόγω των λειτουργιών πολλαπλασιασμού και ως προς μέτρο (modulo). Μια πιο αποτελεσματική μέθοδος παραγωγής διευθύνσεων είναι να υπολογίσουμε το $\Pi(x)$ αναδρομικά:

$$\Pi(x+1) = ((f_2x^2 + f_1x) + (2f_2x + f_1 + f_2)) \bmod N = (\Pi(x) + \Gamma(x)) \bmod N ,$$

όπου

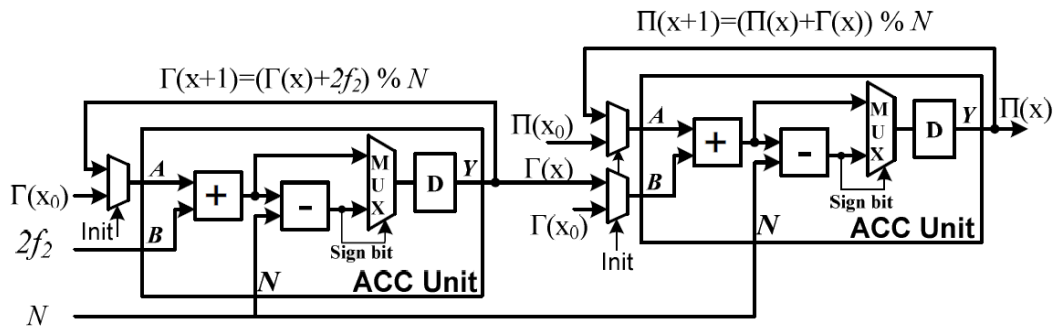
$$\Gamma(x) = (2f_2x + f_1 + f_2) \bmod N ,$$

και το $\Gamma(x)$ μπορεί επίσης να υπολογιστεί αναδρομικά ως εξής:

$$\Gamma(x+1) = (\Gamma(x) + 2f_2) \bmod N .$$

Δεδομένου ότι τα $\Pi(x)$, $\Gamma(x)$ και $2f_2$ είναι όλα μικρότερα από το N , η λειτουργία ως προς μέτρο μπορεί να εφαρμοστεί αποτελεσματικά με αθροιστές και πολυπλέκτες. Για την πρακτική υλοποίηση του διεμπλοκέα QPP στο υλικό (hardware), έχουμε εισαγάγει ένα κύκλωμα παραγωγής διευθύνσεων μέσω υπερχείλισης δύο μονάδων πρόσθεσης σύγκρισης και επιλογής (Add-Compare-Choose - ACC) , όπως παρουσιάζεται στο Σχήμα 12. Όπως μπορεί να φανεί, κανένας πολλαπλασιαστής και διαιρέτης δεν απαιτείται σε αυτή την αρχιτεκτονική. Η κρίσιμη διαδρομή του κυκλώματος περιέχει μόνο δύο αθροιστές και δύο πολυπλέκτες. Μετά τη ρύθμιση μιας αρχικής τιμής για $\Pi(x)$ και $\Gamma(x)$ στο $x = x_0$, το κύκλωμα θα αναπαράγει συνεχώς τη διεύθυνση διεμπλοκής $\Pi(x)$ για $x = x_0+1, x_0+2, x_0+3, \dots$ σε κάθε κύκλο. Παρά το γεγονός ότι η απλούστερη προσέγγιση για την υλοποίηση ενός διεμπλοκέα είναι να αποθηκεύουμε όλα τα πρότυπα διεμπλοκής σε ROMs, αυτή η προσέγγιση γίνεται σχεδόν ακατόρθωτη για έναν αποκωδικοποιητή Turbo ο οποίος υποστηρίζει μπλοκ πολλαπλών μεγεθών. Για παράδειγμα, το πρότυπο 3GPP LTE έχει

καθορίσει 188 διαφορετικά μεγέθη κωδικών Turbo, γεγονός που καθιστά την εφαρμογή διεμπλοκέα με βάση ROM πολύ αναξιόπιστη.



Σχήμα 12. Αρχιτεκτονική QPP διεμπλοκέα

4.2.4 Διεμπλοκέας Σχεδόν Τακτικής Μετάθεσης (Almost Regular Permutation - ARP) στο IEEE 802.16e πρότυπο WiMax

Αντίστοιχα, σύμφωνα με τα διεθνή τηλεπικοινωνιακά Standards (IEEE Std 802.16, 2004), ο διεμπλοκέας που έχει εγκριθεί από το πρότυπο WiMax ονομάζεται ως διεμπλοκέας σχεδόν τακτικής μετάθεσης (ARP) ο οποίος είναι επίσης άνευ διαμάχης (contention-free). Ο διεμπλοκέας ARP χρησιμοποιεί μια διαδικασία διεμπλοκής δύο σταδίων

$$[B_x, A_x] = [A_x, B_x], \text{ if } x \bmod 2 = 1.$$

Στο δεύτερο βήμα, ο διεμπλοκέας ARP υπολογίζει

$$\Pi(x) = \begin{cases} P_0x + 1, & \text{if } x \bmod 4 = 0 \\ P_0x + 1 + N/2 + P_1, & \text{if } x \bmod 4 = 1 \\ P_0x + 1 + P_2, & \text{if } x \bmod 4 = 2 \\ P_0x + 1 + N/2 + P_3, & \text{if } x \bmod 4 = 3 \end{cases},$$

όπου οι παράμετροι P_0, P_1, P_2 και P_3 είναι σταθερές και εξαρτώνται από την N .

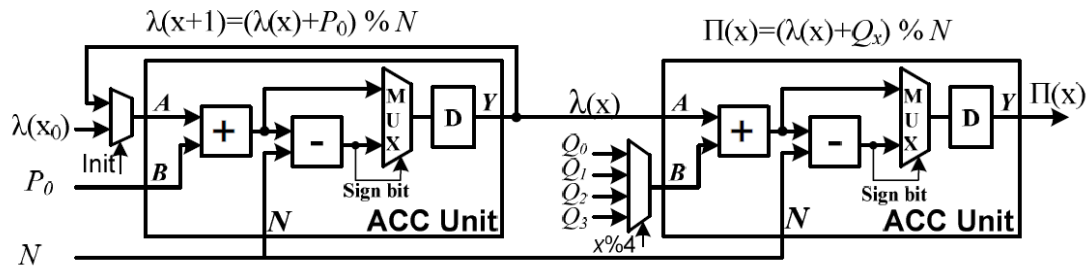
4.2.5 Πρακτική εφαρμογή του Διεμπλοκέα ARP

Ο διεμπλοκέας ARP μπορεί επίσης να υπολογίζεται αναδρομικά. Πιο ενδιαφέρον ακόμα είναι ότι ο ARP διεμπλοκέας μπορεί να εφαρμοστεί με παρόμοιο τρόπο όπως ο διεμπλοκέας QPP με την επαναχρησιμοποίηση των ίδιων δύο μονάδων πρόσθεσης σύγκρισης και επιλογής (Add-Compare-Choose - ACC), όπως φαίνεται στο Σχήμα 13. Έστω $\lambda(x)=P_0x$, $Q_0=1$, $Q_1=1+N/2+P_1$, $Q_2=1+P_2$, $Q_3=1+N/2+P_3$.

Μετά τη ρύθμιση μια αρχικής τιμής για το $\lambda(x=x_0)$, το κύκλωμα θα αναπαράγει συνεχώς τη διεύθυνση διεμπλοκής $\Pi(x)$ για $x = x_0+1, x_0+2, x_0+3, \dots$ σε κάθε ωρολογιακό κύκλο.

Αν συγκρίνουμε τα δύο κυκλώματα που απεικονίζονται στο Σχήμα 12 και στο Σχήμα 13, και οι δύο διατάξεις διεμπλοκής έχουν την ίδια λογική δομή. Οι διαφορές μεταξύ αυτών των δύο κυκλωμάτων είναι οι αρχικές τιμές για τις μονάδες ACC. Όπως μπορεί να φανεί, αυτή η ενοποιημένη αρχιτεκτονική απαιτεί μόνο λίγους αθροιστές και πολυπλέκτες, το οποίο οδηγεί σε πολύ χαμηλή πολυπλοκότητα και μπορεί να υποστηρίξει όλους τους QPP / ARP συνδυασμούς διεμπλοκής Turbo. Σε σύγκριση με τις παραδοσιακές εφαρμογές των διεμπλοκέων, οι οποίες χρειάζονται πολύπλοκες αριθμητικές μονάδες και/ή μνήμες RAM/ROM, ο προτεινόμενος QPP / ARP

διεμπλοκέας παρέχει μια αποτελεσματική λύση για την υποστήριξη πολλαπλών προτύπων διεμπλοκής Turbo.



Σχήμα 12. Αρχιτεκτονική ARP διεμπλοκέας

4.3 Παράλληλη αποκωδικοποίηση Turbo

Σε αυτή την ενότητα, θα γίνει αναφορά στις προκλήσεις αρχιτεκτονική για να σχεδιαστεί ένας υψηλής απόδοσης(διεκπεραιωτικότητας) παράλληλος αποκωδικοποιητής Turbo για λειτουργία multi-mode με μικρή επιβάρυνση. Λόγω της ιδιότητας της δομής του συνελκτικού πλέγματος-χωροδικτύωματος, ένα εκτεταμένο σε μήκος χωροδικτύωμα μπορεί να χωριστεί σε μικρότερα χωροδικτύωματα P. Στη συνέχεια, κάθε μικρότερο χωροδικτύωμα δέχεται ανεξάρτητη επεξεργασία από ένα ειδικό αποκωδικοποιητή MAP. Αγνοώντας το μικρή επιβάρυνση που εισάγεται από τον παράλληλο αλγόριθμο MAP, η διεκπεραιωτικότητα μπορεί να είναι αυξηθεί σχεδόν κατά έναν συντελεστή P. Για παράδειγμα, ο Bougard (2003) κατόρθωσε ρυθμό δεδομένων 75,6 Mbps μέσω απασχόλησης 7 αποκωδικοποιητών SISO που τρέχουν με ρυθμό 160 MHz.

Κλιμακούμενη Αρχιτεκτονική Αποκωδικοποιητή Turbo

Το κύριο ζήτημα στο σχεδιασμό της αρχιτεκτονικής παράλληλου Turbo αποκωδικοποιητή είναι γνωστό ότι είναι το θέμα της διεκπεραιωτικότητας του

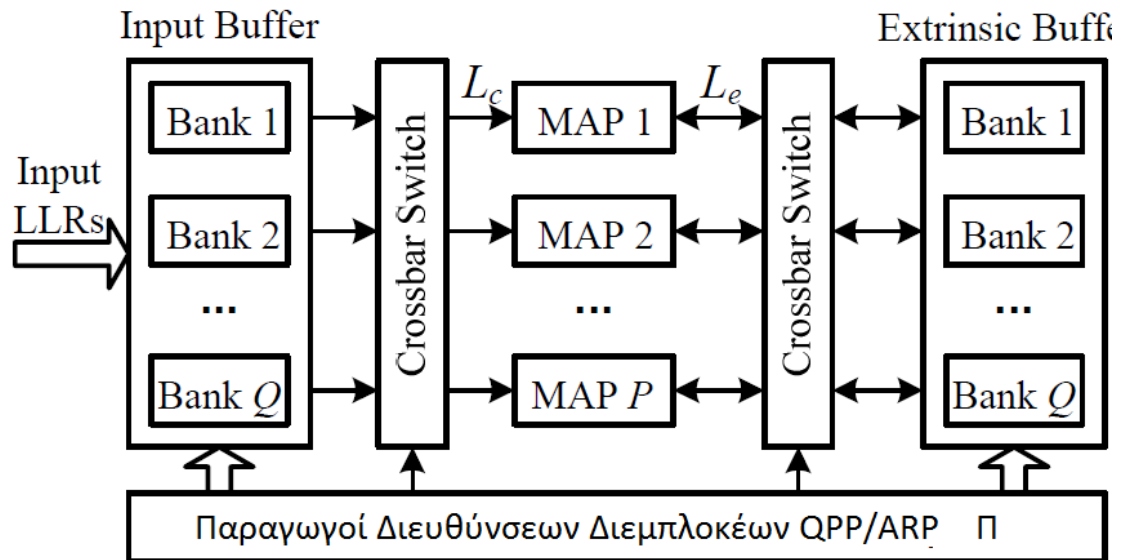
διεμπλοκέα λόγω του προβλήματος σύγκρουσης κατά την πρόσβαση στη μνήμη. Για την αντιμετώπιση του θέματος με την διεκπεραιωτικότητα στην αποκωδικοποίηση καναλιού δεδομένων, τόσο το 3GPP LTE αλλά και το WiMAX 802.16e IEEE απασχολούν τον contention-free διεμπλοκείς. Μία κωδική λέξη μπορεί να διαιρεθεί σε P υπο-κωδικές λέξεις και MAP αποκωδικοποιητές μπορούν να χρησιμοποιηθούν για να αποκωδικοποιήσει κάθε υπο-κωδική λέξη ταυτόχρονα το οποίο οδηγεί σε P -επιπέδων παράλληλη αρχιτεκτονική. Το σχήμα 14 (a) δείχνει την προτεινόμενη αρχιτεκτονική του παράλληλου αποκωδικοποιητή με βάση contention-free 4G διεμπλοκείς. Αυτή η αρχιτεκτονική είναι ευέλικτη στο ότι απασχολεί τόσο εντός όσο και μεταξύ κωδικών λέξεων παράλληλα συστήματα αποκωδικοποίησης για τη βελτίωση της συνολικής αποτελεσματικότητας και απόδοσης. Η λειτουργία ενδο-κωδική λέξης χρησιμοποιείται για την αποκωδικοποίηση των κωδικών λέξεων μεγάλου μεγέθους. Ο παραλληλισμός επιτυγχάνεται διαιρώντας ολόκληρο το μπλοκ N σε P υπο-μπλοκ (sub-blocks SBS) και την ανάθεση P MAP αποκωδικοποιητών που εργάζονται παράλληλα για τη μείωση της χρονοκαθυστέρησης σε $O(N/P)$. Η λειτουργία της ενδιάμεσης (μεταξύ) κωδικής λέξης χρησιμοποιείται για κωδικές λέξεις μικρού μεγέθους, έχοντας P μικρές κωδικές λέξεις να αποκωδικοποιούνται ταυτόχρονα και ανεξάρτητα, ούτως ώστε η συνολική λανθάνουσα καθυστέρηση μειώνεται σε $O(N/P)$, επίσης.

Η δομή της μνήμης έχει σχεδιαστεί για να υποστηρίζει ταυτόχρονη πρόσβαση των LLRs από πολλούς αποκωδικοποιητές MAP τόσο σε γραμμικές διευθύνσεις όσο και σε διαπλέκοντες τρόπους διευθυνσιοδότησης. Αυτό επιτυγχάνεται με διαμερισμό της μνήμη σε P ξεχωριστές θυρίδες/«τράπεζες»(Banks). Κάθε θυρίδα έχει το ίδιο μέγεθος και μπορεί να είναι ανεξάρτητα προσβάσιμη. Επειδή P MAP αποκωδικοποιητές έχουν πάντα πρόσβαση στα δεδομένα συγχρόνως σε μια συγκεκριμένη μετατόπιση x , εγγυάται ότι δεν συμβαίνουν συγκρούσεις πρόσβασης στη μνήμη λόγω της contention-free ιδιότητας του

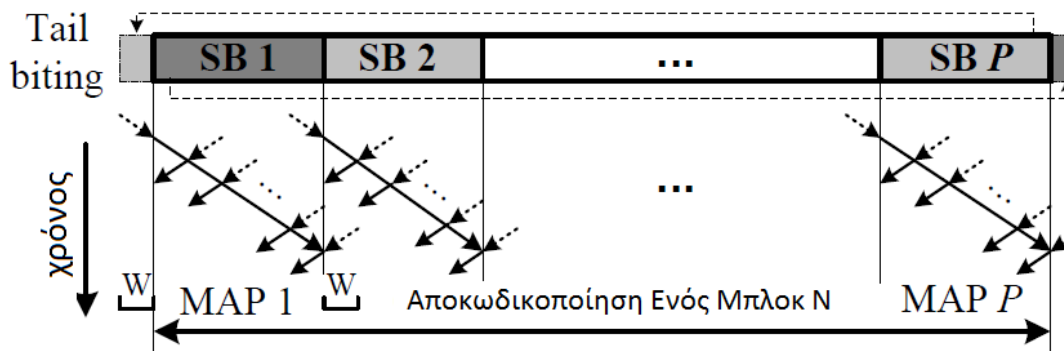
$[\Pi(x+jM) / M] [\Pi(x+kM) / M]$, όπου x είναι η μετατόπιση στο j υπο-μπλοκ και k ($0 \leq j < k < P$),

και το M είναι το μήκος του υπο-μπλοκ ($M = N/P$). Μία πλήρης διασταυρούμενη ράβδος (Crossbar) χρησιμοποιείται για τη δρομολόγηση δεδομένων μεταξύ P αποκωδικοποιητών MAP και P θυρίδων μνήμης. Ένα παράδειγμα παράλληλης αποκωδικοποίησης (σε λειτουργία εντός κωδικής λέξης) για διπλούς δυαδικούς

κώδικες απεικονίζεται στο Σχήμα 14 (b). Αξίζει αν γνωρίζουμε ότι η έννοια αυτή ισχύει και για τους δυαδικούς κώδικες επίσης.



(a) Multi-MAP αρχιτεκτονική αποκωδικοποιητή Turbo

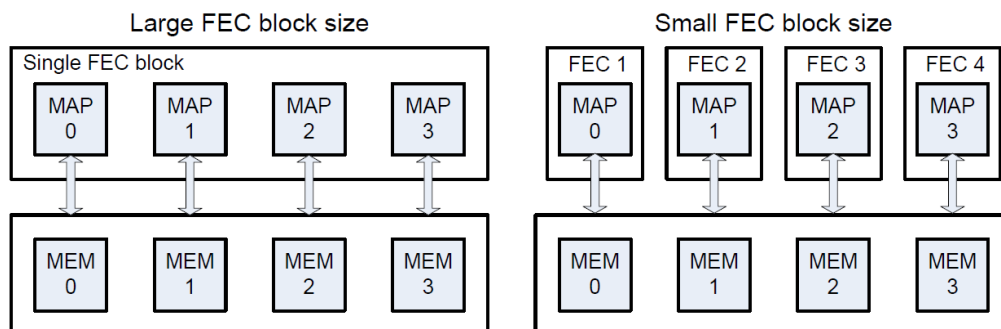


(b) Παράλληλη MAP αποκωδικοποίηση

Σχήμα 14. Αρχιτεκτονική παράλληλο αποκωδικοποιητή

Μελέτη Περίπτωσης αποκωδικοποιητής Turbo 100 Mbps για 3^{ης} κατηγορίας 3GPP LTE συσκευές.

Το P επίπεδο P παραλληλισμού μπορεί να προσαρμοστεί σε μία δεδομένου μέγεθους κωδική λέξη. Για τις μικρότερες κωδικές λέξεις, διαπιστώσαμε ότι η μέθοδος αποκωδικοποίησης P = 1 ικανοποιεί τις απαιτήσεις απόδοσης/διεκπεραιώτητας με λιγότερη λογική πολυπλοκότητα. Υπάρχουν αρκετές τιμές P που μπορούν να καταφέρουν πρόσβαση στη μνήμη άνευ διαμάχης σε όλες τις μεγάλους μεγέθους κωδικές λέξεις. Για να υποστηριχτεί ο ρυθμός δεδομένων κατηγορίας 3 του 3GPP LTE της τάξεως ~ 100 Mbps, θα ορίσουμε το μέγιστο παραλληλισμό να είναι P=4. Όπως φαίνεται στο Σχήμα 15, έχουμε P=4 τράπεζες μνήμης και ανάλογα με το μέγεθος της κωδικής λέξης, μπορούμε να ρυθμίσουμε P=4 αποκωδικοποιητές MAP αναλόγως ώστε να ανταποκριθούν στις απαιτήσεις διεκπεραίωσης για διαφορετικά μεγέθη κωδικών λέξεων.



Σχήμα 15. Κλιμακούμενοι παράλληλοι αποκωδικοποιητές Turbo ανάλογα με το μέγεθος της κωδικής λέξης

Ανάλυση Αρχιτεκτονικής Εξισορρόπησης

Από τις παραπάνω περιγραφές, γνωρίζουμε ότι υψηλής διεκπεραιότητα μπορεί να επιτευχθεί με τη χρήση πολλαπλών αποκωδικοποιητών MAP και πολλαπλών τραπεζών μνήμης. Ωστόσο, η απόδοση (διεκπεραιότητα) δεν μπορεί πάντοτε να αυξάνεται γραμμικά με το επίπεδο παραλληλισμού. Καθώς οι SRAMs γίνονται όλο και μικρότερες, η περιοχή αποδοτικότητας θα μειωθεί. Επίσης, θα υπάρξει μια σταθερή λανθάνουσα καθυστέρηση για το παράθυρο ολίσθησης της αποκωδικοποίησης MAP. Σε αυτή την ενότητα, θα αναλύσουμε τις επιπτώσεις του παραλληλισμού στη διεκπεραιότητα, την περιοχή και την κατανάλωση ενέργειας. Η μέγιστη διεκπεραιότητα (κατά τον Y. Sun) υπολογίζεται ως

$$\text{Διεκπεραιότητα} = \frac{N}{\text{Χρόνος Αποκωδικοποίησης}} \approx \frac{N \cdot f}{2 \cdot I \cdot \left(\frac{\tilde{N}}{P} + 3\tilde{W} \right)},$$

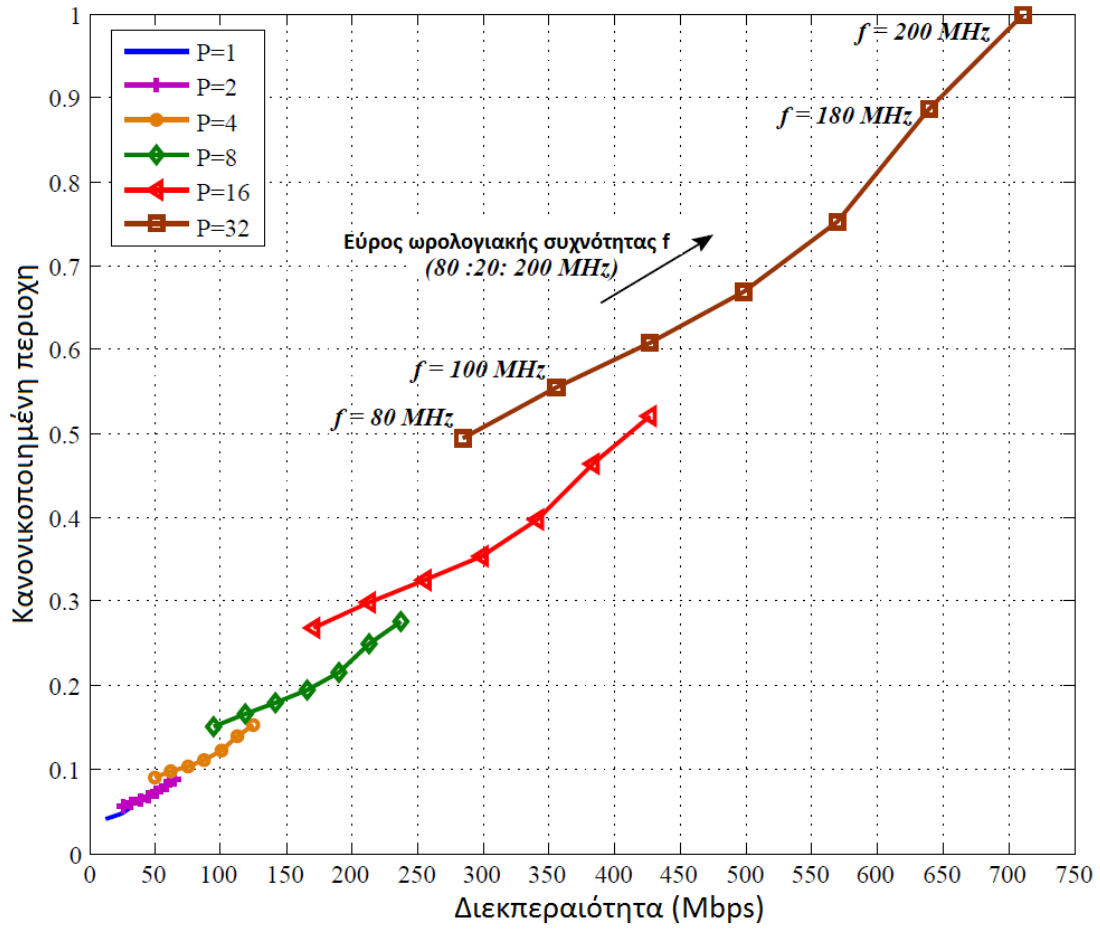
Όπου $\tilde{N} = N/2$ είναι το μήκος του κώδικα (σε Radix-4), I είναι ο αριθμός των επαναλήψεων (περιέχει δύο μισές επαναλήψεις), $3\tilde{W}$ είναι η λανθάνουσα καθυστέρηση αποκωδικοποίησης για κάθε αποκωδικοποιητή MAP, και f είναι η συχνότητα του ρολογιού. Η συνολική έκταση μπορεί να υπολογιστεί ως

$$\text{Area} \approx P \cdot A_{\text{map}}(f) + A_{\text{mem}}(P, N) + A_{\text{route}}(P, f),$$

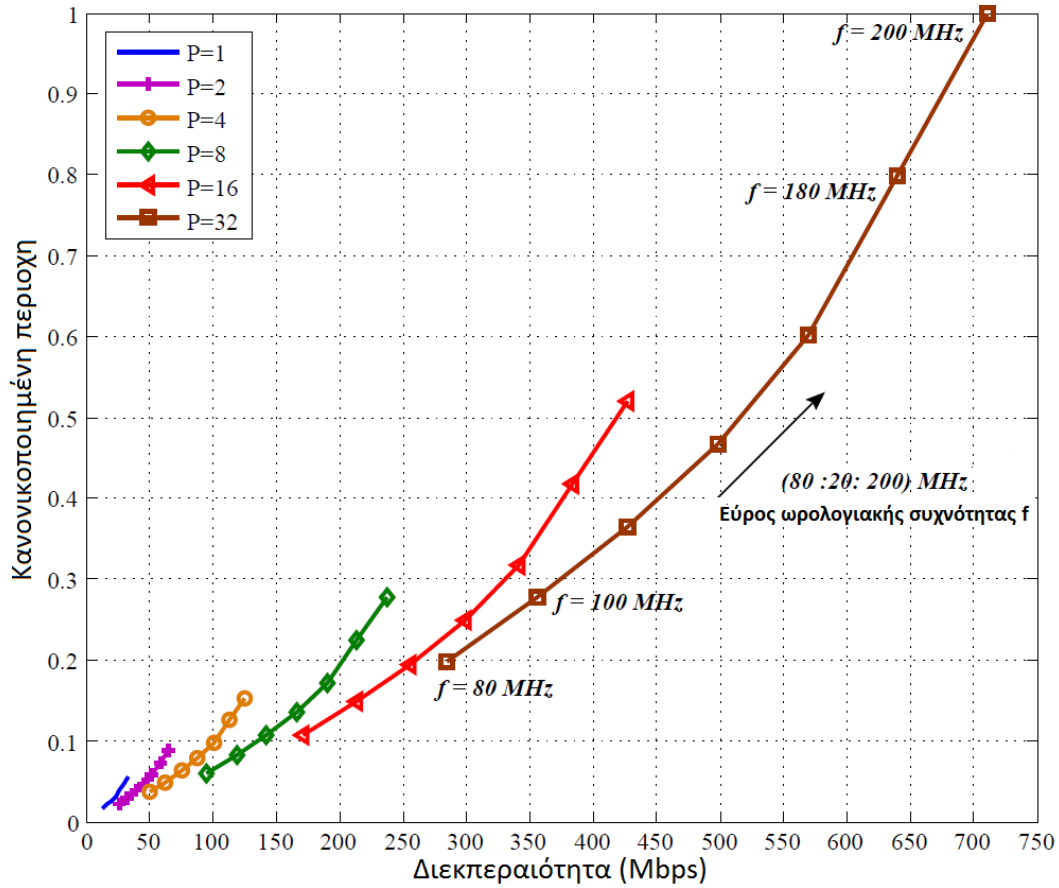
Στο οποίο: A_{map} είναι μια περιοχή ενός αποκωδικοποιητή MAP, η οποία θα αυξηθεί με f, A_{mem} είναι η περιοχή μνήμης που θα αυξηθεί με N και P, και A_{route} είναι το κόστος δρομολόγησης (διασταυρούμενοι ράβδου συν διεμπλοκείς), το οποίο θα αυξηθεί και κατά P και κατά f. Ας επισημάνουμε ότι η πολυπλοκότητα της πλήρους διασταυρούμενης ράβδου αυξάνεται πραγματικά με P^2 . Για να εκτελέσουμε την ανάλυση περιοχής και εξισορρόπησης ενέργειας, ο αποκωδικοποιητής περιγράφηκε σε Verilog RTL και υλοποιήθηκε για τεχνολογία 65 nm που χρησιμοποιεί σχεδιασμό

μεταγλώττισης Synopsys . Η ανάλυση εξισορρόπησης περιοχής παρουσιάζεται στο Σχήμα 16 το οποίο απεικονίζει την κανονικοποιημένη περιοχή σε σχέση με τη διεκπεραιότητα για διαφορετικά επίπεδα παραλληλισμού και τους στόχους της συχνότητας του ρολογιού (80-200 MHz, με βήμα 20 MHz). Το Σχήμα 17 δείχνει τη δυναμική ανάλυση εξισορρόπησης δύναμη(ενέργειας). Όπως μπορεί να φανεί, για μια δεδομένη διεκπεραιότητα μπορεί να υπάρχουν πολλαπλές διαρθρώσεις, αναμενόμενα δε ο υψηλότερος παραλληλισμός (ως εκ τούτου επίσης απαιτεί χαμηλότερη συχνότητα) είναι επωφελής για την εξοικονόμηση ενέργειας, αλλά δυσμενής για την αξιοποίηση περιοχή, και αντίστροφα.

Ο Πίνακας 3 συγκρίνει την ευελιξία της αρχιτεκτονικής και την απόδοση της αποκωδικοποίησης του προτεινόμενου αποκωδικοποιητή με τους υπάρχοντες υπερσύγχρονους αποκωδικοποιητές Turbo. Οι προγραμματιζόμενοι VLIW /SIMD επεξεργαστές (κατά τους Lin, Muller και Ituero & Lopez-Vallejo), έχουν σχεδιαστεί για να υποστηρίξουν την αποκωδικοποίηση πολλαπλών κωδικών Turbo. Σύμφωνα με τους THUL και Bougard, αρχιτεκτονικές υλικού ASIC προτείνονται για απλούς δυαδικούς κώδικες Turbo οι οποίοι βασίζονται στον αλγόριθμο Log-MAP. Κατά τον Prescher, μια διεκπεραιότητα αποκωδικοποίησης 757 Mbps επιτυγχάνεται με τη χρησιμοποίηση 64 υπο-βέλτιστων σταθερών Log-MAP SISO αποκωδικοποιητών. Αν και ο αποκωδικοποιητής του Prescher επιτυγχάνει υψηλή απόδοση με χαμηλό κόστος, έχει μερικούς περιορισμούς, όπως ότι ο παρεμβολέας είναι μη συμβατός με τα πρότυπα και δεν μπορεί να υποστηρίξει διπλούς δυαδικούς Turbo κώδικες. Οι συγκρίσεις του πίνακα 3 μας δείχνουν ότι η προτεινόμενη αρχιτεκτονική μας δεν έχει μόνο ευελιξία στην υποστήριξη πολλαπλών κωδικών Turbo (απλούς δυαδικούς + διπλούς δυαδικούς κώδικες), αλλά επιτυγχάνει επίσης μια πολύ υψηλή απόδοση(διεκπεραιότητα). Επίσης, είναι κλιμακούμενη από την άποψη της παραλληλίας και τον ρυθμό δεδομένων.



Σχήμα 16. Κανονικοποιημένη περιοχή έναντι διεκπεραιότητας-απόδοσης (N = 6144, I = 6, W = 32).



Σχήμα 17. Κανονικοποιημένη ενέργεια-δύναμη έναντι διεκπεραιότητας-απόδοσης (N = 6144, I = 6, W = 32).

Πίνακας 3. Σύγκριση μεταξύ υπάρχοντων αρχιτεκτονικών αποκωδικοποιητών Turbo

Work	Architectures	Flexibility	Algorithm	Parallelism	Frequency	Throughput
Lin, 2006	32-wide SIMD	Multi-code	Max-LogMap	4 window	400 MHz	2.08 Mbps
Ituero, 2006	Cluster VLIW	Multi-code	LogMap	Dual cluster	80 MHz	10 Mbps
Muller, 2006	ASIP SIMD	Multi-code	Max-LogMap	16 ASIP	335 MHz	100 Mbps
Thul, 2006	ASIC	Single-code	LogMap	6 SISO	166 MHz	59.6 Mbps
Bougard, 2003	ASIC	Single-code	LogMap	7 SISO	160 MHz	75.6 Mbps
Prescher, 2005	ASIC	Single-code	Const-LogMap	64 SISO	256 MHz	758 Mbps
Our work	ASIC	Multi-code	LogMap	32 SISO	200 MHz	711 Mbps

Πειραματική ανάλυση λειτουργίας Turbo codes

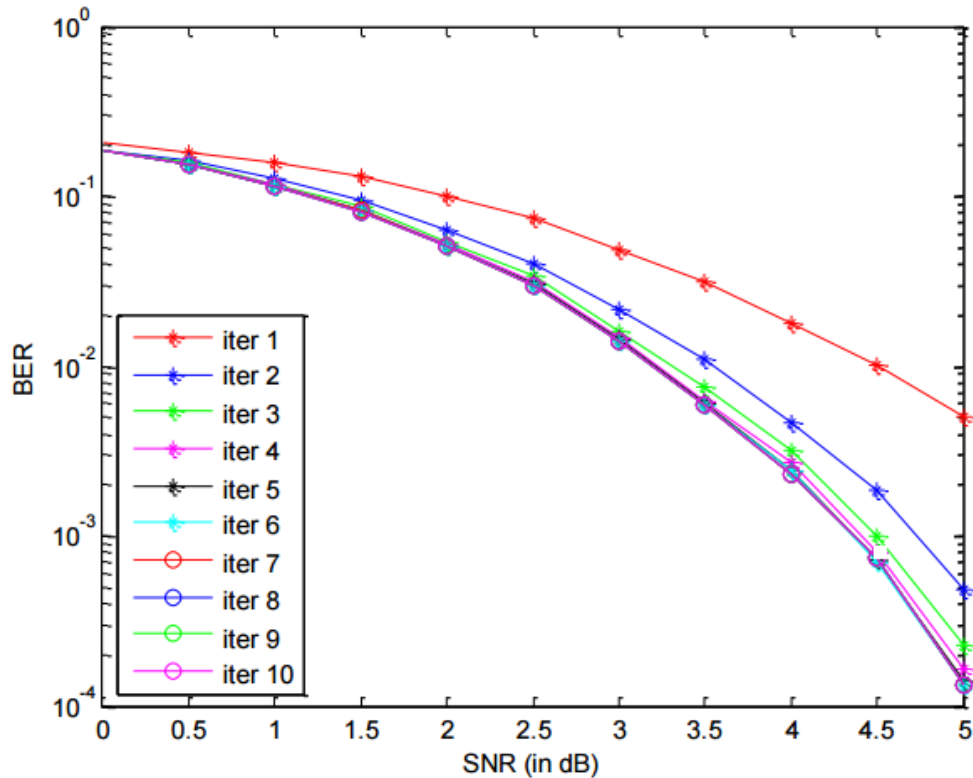
5.1 Περαιτέρω ανάλυση απόδοσης σφάλματος κωδικών Turbo

Όπως έχουμε ήδη δει, οι κώδικες Turbo είναι ικανοί να παράγουν χαμηλά ποσοστά σφάλματος σε εκπληκτικά χαμηλά SNR (Signal to Noise Ratio-εκφράζει το λόγο της ισχύος του σήματος προς την ισχύ του θορύβου) αν το μέγεθος του πλαισίου K διατηρείται μεγάλο και οι Μεταθέσεις επιλέγονται τυχαία. Οι κώδικες Turbo είναι επαναληπτικοί κώδικες και η ανταλλαγή αμοιβαίων πληροφοριών μεταξύ των συστατικών αποκωδικοποιητών, έχει ως αποτέλεσμα την καλύτερη απόδοση του κώδικα. Όσο μεγαλύτερος είναι ο αριθμός των επαναλήψεων, τόσο μεγαλύτερη είναι η ανταλλαγή πληροφοριών μεταξύ των συστατικών κωδικοποιητών (constituent encoders). Ως αποτέλεσμα, ο κώδικας μπορεί να αποδώσει καλύτερα. Ωστόσο, όταν ο αριθμός των επαναλήψεων γίνεται πολύ υψηλός, δεν υπάρχει μεγάλη βελτίωση στην απόδοση. Έτσι, πρέπει να οριοποιηθούν κάποια μέγιστα όρια ως προς τον αριθμό των επαναλήψεων και του SNR για τη βέλτιστη απόδοση του κώδικα.

Καθώς η αποκωδικοποίηση κώδικα turbo είναι υπολογιστικά πολύ εντατική, στο μεγαλύτερο μέρος της προσομοίωσης τα αποτελέσματα των αποδόσεων είναι για μικρά μήκη πλαισίων (frame sizes).

5.1.1 Απόδοση BER για frame size $K = 40$

Το πρόγραμμα του turbo κώδικα προσομοιώθηκε για μέγεθος πλαισίου (frame size) $K = 40$ σε ένα κανάλι με εξασθένιση Rayleigh. Για να διατηρηθεί η προσομοίωση σύντομη χρονικά, ο αριθμός των πλαισίων για κάθε SNR υπολογίστηκε ως 500. Έτσι, για ένα μέγεθος πλαισίου 40, 20000 bits στάλθηκαν σε κάθε τιμή SNR για να πάρουμε το ποσοστό σφάλματος δυαδικού ψηφίου (bit error rate – BER). Το εύρος SNR που χρησιμοποιήθηκε ήταν από 0 έως 5 dB. Ο αριθμός των επαναλήψεων αποκωδικοποιητή επιλέχθηκε να είναι 10. Το BER για τις επαναλήψεις παρουσιάζεται παρακάτω στο Σχήμα 18.



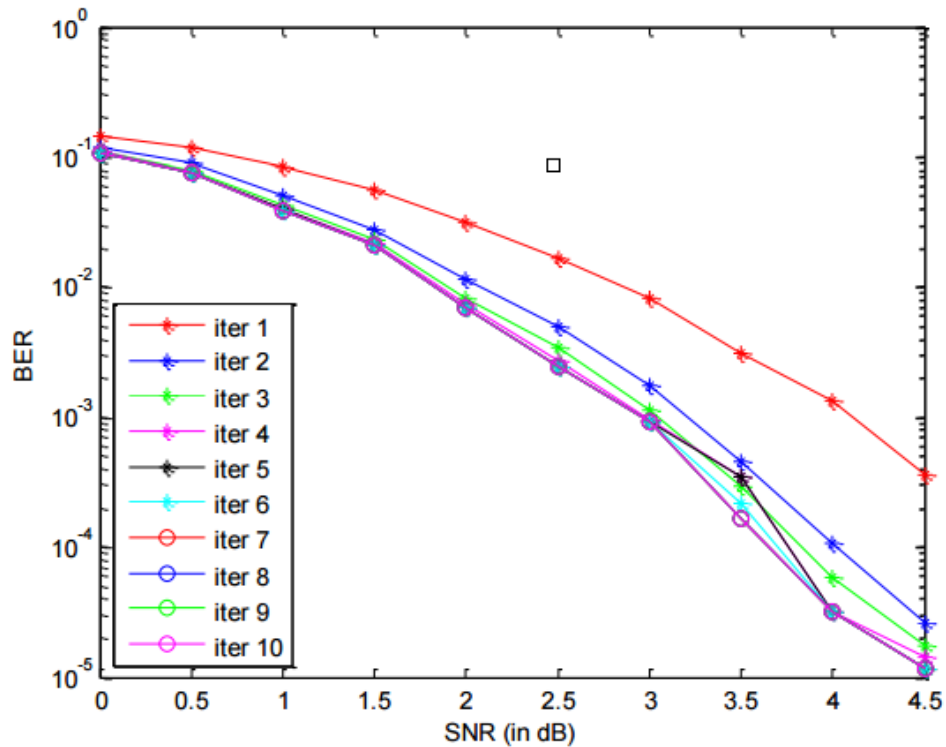
Σχήμα 18. BER για Frame size $K = 40$ κώδικα turbo σε κανάλι Rayleigh

Επανάληψεις	BER	Επανάληψεις	BER
1	0,005	6	$1,338 \times 10^{-4}$
2	$4,886 \times 10^{-4}$	7	$1,339 \times 10^{-4}$
3	$2,285 \times 10^{-4}$	8	$1,339 \times 10^{-4}$
4	$1,655 \times 10^{-4}$	9	$1,34 \times 10^{-4}$
5	$1,418 \times 10^{-4}$	10	$11,34 \times 10^{-4}$

Πίνακας 4. BER για K = 40 κώδικα turbo σε κανάλι Rayleigh.

Οι τιμές BER στο τέλος κάθε επανάληψης δίνονται στον Πίνακα 4 BER για K = 40 κωδικό turbo πάνω σε κανάλι Rayleigh. Ο turbo κώδικας ήταν σε θέση να επιτύχει BER $0,5 \times 10^{-2}$ μετά την πρώτη επανάληψη αποκωδικοποιητή. Ο BER βελτιώθηκε σε $1,34 \times 10^{-4}$ μετά την 10^η επανάληψη. Μπορεί να γίνει κατανοητό ότι καθώς ο αριθμός των επαναλήψεων αυξάνεται, η απόδοση BER βελτιώνεται. Ωστόσο, το ποσοστό βελτίωσης μειώνεται. Αυτό απεικονίζεται με τις επικαλυπτόμενες καμπύλες μετά από 5 επαναλήψεις (η μία πάνω στην άλλη). Το BER μετά από 5 επαναλήψεις είναι 1.418×10^{-4} και δεν παρουσιάζει σημαντική βελτίωση. Έτσι, ο αριθμός των επαναλήψεων πρέπει να διατηρείται τέτοιος ώστε να αποφεύγονται οι επιπλέον αχρείαστοι υπολογισμοί.

Η καμπύλη BER για frame size K = 40 σε ένα κανάλι με το φαινόμενο του λευκού θορύβου AWGN (additive white Gaussian noise) εμφανίζεται Σχήμα 19. Το BER μετά την πρώτη επανάληψη αποκωδικοποιητή είναι 3.628×10^{-4} . Το BER μειώνεται με την αύξηση των επαναλήψεων και στο τέλος της δέκατης επανάληψης είναι 1.152×10^{-5} .



Σχήμα 19. BER σε Frame size $K = 40$ κώδικα turbo σε κανάλι AWGN.

5.1.2 Απόδοση BER για frame size $K = 100$

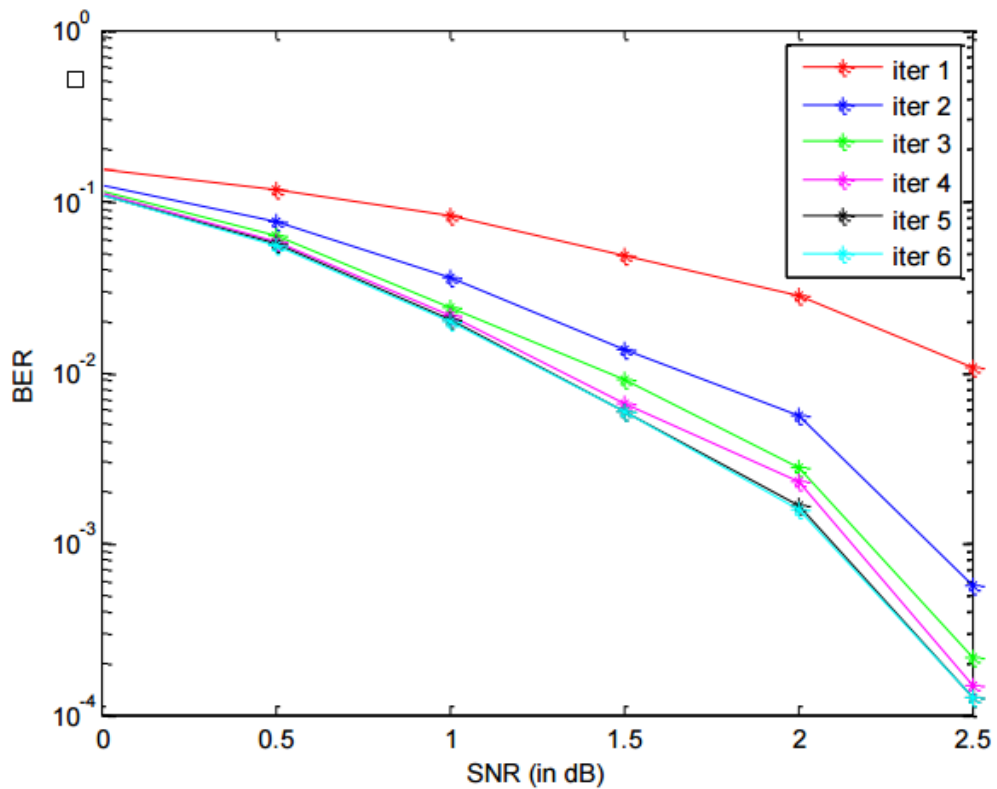
Η απόδοση του Turbo κώδικα μπορεί να βελτιωθεί με την αύξηση του μεγέθους του πλαισίου K . Ο κώδικας μπορεί επιτύχει υψηλότερο BER με την αύξηση του frame size. Αυτό συμβαίνει επειδή ο διεμπλοκέας μεταθέτει τα δεδομένα και ο αποκωδικοποιητής είναι σε θέση να αποκωδικοποιήσει τα δεδομένα καλύτερα. Η προσομοίωση για τον κώδικα turbo έγινε με μέγεθος πλαισίου $K = 100$ κρατώντας τον αριθμό των επαναλήψεων σε 6. Ο BER για τις επαναλήψεις δίνεται στον Πίνακα 5. Η καμπύλη του BER για τον turbo κώδικα με frame size $K = 100$ φαίνεται στο Σχήμα 20. Η εικόνα δείχνει μια βελτίωση στην απόδοση BER σε σχέση με την εικόνα του frame size $K = 40$. Το BER μειώνεται με την αύξηση του αριθμού των επαναλήψεων. Αυτή η συμπεριφορά είναι παρόμοια με την περίπτωση με μέγεθος πλαισίου 40. Ωστόσο, το frame size $K = 100$ είναι σε θέση να επιτύχει BER σχεδόν

10^{-4} σε SNR 2,5 dB. Το frame size $K = 40$ ήταν σε θέση να επιτύχει τέτοιο BER σε SNR 5 dB. Έτσι, μπορούμε να συμπεράνουμε ότι με την αύξηση του frame size(του μεγέθους του πλαισίου δηλαδή), ο κώδικας

μπορεί να επιτύχει το ίδιο BER σε πολύ χαμηλότερα SNR.

Επαναλήψεις	BER	Επαναλήψεις	BER
1	$1,07 \times 10^{-2}$	4	$1,472 \times 10^{-4}$
2	$5,62 \times 10^{-4}$	5	$1,245 \times 10^{-4}$
3	$2,1518 \times 10^{-4}$	6	$1,246 \times 10^{-4}$

Πίνακας 5. BER για frame size $K = 100$ κώδικα turbo.



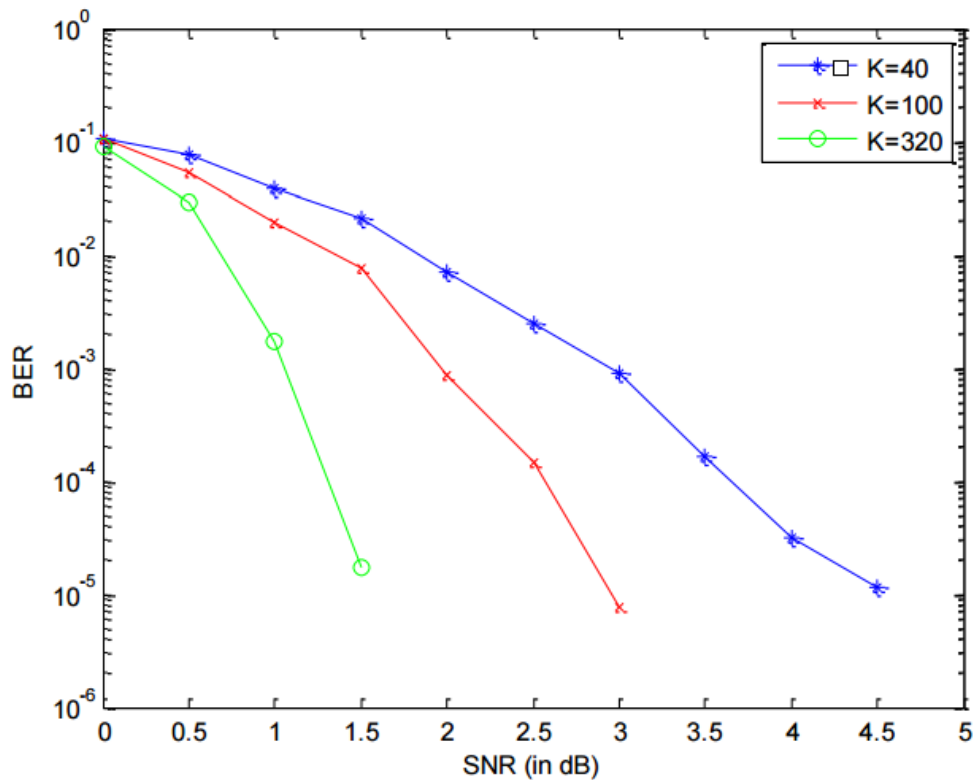
Σχήμα 20. BER για frame size $K = 100$ κώδικα turbo μετά από 6 επαναλήψεις αποκωδικοποιητή.

5.1.3 Ο BER ως λειτουργία του frame size K (Latency-λανθάνουσα κατάσταση)

Η σύγκριση των επιδόσεων του κώδικα turbo μπορεί να γίνει με τη γραφική αναπαράσταση του BER για διαφορετικά frame sizes K όπως φαίνεται στο Σχήμα 21. Το σχήμα δείχνει ότι με την αύξηση του frame size K, η απόδοση BER του κώδικα βελτιώνεται. Ως αποτέλεσμα, χαμηλότερο BER μπορεί να επιτευχθεί με τη διατήρηση σταθερής SNR. Οι τιμές BER για το μέγεθος του πλαισίου K 40, 100 και 320 δίνονται στον Πίνακα 6. Μπορεί να φανεί στο σχήμα ότι για το μέγεθος πλαισίου K 320, BER $1,75 \times 10^{-5}$ επιτυγχάνεται σε SNR 1,5 dB. Ωστόσο, για μέγεθος πλαισίου K 100 και 40 μπορεί να επιτευχθεί μόνο BER $0,76 \times 10^{-2}$ και 0,0212 αντίστοιχα, για το ίδιο SNR. Μεγαλύτερα frame sizes σημαίνει μεγαλύτερη καθυστέρηση, εφόσον η κωδικοποίηση και η αποκωδικοποίηση γίνεται ανά frame. Έτσι, η βελτίωση απόδοσης επιτυγχάνεται με το κόστος της αυξημένης λανθάνουσας κατάστασης.

Frame size K	BER	SNR (in dB)
40	$1,152 \times 10^{-5}$	4,5
100	$7,745 \times 10^{-6}$	3
320	$1,75 \times 10^{-5}$	1,5

Πίνακας 6. BER για frame size K = 40, 100 και 320.



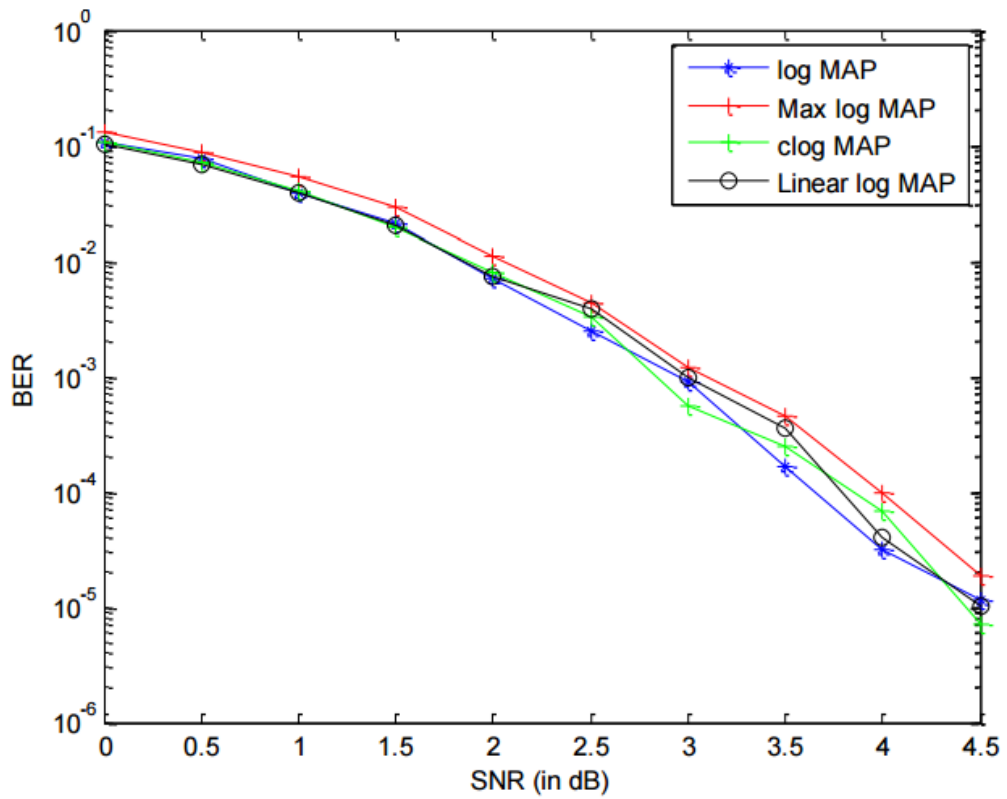
Σχήμα 20. Σύγκριση επιδόσεων κώδικα Turbo μετά απο 10 επαναλήψεις αποκωδικοποιητή για frame sizes K 40, 100 και 320.

5.1.4 Απόδοση BER σε κανάλι AWGN για αλγόριθμο max *.

Το πρόγραμμα turbo κώδικα προσομοιώθηκε για frame size $K = 40$ σε κανάλι με θόρυβο AWGN. Ο αριθμός των πλαισίων για κάθε SNR ελήφθη ως 500 και ο κώδικας εκτελέστηκε για 10 επαναλήψεις αποκωδικοποιητή. Η αποκωδικοποίηση έγινε χρησιμοποιώντας και τις τέσσερις παραλλαγές του αλγορίθμου max * δηλαδή, log MAP, max-log MAP, constant-log MAP (σταθερής καταγραφής), linear-log MAP (γραμμικής καταγραφής). Για να συγκρίνουμε τις αποδόσεις, ο BER μετά από 10 επαναλήψεις αναπαραστάται στο Σχήμα 21. Οι τιμές του BER μετά από 10 επαναλήψεις αποκωδικοποιητή δίνονται στον Πίνακα 7.

Algorithm	BER
Log MAP	$1,152 \times 10^{-5}$
Max-log MAP	$1,8967 \times 10^{-5}$
Constant-log MAP	$0,7137 \times 10^{-5}$
Linear -log MAP	$1,0358 \times 10^{-5}$

Πίνακας 7. BER για frame size $K = 40$ μετά από 10 επαναλήψεις αποκωδικοποίησης.



Σχήμα 21. BER για frame size $K = 40$ κώδικα turbo μετά απο 10 επαναλήψεις αποκωδικοποίησης σε κανάλι με θόρυβο AWGN.

Το σχήμα δείχνει ότι ο κώδικας ήταν σε θέση να επιτύχει BER τόσο χαμηλά όσο 10^{-5} για τα bits εισόδου. Οι log MAP, linear log MAP και constant log MAP αλγόριθμοι παρουσιάζουν δυσδιάκριτες επιδόσεις για το καθορισμένο μέγεθος του πλαισίου $K = 40$. Ωστόσο, στην πράξη η απόδοση του αλγορίθμου Log MAP είναι η καλύτερη από όλους. Αυτό επίσης ισχύει και στην προσομοίωση μας, όταν το μέγεθος πλαισίου K αυξάνεται. Βέβαια, ο κώδικας έχει προσομοιωθεί για $K = 40$ καθώς τα μεγαλύτερα frame sizes διαρκούν πολύ περισσότερο χρόνο για τις προσομοιώσεις για να τρέξουν.

Μπορεί να δει κανείς ότι η απόδοση του αλγορίθμου max-log MAP είναι σημαντικά χειρότερη ανάμεσα στους τέσσερις αλγόριθμους. Ωστόσο, έχει το λιγότερη πολυπλοκότητα και χρειάζεται το μικρότερο χρόνο για την προσομοίωση. Έτσι, υπάρχει ένας συμβιβασμός μεταξύ πολυπλοκότητας και απόδοσης.

Στο σχήμα, παρατηρείται ότι η απόδοση του linear-log MAP και του constant-log MAP μερικές φορές είναι ελαφρώς καλύτερη από τον αλγόριθμο log MAP. Αυτό είναι ένα ενδιαφέρον και μη αναμενόμενο φαινόμενο. Ο λόγος για αυτό είναι ότι οι δύο συστατικοί αποκωδικοποιητές είναι οι καταλληλότεροι όσον αφορά την ελαχιστοποίηση του τοπικού BER, αλλά η συνολική αποκωδικοποίηση turbo δεν εγγυάται την ελαχιστοποίηση του ολικού BER. Αυτό συμβαίνει λόγω των τυχαίων διαταραχών λόγω προσέγγισης στον υπολογισμό μερικών μετρικών διαδρομών και LLRs στους αλγόριθμους linear-log MAP και constant-log MAP. Αυτές οι διαταραχές είναι μικρές και τα αποτελέσματα του linear-log MAP και του constant-log MAP είναι πολύ κοντά σε αυτά από τον αλγόριθμο log MAP.

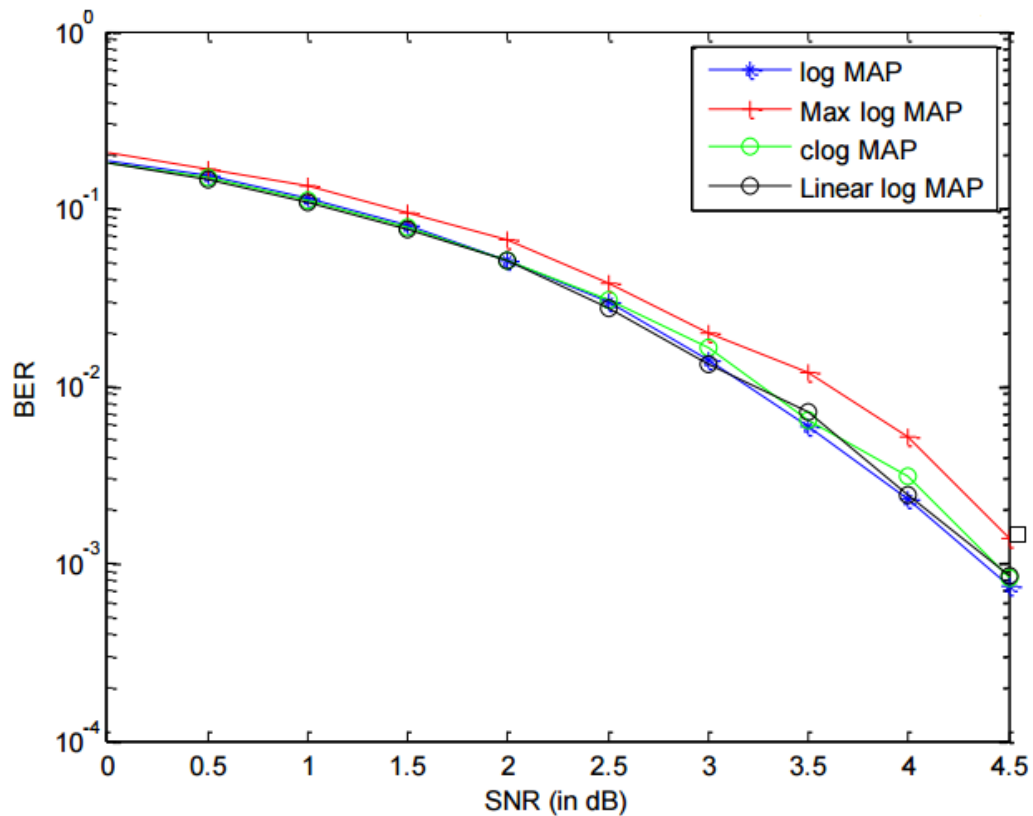
5.1.5 Απόδοση BER σε κανάλι Rayleigh για αλγόριθμο max * .

Το πρόγραμμα του turbo κώδικα προσομοιώθηκε frame size $K = 40$ σε ένα κανάλι με εξασθένιση Rayleigh. Οι παράμετροι παρέμειναν οι ίδιοι με την προηγούμενη προσομοίωση. Το BER μετά 10 επαναλήψεις αναπαριστάται ως συνάρτηση του SNR στο Σχήμα 22. Μετά από 10 επαναλήψεις αποκωδικοποίησης, ο αλγόριθμος log MAP επιτυγχάνει BER περίπου $0,74 \times 10^{-3}$ σε SNR 4.5 dB. Σε περίπτωση καναλιού με θόρυβο AWGN, ο BER που επιτεύχθηκε ήταν περίπου 10^{-5} . Όπως φάνηκε παραπάνω, η απόδοση του κώδικα turbo σε Rayleigh κανάλι για ένα συγκεκριμένο SNR είναι χειρότερη σε σύγκριση με το κανάλι AWGN. Επίσης, το BER του max-log MAP είναι το υψηλότερο από όλους και αυτό του log MAP είναι το χαμηλότερο όλων. Έτσι, ο αλγόριθμος log MAP παρουσιάζει την καλύτερη απόδοση με τίμημα την υψηλότερη πολυπλοκότητα και ο max-log MAP παρουσιάζει τη χειρότερη απόδοση, αλλά είναι επίσης ο λιγότερο πολύπλοκος από όλους. Η

πολυπλοκότητα και η απόδοση των linear-log MAP και constant-log MAP είναι ανάμεσα στους αλγόριθμους log MAP και max-log MAP.

Algorithm	Log MAP	Max-log MAP	Constant-log MAP	Linear -log MAP
BER	$0,74 \times 10^{-3}$	$0,14 \times 10^{-2}$	$0,82 \times 10^{-3}$	$0,84 \times 10^{-3}$

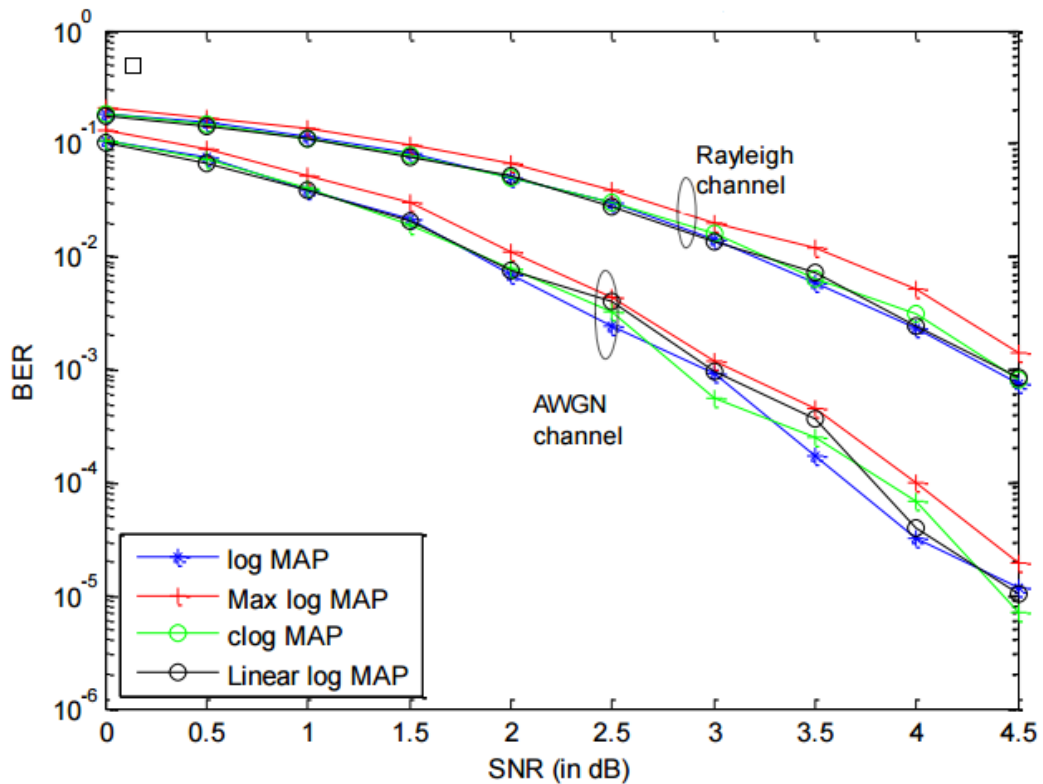
Πίνακας 8. Απόδοση BER για frame size $K = 40$ σε κανάλι με διάλεια Rayleigh (φθίνων) μετά από 10 επαναλήψεις αποκωδικοποίησης.



Σχήμα 22. Απόδοση BER για frame size $K = 40$ κώδικα turbo σε κανάλι με διάλειαση Rayleigh (φθίνων) μετά απο 10 επαναλήψης αποκωδικοποίησης.

5.1.6 Σύγκριση απόδοσης σε κανάλια Rayleigh και AWGN

Για τη σύγκριση, οι καμπύλες BER για AWGN και Rayleigh φθίνων κανάλια αναπαριστώνται μαζί. Το σχήμα 23 δείχνει τη σύγκριση. Τα διαγράμματα παρουσιάζονται για εύρος SNR 0 έως 4,5 dB. Ο κώδικας είναι σε θέση να επιτύχει πολύ χαμηλότερο BER σε μια καθορισμένη SNR σε ένα AWGN κανάλι σε σύγκριση με ένα κανάλι διάλειασης Rayleigh. Το BER περίπου 10^{-2} επιτεύχθηκε σε κανάλι διάλειασης Rayleigh με SNR 3 dB. Για ένα κανάλι με θόρυβο AWGN, το ίδιο BER επιτεύχθηκε με SNR 1.75 dB. Έτσι, υψηλότερο SNR ή μεγαλύτερος αριθμός επαναλήψεων απαιτείται για να επιτευχθεί η ίδια απόδοση του καναλιού AWGN, σε ένα κανάλι με διάλειαση Rayleigh.

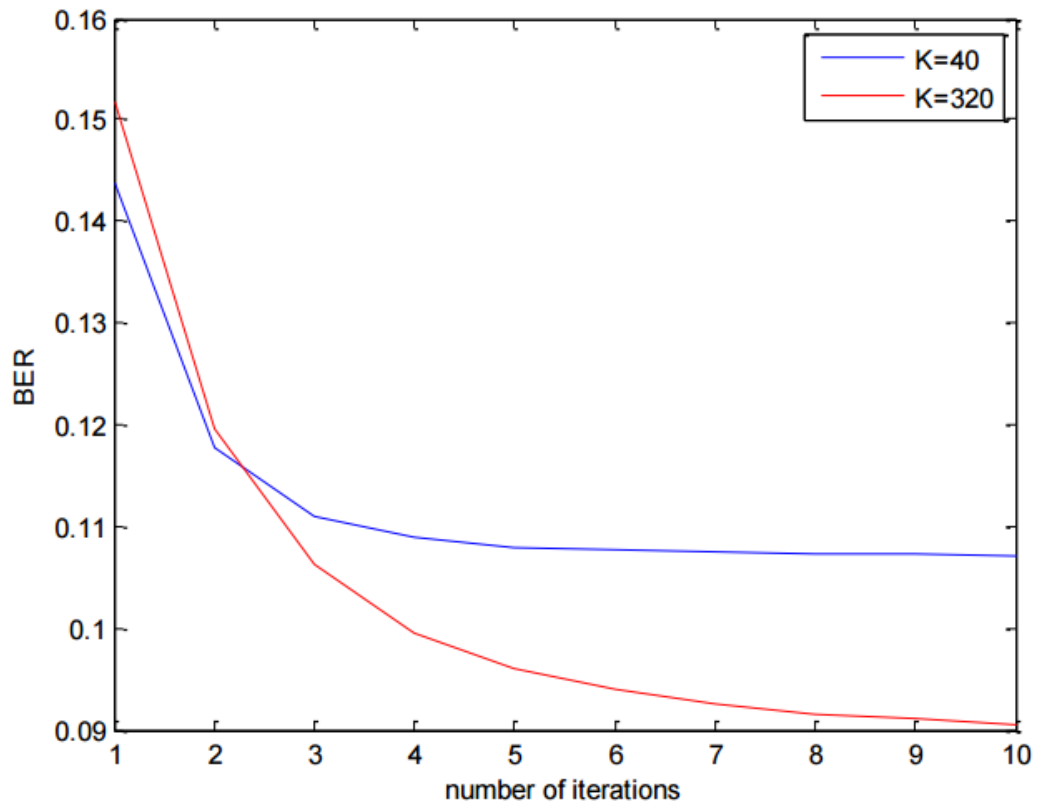


Σχήμα 23. BER για κώδικα turbo με frame size $K = 100$ μετά από 10 επαναλήψεις αποκωδικοποίησης.

5.1.7 Το BER σε συνάρτηση του αριθμού των επαναλήψεων

Η απόδοση του κώδικα turbo βελτιώνεται με την αύξηση του αριθμού των επαναλήψεων. Ωστόσο, μετά από ένα συγκεκριμένο αριθμό επαναλήψεων, το BER παραμένει σταθερό και δεν μειώνεται περαιτέρω. Για την επίτευξη χαμηλότερων BER είτε το SNR ή το μέγεθος του πλαισίου (frame size) K πρέπει να αλλάξει. Με την αύξηση του SNR, η ένταση του σήματος αυξάνεται. Ως αποτέλεσμα, η εκ των προτέρων πληροφορία (a priori information) των διαθέσιμων δεδομένων βελτιώνονται και αυτό οδηγεί σε καλύτερη αποκωδικοποίηση των δεδομένων. Με αυτό τον τρόπο μπορεί να επιτευχθεί χαμηλότερο BER. Ομοίως, όταν το μέγεθος του πλαισίου K αυξάνεται, ο διεμπλοκείας προσθέτει περισσότερη

τυχαιότητα στα δεδομένα. Αυτό οδηγεί επίσης σε καλύτερη αποκωδικοποίηση και χαμηλότερο BER.



Σχήμα 24. Η εξέλιξη του BER ως συνάρτηση του αριθμού των επαναλήψεων χρησιμοποιώντας log MAP αλγόριθμο.

Το σχήμα 24 δείχνει την εξέλιξη του BER ως συνάρτηση του αριθμού των επαναλήψεων με SNR 0 dB. Μπορεί να φανεί ότι για μέγεθος πλαισίου $K = 40$, το BER είναι πολύ υψηλότερο σε σύγκριση με το μέγεθος πλαισίου $K = 320$. Αυτές οι γραφικές παραστάσεις δείχνουν ότι ο αποκωδικοποιητής είναι σωστά υπολοισμένος καθώς το BER διαφοροποιείται μετά από κάθε επανάληψη. Επίσης, το BER μειώνεται εκθετικά και προσεγγίζει μια πολύ χαμηλή τιμή, καθώς ο αριθμός των επαναλήψεων αυξάνεται.

5.2 Ανάλυση εξόδου διαγράμματος.

Το διάγραμμα BER για επαναληπτική αποκωδικοποίηση έχει τρία μέρη:

- Η περιοχή με αμελητέα μείωση του BER.
- Η απότομη περιοχή («γκρεμος») με την επίμονη μείωση του BER για πολλές επαναλήψεις.
- Το ομαλό («τραπέζι») BER, όπου μπορεί να επιτευχθεί χαμηλό BER μετά από μόλις λίγες επαναλήψεις.

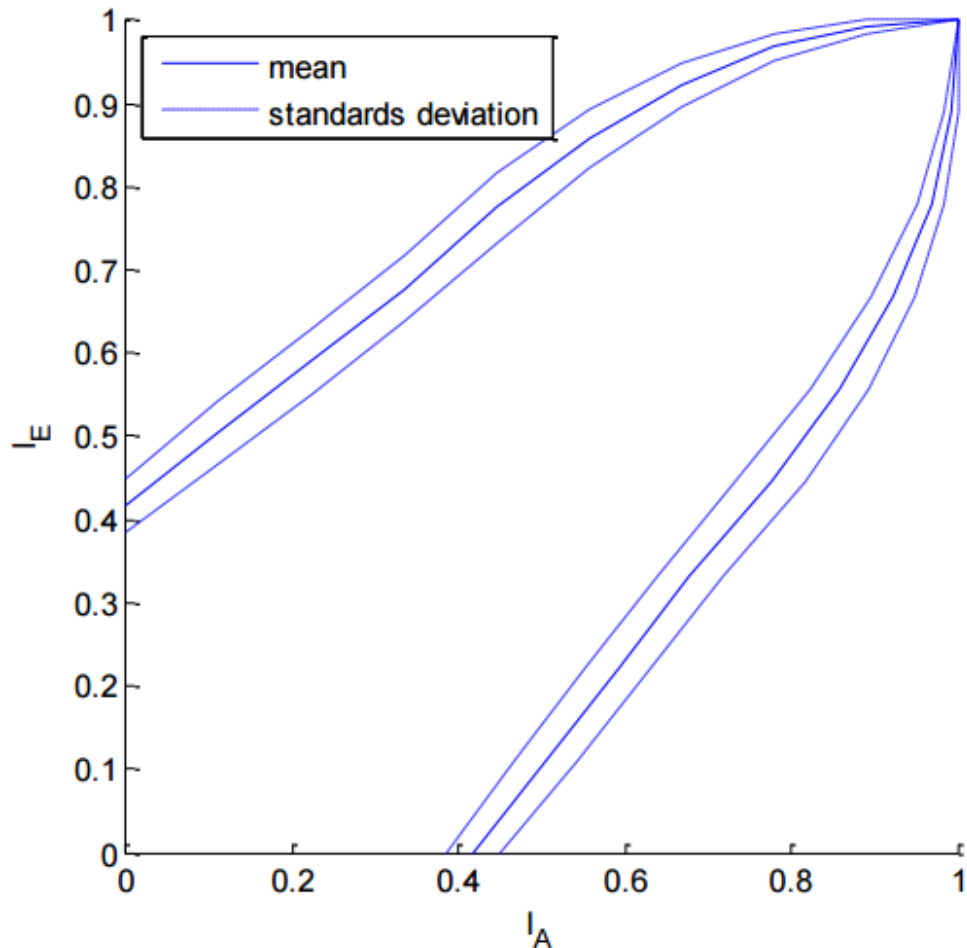
Αν και χρησιμοποιούνται τεχνικές οριοθέτησης για να τερματιστεί ο κώδικας και να αποφευχθούν οι επιπλέον υπολογισμοί, ακόμα η ανάλυση διόρθωσης σφαλμάτων (BER) δεν είναι αρκετή για την ανάλυση των επιδόσεων του κώδικα. Αυτό συμβαίνει επειδή οι τεχνικές οριοθέτησης δεν είναι οι βέλτιστες και έχουν κάποιους περιορισμούς.

Το διάγραμμα εξόδου χρησιμοποιείται για να απεικονίσει τη ροή της μεταφοράς πληροφοριών μέσω των συστατικών αποκωδικοποιητών. Είναι χρήσιμο για την ανάλυση των επιδόσεων του αποκωδικοποιητή turbo στις χαμηλές περιοχές BER όπου η καμπύλη BER έμενε σταθερή. Για να γίνει αυτό, η αμοιβαία πληροφορία τοποθετείται μεταξύ της εξωγενούς πληροφορίας στην έξοδο του συστατικού αποκωδικοποιητή και το εισερχόμενο μήνυμα λαμβάνοντας υπόψιν την αμοιβαία πληροφορία, μεταξύ της εκ των προτέρων πληροφορίας (a priori) στην είσοδο του συστατικού αποκωδικοποιητή και του μηνύματος.

Για να ληφθεί υπόψη για την επαναληπτική φύση της υποβέλτιστων αλγορίθμων αποκωδικοποίησης, και των δύο αποκωδικοποιητών τα χαρακτηριστικά απεικονίζονται σε ένα ενιαίο σχήμα. Για να γίνει αυτό, στα χαρακτηριστικά του δεύτερου αποκωδικοποιητή οι άξονες ανταλλάσσονται. Το διάγραμμα αναφέρεται ως διάγραμμα EXIT.

Η ανταλλαγή των εξωγενών πληροφοριών μπορεί να απεικονιστεί ως μια τροχιά αποκωδικοποίησης μεταξύ των δύο καμπυλών. Το διάγραμμα εξόδου για το frame size $K = 320$ με SNR 0 dB παρουσιάζεται στο 25. Η προσομοίωση έτρεξε για 100 καρέ(frames) και, στη συνέχεια, απεικονίζεται η μέση εξωγενή πληροφορία Οι

διακεκομμένες γραμμές δείχνουν την τυπική απόκλιση της εξωγενούς πληροφορίας σχετικά με τη μέση τιμή.

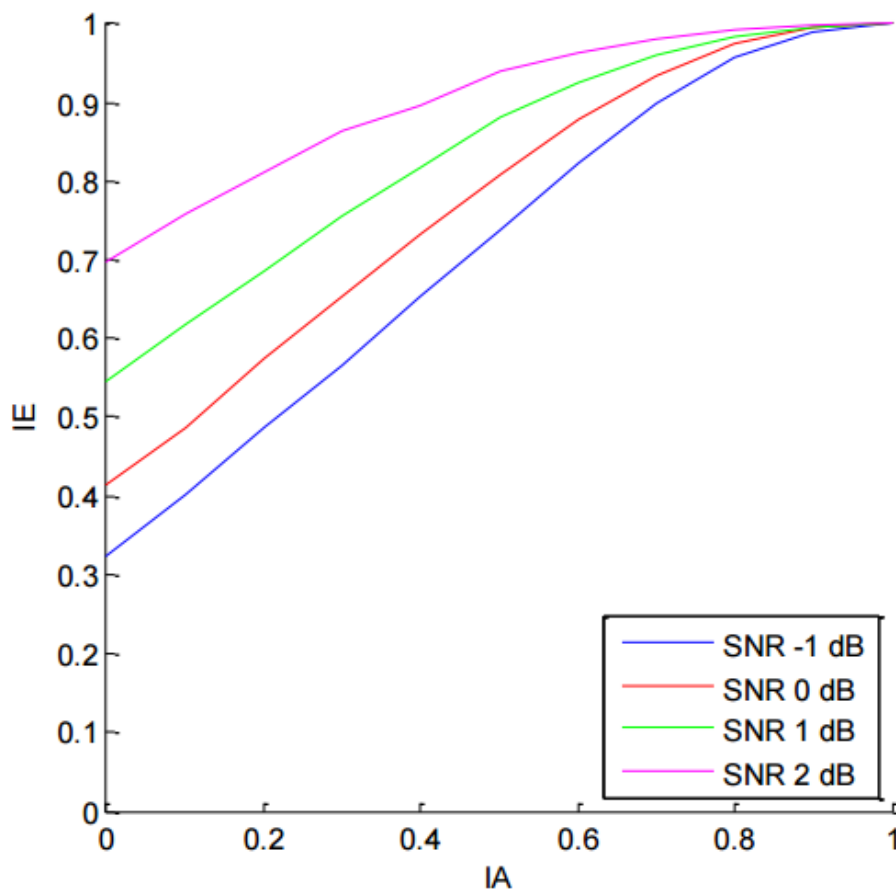


Σχήμα 25. Διάγραμμα εξόδου που παρουσιάζει τη μέση και την κανονική απόκλιση του κώδικα turbo.

5.2.1 Εξωγενείς πληροφορίες ως συνάρτηση των εκ των προτέρων πληροφοριών (a priori information).

Οι εξωγενείς πληροφορίες απεικονίζονται σε συνάρτηση με τις εκ των προτέρων πληροφορίες για διαφορετικές τιμές SNR στο σχήμα 26. Μπορεί να φανεί ότι καθώς το SNR αυξάνεται, οι εξωγενείς πληροφορίες (στήλη I_e) αυξάνονται. Ως αποτέλεσμα, παίρνουμε υψηλότερες τιμές του εξωγενών πληροφοριών (στήλη I_e) ακόμα και όταν

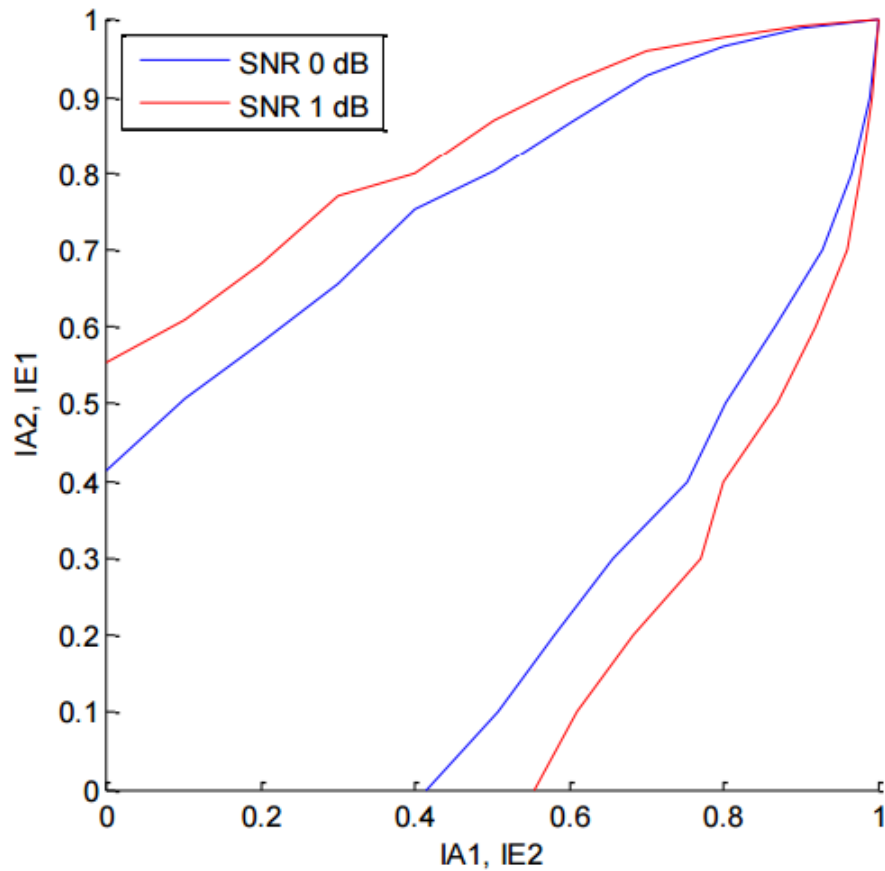
το διάγραμμα στην στήλη Ia είναι μηδενικό. Όταν το κανάλι είναι καλό ,για παράδειγμα με υψηλό SNR,τα δεδομένα είναι λιγότερο κατεστραμμένα αφού περάσουν μέσα από το κανάλι. Καθώς η είσοδος στον αποκωδικοποιητή τώρα περιέχει λιγότερα σφάλματα, μέχρι ακόμη και αν οι πληροφορίες εκ των προτέρων (a priori) είναι 0, ο αποκωδικοποιητής μπορεί ακόμα να εξάγει ένα μήνυμα που να έχει νόημα. Έτσι, ο αποκωδικοποιητής είναι ικανός να αποδίδει καλύτερα με χαμηλότερες τιμές BER. Αντίθετα, όταν το κανάλι είναι πιο θορυβώδεις, ο αποκωδικοποιητής αποτυγχάνει να υπολογίσει το μήνυμα χωρίς επαρκής αποδοτικές εκ των προτέρων πληροφορίας.



Σχήμα 26. Εξωγενείς πληροφορίες εκφραζόμενες ως συνάρτηση των εκ των προτέρων γνωστών πληροφοριών.

Το διάγραμμα εξόδου για SNR 0 και 1 dB φαίνεται παρακάτω στο σχήμα 27. Το σχήμα δείχνει ότι για SNR 1 dB η εξωγενής πληροφορία είναι υψηλότερη σε

σύγκριση με μία εκ των προτέρων πληροφορία. Έτσι, όπως συζητήθηκε νωρίτερα, ο αποκωδικοποιητής είναι σε καλύτερη θέση να αποκωδικοποιήσει τα δεδομένα.

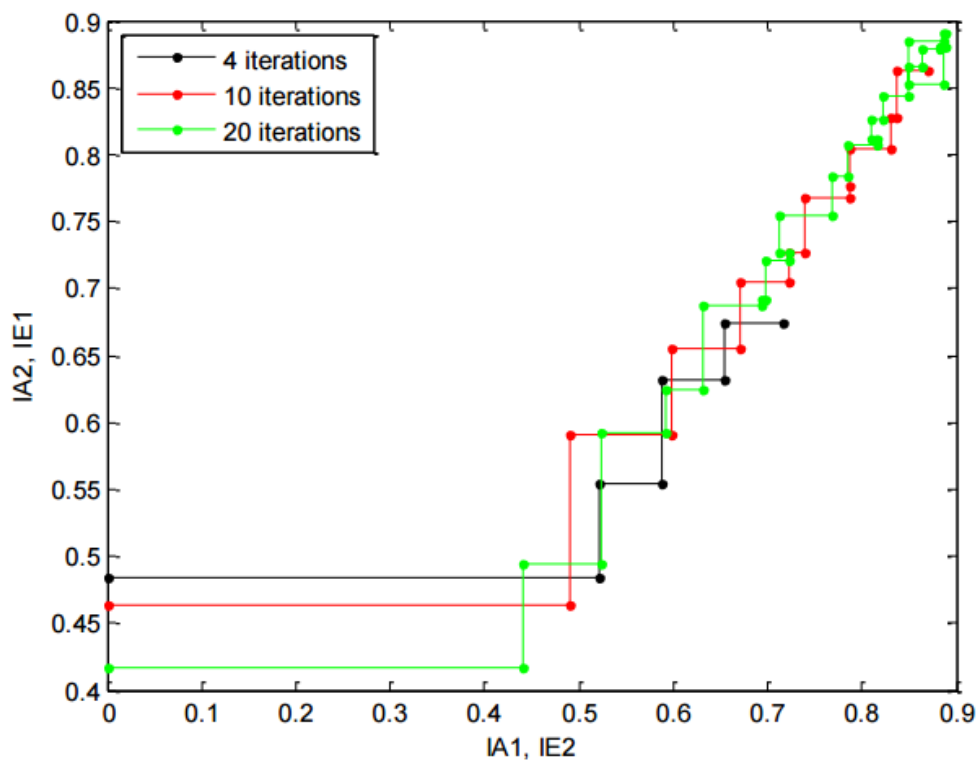


Σχήμα 27. Η πορεία μίας εξωγενούς πληροφορίας ως συνάρτηση του SNR.

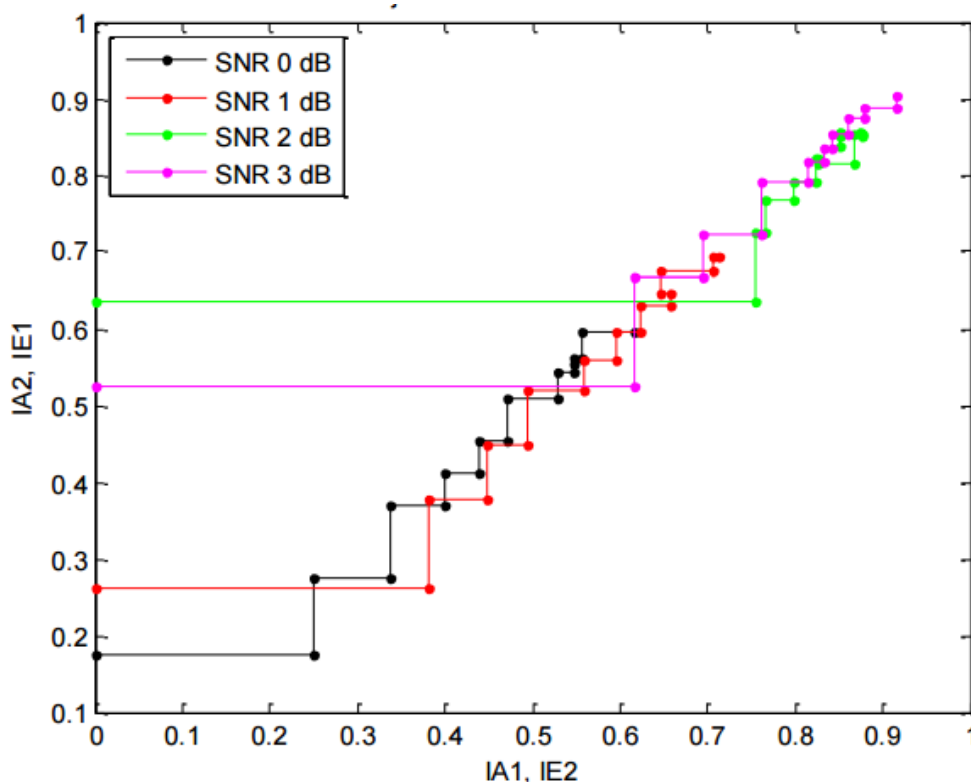
5.2.2 Τροχιά αποκωδικοποίησης

Η τροχιά αποκωδικοποίησης δίνει την οπτικοποίηση της ανταλλαγής των εξωγενών πληροφοριών στο διάγραμμα εξόδου. Η τροχιά του αποκωδικοποιητή για μέγεθος πλαισίου (frame size-καρέ) $K = 40$ κώδικα turbo απεικονίζεται στο Σχήμα 28. Μπορεί να φανεί ότι με 4 επαναλήψεις, η τροχιά της εξωγενούς πληροφορίας και της εκ των προτέρων πληροφορίας φθάνει περίπου στο σημείο (0.7, 0.65) στο γράφημα. Καθώς

ο αριθμός των επαναλήψεων αυξάνεται από 4 σε 10, η τροχιά φθάνει περίπου στο σημείο (0.85, 0.85). Αυξάνοντας περαιτέρω τις επαναλήψεις αποκωδικοποιήσεις σε 20, δεν υπάρχει μεγάλη βελτίωση στην εκ των προτέρων και στην εξωγενή πληροφορία. Μια αύξηση 10 επαναλήψεων αποκωδικοποίησης οδηγεί μόνο σε μια πολύ μικρή βελτίωση στην εξωγενή πληροφορία και η τροχιά δεν κινείται περαιτέρω προς το σημείο(1, 1). Δείχνει μάλλον να ταλαντώνεται γύρω από το ίδιο σημείο. Έτσι, οι επαναλήψεις πρέπει να επιλέγονται έχοντας κατά νου την άμεση αντιστάθμιση μεταξύ πολυπλοκότητας και απόδοσης του συστήματος.



Σχήμα 28. Τροχιά αποκωδικοποίησης για frame size $K = 40$ για διάφορους αριθμούς επαναλήψεων.

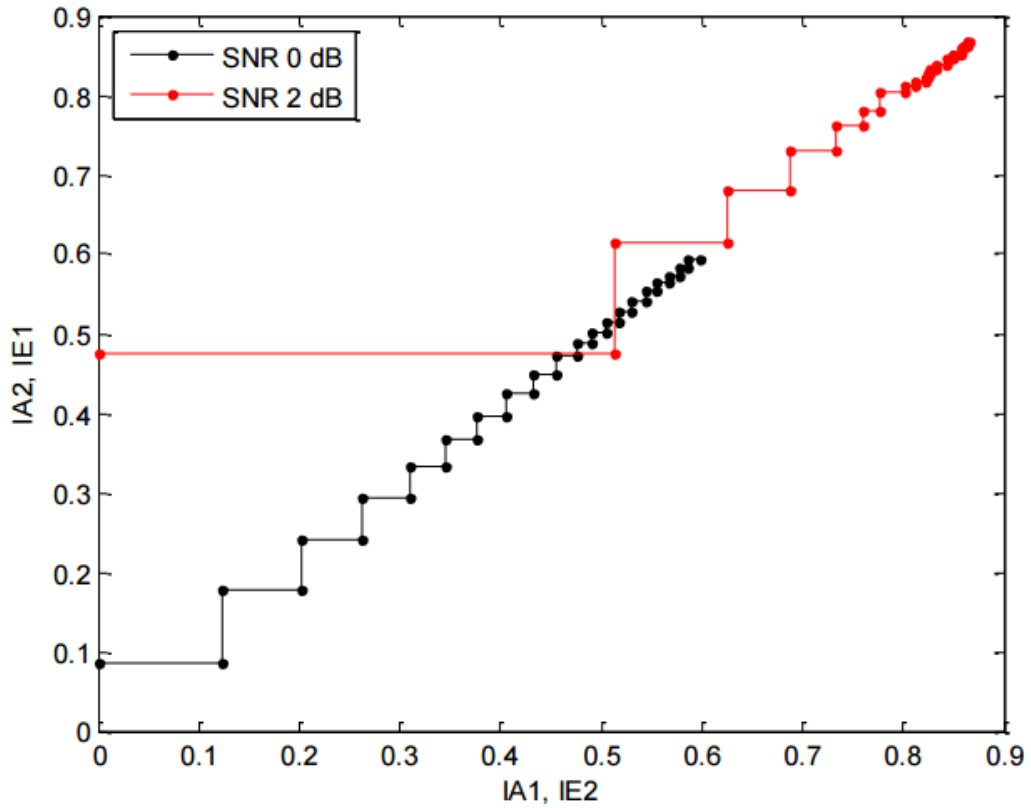


Σχήμα 29. Τροχιά αποκωδικοποίησης για frame size $K = 40$ κώδικα turbo μετά από 10 επαναλήψεις με διαφορετικά SNR.

Οι πορείες-τροχιές αποκωδικοποίησης για καρέ (frame size) $K = 40$ μετά από 10 επαναλήψεις αποκωδικοποίησης για διαφορετικά SNR φαίνεται στο Σχήμα 29. Καθώς το SNR αυξάνεται, οι a priori πληροφορίες στον αποκωδικοποιητή 1 αυξάνεται. Αυτό οδηγεί επίσης σε αυξημένες εξωγενείς πληροφορίες. Υπάρχει μία βελτίωση της απόδοσης του συστήματος όπως φαίνεται από τη βελτίωση στις εξωγενείς πληροφορίες 2 στο τέλος των επαναλήψεων. Δεδομένου ότι το SNR αυξάνεται από 0 σε 2 dB, οι εξωγενείς πληροφορίες του αποκωδικοποιητή 2 στο τέλος της 10ης επανάληψης βελτιώνονται σημαντικά από 0,65 σε 0,9. Σε περαιτέρω αύξηση του SNR από 2 dB στα 3 dB, οι εξωγενείς πληροφορίες δείχνουν πολύ μικρή βελτίωση. Η τροχιά αποκωδικοποίησης για μέγεθος καρέ $K = 320$ μετά από 20 επαναλήψεις αποκωδικοποίησης δείχνουν παρόμοια αποτελέσματα όπως φαίνεται στο Σχήμα 30.

Μπορεί να γίνει άμεσα αντιληπτό ότι με την αύξηση του SNR από 0 σε 2 dB, οι εξωγενείς πληροφορίες μετά από 20 επαναλήψεις βελτιώνονται από 0,65 σε 0,85

περίπου. Ωστόσο, οι περαιτέρω επαναλήψεις δεν δείχνουν μεγάλη βελτίωση στην απόδοση. Έτσι, υπάρχει ένας συμβιβασμός μεταξύ ενεργειακής απόδοσης, πολυπλοκότητας και απόδοσης του συστήματος.



Σχήμα 30. Πορεία-τροχιά αποκωδικοποίησης για frame size $K = 320$ κώδικα turbo μετά από 20 επαναλήψεις αποκωδικοποίησης.

6.1 Περίληψη

Οι κώδικες Turbo είναι μια κατηγορία υψηλής απόδοσης κώδικων διόρθωσης σφάλματος με απόδοση σφάλματος κοντά στο όριο του Shannon. Κάνουν χρήση της παράλληλα συνεχόμενων συνελκτικών-αναδρομικών κώδικων για να παρέχουν αρκετή προστασία σφάλματος με λογική πολυπλοκότητα. Ο βασικός κωδικοποιητής κώδικα turbo αποτελείται από δύο RSC κωδικοποιητές συνδεδεμένους παράλληλα. Οι κωδικοποιητές RSC παράγουν κωδικές λέξεις υψηλότερου βάρους σε σύγκριση με τους μη RSC κωδικοποιητές και έτσι είναι καλύτερα προσαρμοσμένοι για τους κώδικες turbo. Ένας διεμπλοκείας χρησιμοποιείται μεταξύ των δύο κωδικοποιητών για να εισαγάγει την τυχαιότητα καθώς οι τυχαίοι κώδικες επιτυγχάνουν καλύτερη χωρητικότητα καναλιού. Η έξοδος του κωδικοποιητή αποτελείται από συστηματικά bits, καθώς και κωδικοποιημένα bits από τους δύο κωδικοποιητές.

Ο αποκωδικοποιητής λαμβάνει υπόψη του τη εισερχόμενη κωδική λέξη και μαζί με τη γνώση της δομής του κώδικα υπολογίζει τις LLRs. Η αποκωδικοποίηση πραγματοποιείται με ένα επαναληπτικό τρόπο με τις πληροφορίες να ανταλλάσσονται μεταξύ των συστατικών κωδικοποιητών για καλύτερη απόδοση. Ο αποκωδικοποιητής υπολογίζει πρώτα τις μετρήσεις κλάδου για όλους τους κόμβους του χωροδικτύωματος. Το επόμενο βήμα είναι να υπολογιστεί η αντίστροφη διαδρομή για το λαμβανόμενο πλαίσιο δεδομένων και το αποτέλεσμα αποθηκεύεται στη μνήμη. Μετά από αυτό, οι προς τα εμπρός διαδρομές μετρήσεων υπολογίζονται. Στο τέλος οι αναλογίες πιθανοφάνειας (log-likelihood ratios- LLRs) των δεδομένων για το σύνολο του πλαισίου υπολογίζεται και οι πληροφορίες διαβιβάζονται στον άλλο αποκωδικοποιητή. Μετά το τέλος των επαναλήψεων, μια δύσκολη απόφαση λαμβάνεται με τις LLRs όσον αφορά τα ληφθέντα bits.

Ο αποκωδικοποιητής RSC χρησιμοποιεί ένα τροποποιημένο αλγόριθμο (Jacobi) για τις μεταβάσεις μέσα στο χωροδικτύωμα. Τέσσερις κύριες εκδοχές αλγόριθμου χρησιμοποιούνται κυρίως στις εφαρμογές, log MAP, max-log MAP, constant-log MAP (σταθερής καταγραφής), linear-log MAP (γραμμικής καταγραφής). Η σύγκριση των επιδόσεων των αλγόριθμων έγινε σύμφωνα και με τις διαφορές στις καμπύλες

των BER. Ο αλγόριθμος υλοποιείται δύο φορές σε κάθε μισή επανάληψη, και έτσι αποτελεί ένα σημαντικό τμήμα της του πόσο πολύπλοκη γίνεται η αποκωδικοποίηση.

6.2 Συμπεράσματα

Οι κώδικες Turbo χρησιμοποιούνται και σε 3G και σε 4G συστήματα κινητής τηλεφωνίας, WiMAX και δορυφορικά συστήματα επικοινωνίας. Δείχνουν εξαιρετικές επιδόσεις σε χαμηλά SNRs και πλησιάζουν το όριο του Shannon. Ωστόσο, σε υψηλότερα SNRs, η καμπύλη BER αρχίζει να γίνεται επίπεδη και γίνεται αντιληπτή η αδυναμία να επιτευχθούν εξαιρετικά μικρά ποσοστά σφάλματος bit. Τα αποτελέσματα της προσομοίωσης έδειξαν ότι οι αποδόσεις των turbo codes εξαρτώνται από έναν αριθμό παραμέτρων, συμπεριλαμβανομένου του μεγέθους του πλαισίου K , τον αριθμό των επαναλήψεων του αποκωδικοποιητή, το SNR και την επιλογή του αλγορίθμου Log MAP που θα χρησιμοποιηθεί.

Οι κώδικες Turbo είναι επαναληπτικοί κώδικες και η απόδοση τους βελτιώνεται με την αύξηση του αριθμού των επαναλήψεων. Τυπικά 5 έως 10 επαναλήψεις παράγουν το μεγαλύτερο μέρος της βελτίωσης. Στα αποτελέσματα που πήραμε, το BER μειώθηκε σημαντικά από την πρώτη έως την πέμπτη επανάληψη. Ωστόσο, δεν υπήρχε μεγάλη βελτίωση από την πέμπτη έως την δέκατη επανάληψη. Έτσι, μπορούμε να συμπεράνουμε ότι καθώς ο αριθμός των επαναλήψεων αυξάνεται, το ποσοστό της απόδοσης βελτίωση μειώνεται. Για την επίτευξη χαμηλότερων BER είτε τα SNR ή το frame size K πρέπει να αλλάξει.

Όταν η ισχύς του σήματος είναι αυξημένη, το SNR αυξάνεται. Ως αποτέλεσμα αυτού, οι εκ των προτέρων πληροφορίες των διαθέσιμων στοιχείων βελτιώνονται. Καθώς η είσοδος στον αποκωδικοποιητή περιέχει τώρα λιγότερα λάθη, ο αποκωδικοποιητής μπορεί ακόμα να εξάγει ένα μήνυμα που να έχει νόημα ενώ παράλληλα χαμηλότερο BER μπορεί να επιτευχθεί.

Η απόδοση του κώδικα turbo εξαρτάται έντονα από το πόσο τυχαία είναι τα δεδομένα που θα εισαχθούν από τον διεμπλοκέα. Ένας μεγαλύτερος διεμπλοκέας μεγέθους K έχει ως αποτέλεσμα υψηλότερη πιθανότητα κωδικών λέξεων με μεγάλο βάρος, οδηγώντας έτσι σε καλύτερη απόδοση αποκωδικοποίησης. Αυτό φάνηκε στη προσομοίωση όπου το BER μετά από 10 επαναλήψεις αποκωδικοποίησης έπεσε από 10^{-2} σε 10^{-5} , αυξάνοντας το μέγεθος του πλαισίου K από 40 στα 320. Έτσι, αυξάνοντας το μέγεθος του πλαισίου K , μπορεί να επιτευχθεί χαμηλότερο BER με τη διατήρηση της σταθεράς SNR αλλά με το κόστος της αυξημένης λανθάνουσας κατάστασης. Επίσης, ο κώδικας είναι σε θέση να επιτύχει πολύ χαμηλότερο BER σε προκαθορισμένο SNR σε ένα κανάλι με θόρυβο AWGN παρα σε σύγκριση με ένα κανάλι με διάλειαση Rayleigh.

Οι τέσσερις παραλλαγές του κλασσικού αλγορίθμου log MAP εφαρμόστηκαν. Έγινε προφανές ότι η απόδοση του αλγορίθμου max-log MAP ήταν σημαντικά χειρότερη από τα τέσσερις αλγόριθμους και ο αλγόριθμος log MAP είχε την καλύτερη απόδοση. Ωστόσο, ο max-log MAP αλγόριθμος έχει τη λιγότερη πολυπλοκότητα και χρειάζεται το λιγότερο χρόνο για την προσομοίωση. Έτσι, υπάρχει μια άμεση αμφίδρομη σχέση μεταξύ πολυπλοκότητας και απόδοσης. Παρά το γεγονός ότι ο αλγόριθμος log MAP είναι περισσότερο πολύπλοκος, μπορεί να επιτύχει παρόμοια απόδοση BER με εκείνη του αλγορίθμου max-log MAP χρησιμοποιώντας λιγότερες επαναλήψεις.

Έτσι, μπορούμε να συμπεράνουμε την αναγκαιότητα να χρησιμοποιούμε Turbo κώδικες για τις 3G και 4G επικοινωνίες καθώς μας παρέχουν πολλά παραπάνω από όλες τις προηγούμενες μορφές κωδικοποίησης και αποκωδικοποίησης δεδομένων και χρησιμοποιώντας όποια συγκεκριμένη κατηγορία αλγόριθμου επιλέξουμε, δίνουμε βάρος στο τομέα που αποζητούμε, είτε είναι η ενεργειακή αποδοτικότητα, το εύρος ζώνης της αποδοτικότητας, η λανθάνουσα κατάσταση, η πολυπλοκότητα ή η απόδοση σφάλματος BER.

ΒΙΒΛΙΟΓΡΑΦΙΑ

1. A CMI design for turbo codes"3GPP TS 25.212, Ver.3.4.0 J. Yuan Feng, W. and B" Vucetic, 2002
2. Weight distribution analysis for proposed asymmetric turbo code for improved performance" 3GPP2, 2002 K and M U. Siddiqi C. Ramasamy
3. "A 188-size 2.1mm² reconfigurable turbo decoder chip with parallel architecture for 3GPP LTE system," June 2009 H.-C. C. Y.-Yu Lee and C.-C. Wong
4. A 150MBit/s 3GPP LTE turbo code decoder," Germany, Mar. 2010 T. Inseher, W. Raab, M. May and N. When
5. "ARP and QPP interleavers for LTE turbo coding," USA, Mar. 2008 Y. Blankenship, A. Nimbalkar, T. K. Blankenship and B. Classon
6. A 24Mb/s radix-4 logMAP turbo decoder for 3GPP-HSDPA mobile wireless," USA, Feb. 2003 M. Bickerstaff, C. Nicol, L. Davis, C. Thomas and D. Garrett
7. "FPGA implementation of a 3GPP turbo codec," 4-7 Nov. 2001 C. Dick and J. Steensma
8. "A Programmable Turbo Decoder for Multiple 3G Wireless Standards," February 11, 2003 Myoung-Cheol Shin and In-Cheol Park
9. "Universal Mobile Telecommunications System (UMTS): Multiplexing and Channel Coding (FDD)," 3 GPP TS 125.212 Version 3.4.0 23 September, 2000 European Telecommunications Standards Institute (ETSI)
10. "Physical Layer Standard for cdma2000 Spread Spectrum Systems, Release C," 3GPP2 C.S0002-C, Version 1.0 May 28, 2002 Third Generation Partnership
11. Layered Adaptive Modulation and Coding for 4G Wireless Networks 2010 H. Ji, J. Mei, J. She, J. Ho and Pin-Han Ho
12. 3G and 4G Cellular Standards November 5, 2008 Prof. Jeffrey G. Andrews
13. "LTE for 4G Mobile Broadband" Farooq Khan

14. "Design and ASIC Implementation of a 3GPP LTE- Advance Turbo Encoder and Turbo Decoder July-August 2012 Manjunatha K N , Prasanna Kumar and Kiran B
15. "3GPP LTE/SAE" 5 October, 2010 S M Chadchan Member IEEE and C. B.Akki
16. "Evaluation of Soft Output Decoding for Turbo Codes" 1997 Fu-hua Huang
17. "Psuedo-Random recursive convolutional coding for near-capacity performance" Dec. 1993, C. Berrou, A. Glavieux and G. Battail
18. "Unveiling turbo codes: some results on parallel concatenated coding schemes," March 1996, G. Montorsi and S. Benedetto
19. "Performance evaluation of turbo codes for short frame transmission systems", Jan. 20, 1994, M. Nasshan and P. Jung
20. "Turbo code termination and inteleaver conditions" , November 1995, E. K. Hall, S. G. Wilson and W. J. Blackhert
21. "Terminating the trellis of turbo codes" August 1994, H. Meyr and O. Joerssen
22. "A distance spectrum interpretation of turbo codes" November 1996, L. Perez, D. J. C. Jr. and J. Sephers "Multilevel turbo coding with short interleavers" February 1998, H. Herzberg
23. "A turbo code interleaver design criterion based on the performance of iterative decoding" May 2000, J. Hokfelt
24. "Illuminating the structure of code and decoder of parallel concatenated recursive systematic turbo codes" Dec. 1994, P. Robertson
25. "A mathematical theory of communication," 1948 C. E. Shannon
26. "Error bound for convolutional codes and asymptotically opti-mum decoding algorithm," Apr. 1967, A. J. Viterbi
27. "Low-density parity check codes," 1963, R. Gallager
28. "An upper bound on turbo codes performance over quasi-static fading channels," July 2003, S. L. Miller and H. Bouzekri
29. "Iterative decoding of binary block and convolutional codes," Mar. 1996, L. Papke, J. Hagenauer and E. Offer
30. "Convolutional codes and their performance in communication systems," Oct. 1971, A. J. Viterbi
31. "Near optimum error correcting coding and decod-ing: Turbo codes," Oct.

- 1996, A. Glavieux and C. Berrou
32. "Concatenated convolutional codes and iterative decoding," 2003, W. E. Ryan
33. "Design and analysis of turbo codes on rayleigh fading channels," Feb. 1998, S. G. Wilson and E. K. Hall