Ονοματεπώνυμο: Δοϊνάκης Μιχαήλ  
ΑΕΜ: 9292  
e-mail: [doinakis@ece.auth.gr](mailto:doinakis@ece.auth.gr)

Εισαγωγή

Στόχος της παρούσας εργασίας είναι η υλοποίηση ενός κωδικοποιητή/αποκωδικοποιητή ήχου κατά το πρότυπο Advanced Audio Coding, με τις απλουστεύσεις που αναφέρονται στην εκφώνηση της εργασίας.

Η αναφορά χωρίζεται σε τρία μέρη, ακριβώς όπως και η εργασία. Το πρώτο κομμάτι πραγματεύεται την υλοποίηση της βαθμίδας Sequence Segmentation Control (SSC),τη βαθμίδα filterbank. Το δεύτερο κομμάτι της εργασίας ασχολείται με την υλοποίηση της βαθμίδας Temporal Noise Shaping. Τέλος, το τρίτο κομμάτι υλοποιεί το απλουστευμένο ψυχοακουστικό μοντέλο που παρουσιάζεται στην εκφώνηση της εργασίας. Για όλες τις βαθμίδες που υλοποιούνται στα τρία μέρη της εργασίας υλοποιούνται και οι αντίστροφες αυτών για να δημιουργηθεί τελικά το σύστημα κωδικοποίησης αποκωδικοποίησης του αρχικού σήματος. Στα πλαίσια της εργασίας θεωρήθηκε ότι το σήμα εισόδου είναι δικαναλικό με συχνότητα δειγματοληψίας 48 kHz.

Επιπλέον, για κάθε ένα από τα τρία μέρη, το σήμα χωρίζεται γεμίζει με 1024 μηδενικά στην αρχή και στο τέλος του και έπειτα σε frames μήκους 2048 τα οποία επικαλύπτονται κατά 50% μεταξύ τους. Το zero pad στην αρχή και το βοηθά για να μη χαθεί η πληροφορία που υπάρχει στα 2 άκρα του αρχικού και του τελικού frame.

Μέρος πρώτο

Στο πρώτο μέρος, όπως αναφέρθηκε και παραπάνω, υλοποιούνται οι βαθμίδες SSC, filterbank και ifilterbank.

Sequence Segmentation Control

Για την υλοποίηση της βαθμίδας SSC δημιουργήθηκε η ομώνυμη συνάρτηση η οποία δέχεται ως ορίσματα το τρέχον frame τον τύπο του προηγούμενου frame και το επόμενο frame με στόχο να αποφασίσει τον τύπο του τρέχοντος frame. Για την υλοποίηση ακολουθήθηκε ο αλγόριθμος που αναφέρεται στην εκφώνηση και υπάρχουν αναλυτικά σχόλια στον κώδικα για την επεξήγηση του αλγορίθμου. Αυτό που πρέπει να σημειωθεί είναι πως ο αλγόριθμος της εκφώνησης δεν χρησιμοποιεί κάπως τα δείγματα τους τρέχοντος frame, για την εξαγωγή του τύπου του, παρά μόνο τον τύπο του προηγούμενου και τα δείγματα του επόμενου frame.

Filterbank και iFiltrebank

Αφού γίνει η κατηγοριοποίηση όλων των frame από τη βαθμίδα SSC, καλείται η βαθμίδα filterbank η οποία έχει ως στόχο την αποσυχέτιση των δειγμάτων καθώς μεταβαίνει και στο πεδίο της συχνότητας εφαρμόζοντας στο εκάστοτε frame Modified Discrete Cosine Transform. Ο λόγος που χρησιμοποιείται ο MDCT είναι γιατί αποφεύγει τα προβλήματα που εμφανίζονται στα όρια των παραθύρων (frames) όπως αυτά που μελετά η εργασία. Πριν την εφαρμογή του MDCT επιλέγεται ένα παράθυρο το οποίο πολλαπλασιάζεται με τα δείγματα στο χρόνο έτσι ώστε να είναι δυνατή η ανακατασκευή του σήματος στον αποκωδικοποιητή. Υπάρχει δυνατότητα επιλογής παραθύρων Kaiser-Bessel Derived και Sinusoid. Επίσης, γίνεται η υπόθεση ότι το παράθυρο των frames επιλέγεται στη συνάρτηση AACoder1 και είναι το ίδιο για όλα. Η μορφή του παραθύρου εξαρτάται από το είδος του εκάστοτε frame. Όμοια, η αντίστροφη διαδικασία γίνεται από την συνάρτηση ifilterbank με τη διαφορά ότι το frame πολλαπλασιάζεται με το αντίστοιχο παράθυρο αφού πρώτα εφαρμοστεί ο αντίστροφος MDCT. Παράλληλα, όταν το frame είναι eight short, η ανακατασκευή του γίνεται zero pad στα πρώτα και τελευταία 448 δείγματά του και τα υπόλοιπα 1152 προέρχονται από τον πολλαπλασιασμό με το παράθυρο και την πρόσθεση των δειγμάτων μεταξύ τους όπου υπάρχουν επικαλύψεις (τα subframes επικαλύπτονται κατά 50% το ένα με το άλλο όπως και τα κανονικά frames). Για τους υπόλοιπους τύπους frames μετά τον αντίστροφο MDCT το frame απλά πολλαπλασιάζεται με το παράθυρο.

Για τα frame τύπου eight short τα subframes τοποθετούνται σε έναν πίνακα 2024 επί 2 (δηλαδή 8 sub frames 256 επί 2) τοποθετημένα το ένα κάτω από το άλλο για να γίνει ο πολλαπλασιασμός τους με το παράθυρο για όλα ταυτόχρονα (για περισσότερες λεπτομέρειες αρχείο filterbank lines:76,108).

Σχόλιο: Η παράμετρος α για τα παράθυρα τέθηκε όπως στην εκφώνηση 6 για long και 4 για short παράθυρα, και όχι 4 για long και 6 για short όπως αναφέρει στο αρχείο w2203tfa.

AACoder1 και iAACoder1

Η εφαρμογή όλων των παραπάνω γίνεται με τις συναρτήσεις AACoder1 και iAACoder1. Η AACoder1(κωδικοποιητής) διαβάζει το αρχείο στο matlab εφαρμόζει το zero pad στην αρχή και το τέλος του σήματος και αφού δημιουργήσει τα frames με 50% επικάλυψη, για κάθε frame υπολογίζει τον τύπο του μέσω του SSC και εφαρμόζει τη filterbank. Για το πρώτο frame, επειδή δεν έχει προηγούμενο frame, θεωρήθηκε ότι ο τύπος του προηγούμενου frame ήταν OLS. Έτσι, δημιουργείται ένα struct το οποίο περιέχει τους συντελεστές MDCT για όλα τα frames (για αριστερό και δεξί κανάλι ήχου), τον τύπο του frame και τον τύπο του παραθύρου που εφαρμόστηκε στο πεδίο του χρόνου.

Η iAACoder1(αποκωδικοποιητής), αφού έχει έτοιμους του συντελεστές MDCT και τον τύπο του frame και του παραθύρου που εφαρμόστηκε σε αυτό, καλεί την ifilterbank, η οποία υπολογίζει τον αντίστροφο MDCT και μετά προσθέτει τις επικαλύψεις των παραθύρων μεταξύ τους προκειμένου να δημιουργηθεί ξανά το αρχικό σήμα. Επίσης, αφαιρεί και το zero pad που είχε εφαρμοστεί και στο αρχικό σήμα («πετάει» τα πρώτα 1024 δείγματα του σήματος που παράχθηκε).

demoAAC1

Η εφαρμογή όλων των βαθμίδων που αναφέρθηκαν παραπάνω δεν εισάγουν κάποιο σφάλμα στο σήμα και θεωρητικά η διαδικασία είναι πλήρως αντιστρέψιμη. Στη συνάρτηση demoAAC1 αυτό αποδεικνύεται με τον υπολογισμό του SNR. Ως SNR ορίστηκε ο λόγος του αρχικού σήματος y προς τη διαφορά του αρχικού σήματος από το τελικό σήμα, τα οποία τοποθετήθηκαν στην συνάρτηση SNR του matlab. Με βάσει τον παραπάνω ορισμό για το SNR αναμένεται, αφού η διαδικασία πρέπει να είναι πλήρως αντιστρέψιμη το SNR να βγει αρκετά μεγάλο. Πράγματι, για παράθυρα **KBD** το **SNR** βγήκε **253.9681 dB** ενώ για SIN **253.9881 dB**.

Μέρος δεύτερο

Στο δεύτερο μέρος υλοποιείται η βαθμίδα Temporal Noise Shaping η οποία έχει ως στόχο το μετασχηματισμό των συντελεστών MDCT έτσι ώστε να εξαλείψει τις περιοδικότητες που μπορεί να εμφανίζονται λόγο κάποιου έντονου ήχου ή κάποιου pitch της φωνής. Η βαθμίδα αυτή εφαρμόζεται σε συνδυασμό με τις βαθμίδες του μέρους πρώτου της εργασίας. Το TNS εφαρμόζεται μετά την εφαρμογή του filterbank, στους συντελεστές MDCT και στον αποκωδικοποιητή δίνονται οι μετασχηματισμένοι συντελεστές MDCT μαζί με τις προδιαγραφές του φίλτρου που εφαρμόστηκε για την αποσυσχέτιση (TNS coeficients).

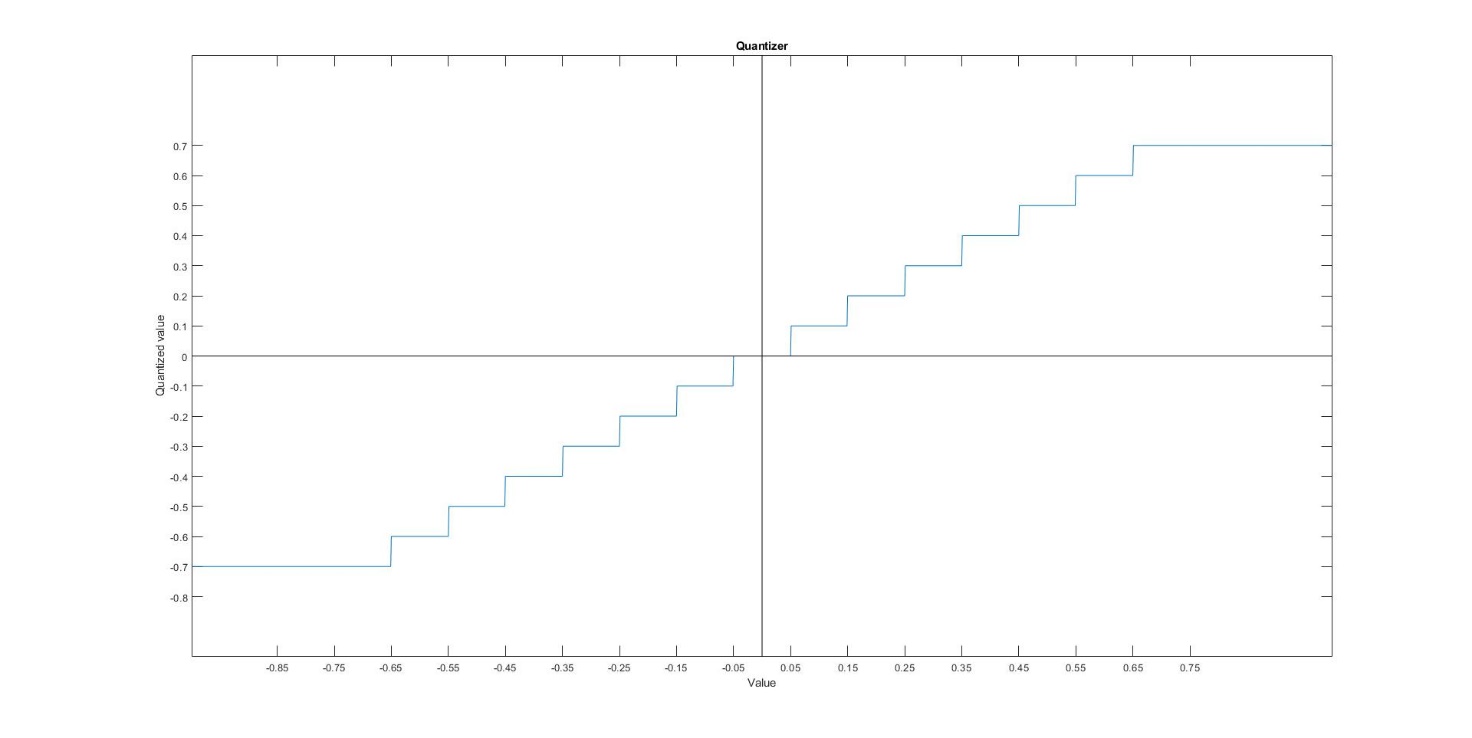
Temporal Noise Shaping

Στόχος του TNS είναι να επιστρέψει τους μετασχηματισμένους συντελεστές MDCT αλλά και τους συντελεστές TNS που υπολόγισε ώστε η διαδικασία να μπορεί να αντιστραφεί στον αποκωδικοποιητή.

Η διαδικασία ξεκινά υπολογίζοντας τις ενέργειες κάθε μπάντας, με στόχο τον υπολογισμό των συντελεστών κανονικοποίησης για κάθε συντελεστή MDCT. Αφού υπολογιστούν αυτοί, εξομαλύνονται με τη διαδικασία που περιγράφεται στην εκφώνηση. Για τον υπολογισμό των συντελεστών (για φίλτρο γραμμικής πρόβλεψης σταθερής τάξης ίση με 4) πρέπει να ελαχιστοποιούν το τετράγωνο του σφάλματος πρόβλεψης και έπειτα από επίλυση των εξισώσεων προκύπτουν οι εξής σχέσεις:

όπου , όπου r είναι οι τιμές τις συνάρτησης αυτοσυσχέτισης των συντελεστών MDCT που υπολογίστηκαν με τη βοήθεια της συνάρτησης xcorr του matlab.

Μετά τον υπολογισμό των συντελεστών α, προκειμένου η διαδικασία να είναι αντιστρέψιμη, οι συντελεστές κβαντίζονται με κβαντιστή 4 bits και βήματος 0.1 όπως φαίνεται και στο παρακάτω σχήμα. Για να είναι ο κβαντιστής συμμετρικός και ομοιόμορφος το ένα επίπεδο του παραλείπεται. Ο κβαντιστής φαίνεται στο παρακάτω σχήμα.



Αφού γίνει και ο κβαντισμός των τελεστών TNS, εφαρμόζεται το FIR φίλτρο στην αρχική ακολουθία δειγμάτων συντελεστών MDCT. Για τον έλεγχο της ευστάθειας του αντίστροφου φίλτρου χρησιμοποιήθηκε η συνάρτηση isstable του matlab και όλα τα αντίστροφα φίλτρα ήταν ευσταθή. Επίσης, σε περίπτωση που το frame είναι eight short υπολογίζονται συντελεστές TNS για όλα τα subframes. Η αντίστροφη διαδικασία εφαρμόζεται στο iTNS και είναι η απλή εφαρμογή του αντίστροφου φίλτρου στους μετασχηματισμένους συντελεστές.

demoAAC2

Όπως στο πρώτο μέρος έτσι και στο δεύτερο υλοποιούνται οι συναρτήσεις AACoder2 και iAACoder2 οι οποίες είναι παρόμοιες με αυτές του πρώτου μέρους με τη διαφορά ότι τώρα μετά την εφαρμογή της filterbank στο AACoder2 εφαρμόζεται η TNS σε κάθε frame για να μειώσει την περιοδικότητα και στον αποκωδικοποιητή (iAACoder2) πριν το ifilterbank εφαρμόζεται η iTNS. Επιπλέον, και η διαδικασία εφαρμογής του TNS και iTNS δεν πρέπει να προκαλέσει αλλοίωση στη ποιότητα του σήματος και η διαδικασία πρέπει να είναι πλήρως αντιστρέψιμη, όπως και στο μέρος πρώτο. Έχοντας τον ίδιο ορισμό για το SNR όπως και παραπάνω παίρνουμε για παράθυρα KBD και SIN **253.9680 dB** και **253.9881 dB** αντίστοιχα. Συνεπώς ο σηματοθορυβικός λόγος είναι αρκετά μεγάλος και δεν υπάρχει αλλοίωση στην πληροφορία του σήματος.

Μέρος τρίτο

Στο τρίο κομμάτι της εργασίας υλοποιείται το ψυχοακουστικό μοντέλο, με το οποίο επιτυγχάνεται και η συμπίεση του αρχείου ήχου. Επιπλέον, χρησιμοποιούνται και οι βαθμίδες που υλοποιήθηκαν στα δύο προηγούμενα μέρη τις εργασίας. Η διαδικασία που ακολουθείται περιγράφεται παρακάτω.

Παραδοχές για το τρίτο μέρος τις εργασίας

Για τη συνάρτηση psycho το παράθυρο hanning της εκφώνησης ήταν το μισό και αντί αυτού χρησιμοποιήθηκε αυτό το οποίο παρέχεται από το πρότυπο. , όπου N ο αριθμός των δειγμάτων του παραθύρου (256 για ESH, 2048 για τα άλλα). Επιπλέον, από το παράθυρο περάστηκαν και τα προηγούμενα frames πριν τον υπολογισμό του fft τους. Παράλληλα, επειδή πολλές από τις τιμές των tb ήταν είτε αρνητικές είτε μεγαλύτερες από τη μονάδα, αφού υπολογίζονταν κανονικοποιούνταν στο διάστημα (0,1).

Για τον κβαντιστή μετά τον υπολογισμό των scalefactors και μετά την εφαρμογή του DPCM, οι συντελεστές στρογγυλοποιούνται για να μπορέσουν να εισαχθούν στη συνάρτηση encodeHuff που παρέχεται. Το Global gain του frame δεν κωδικοποιείται και δεν βρίσκεται μέσα στον πίνακα των sfc.

Τέλος, κατά την αποκωδικοποίηση όσες τιμές είναι μικρότερες από το -1, ή μεγαλύτερες από 1, τίθενται -1 και 1 αντίστοιχα.

AACoder3 και iAACoder3

Κατά την κωδικοποίηση τα frames, αφού πρώτα περάσουν από τη βαθμίδα SSC και TNS, εισάγονται στη συνάρτηση psycho στην οποία υπολογίζεται το Signal to Mask Ratio για το εκάστοτε frame. Έπειτα, οι συντελεστές MDCT που προέκυψαν από την βαθμίδα TNS κβαντίζονται ακριβώς όπως αναφέρεται στην εκφώνηση της εργασίας στη συνάρτηση AACquantizer και ακολουθεί η κωδικοποίηση Huffman τόσο των scale factors όσο και των κβαντισμένων συντελεστών MDCT.

Η αντίστροφη διαδικασία ακολουθείται στον αποκωδικοποιητή. Αποκωδικοποιούνται τα scale factors,οι συντελεστές MDCT και στη συνέχεια αποκβαντίζονται. Επιπλέον εφαρμόζεται ο αντίστροφος TNS και filterbank και παράγονται τα frames.

Αποτελέσματα 3ου μέρους

Είναι σαφές ότι το τρίτο επίπεδο, σε αντίθεση με τα άλλα, εισάγει σφάλμα στη διαδικασία και επομένως αναμένεται σημαντική πτώση στο SNR σε σχέση με προηγουμένως. Πράγματι, για παράθυρο SIN έχουμε **12.8972dB** ενώ για KBD **12.7862dB**.

Για τον υπολογισμό του bitrate έγιναν οι εξής παραδοχές:

* Εφόσον υπάρχουν 2 τύποι παραθύρων, που προδιαγράφονται από το πρότυπο, 1 bit είναι αρκετό για αποστολή στο δέκτη
* Όμοια για τους τύπους των frames αφού είναι 4 απαιτούνται 2 bits για κάθε frame
* Οι συντελεστές TNS κωδικοποιούνται με 4 bits, όπως προκύπτει από τον κβαντιστή που χρησιμοποιήθηκε
* Για το global gain χρησιμοποιούνται 8 bits

Στον παρακάτω πίνακα συνοψίζονται τα αποτελέσματα των πειραμάτων.

|  |  |  |
| --- | --- | --- |
| Παράθυρο | Sinusoid | Kaiser-Bessel-Derived |
| Bitrate(Kbits per second) | 173.09 | 172.57 |
| Compression | 8.8742 | 8.9007 |

Για όλα τα μέρη υπάρχουν αναλυτικά σχόλια στον κώδικα. Τα μέρη 1 και 2 τερματίζουν σε περίπου 10-13 δευτερόλεπτα. Το 3ο μέρος απαιτεί περίπου 95 δευτερόλεπτα με τον περισσότερο χρόνο να καταλαμβάνεται από τις συναρτήσεις encodeHuff και AACquantizer. Επιπλέον, τα αποτελέσματα που παρουσιάζονται, το bitrate και το compression, φαίνονται να είναι αρκετά κοντά με τα πραγματικά standards για το AAC.