Ψηφιακή Επεξεργασία Σημάτων

2η εργαστηριακή Άσκηση

Κατσαϊδώνης Νικόλαος 03121868

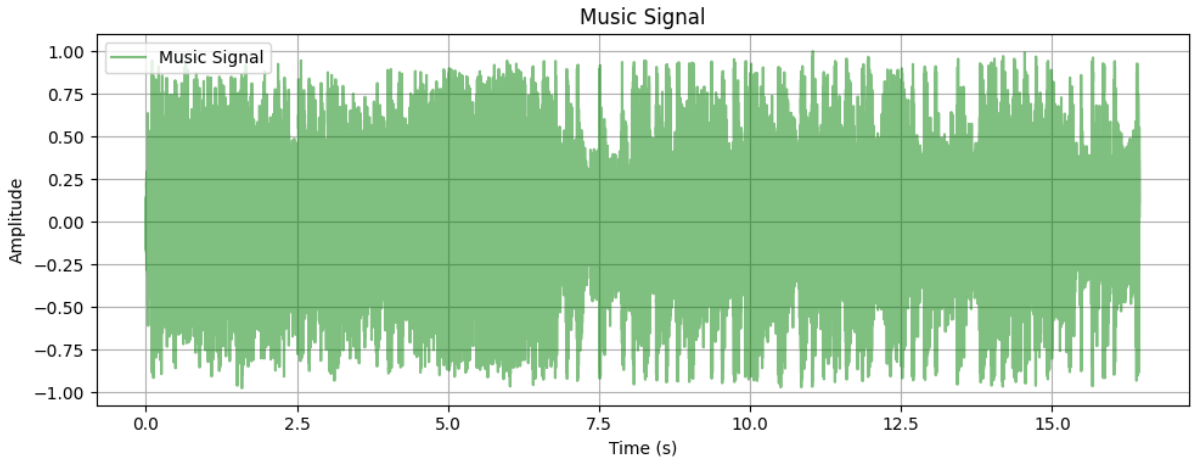
Τζαμουράνης Γεώγιος 03121141

Μέρος 1. Ψυχοακουστικό Μοντέλο 1

Τα σύγχρονα μοντέλα κωδικοποίησης του ήχου, για να συμπιέσουν μία ηχογράφηση, λειτουργούν δίνοντας **έμφαση στις συχνότητες που γίνονται αντιληπτές** από το **σύστημα ακοής του ανθρώπου** σύμφωνα με το ψυχοακουστικό μοντέλο. Για να επιτευχθεί η κωδικοποίηση των μουσικών σημάτων χρησιμοποιήσαμε αλγορίθμους όπως ο MPEG, ο οποίος βασίζεται στην φασματική ανάλυση, τον εντοπισμό **μασκών τόνων και θορύβου και στον υπολογισμό κατωφλιών κάλυψης**.

Βήμα 1.0: Προεπεξεργασία του σήματος

Σε αυτό το πρώτο βήμα διαβάσαμε το σήμα **music.wav** όπου και το μετατρέψαμε από stereo format σε **mono**. Στη συνέχεια το σήμα αυτό το κανονικοποιήσαμε σε όλο το μήκος του στο διάστημα [-1,1], διαιρώντας τα δείγματα τα του με την απόλυτη μέγιστη τιμή του. Έτσι πλοτάραμε το σήμα μουσικής όπως φαίνεται παρακάτω:

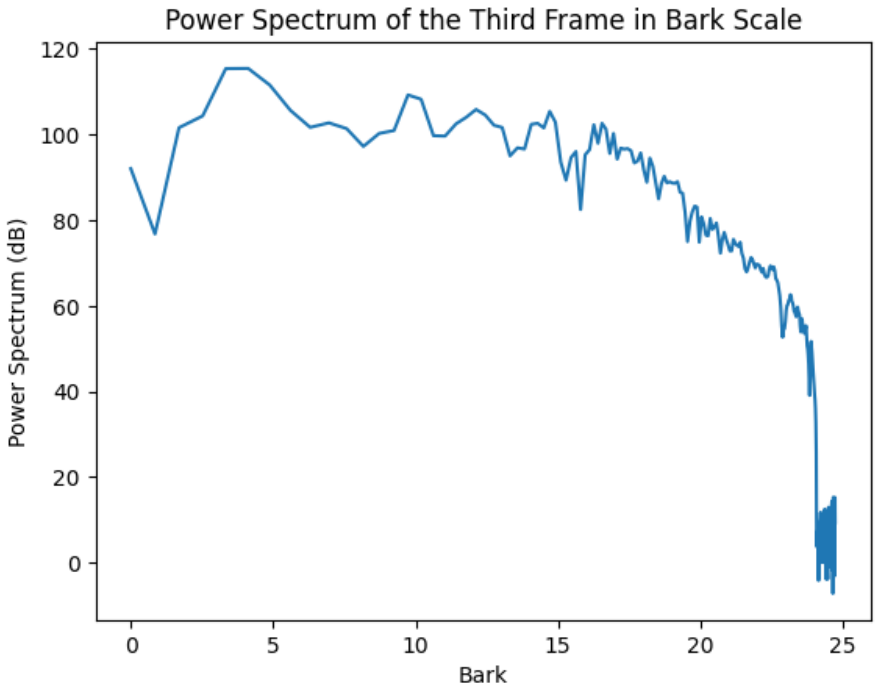


Επίσης σε αυτό το σημείο **παραθυρώνουμε το σήμα με παράθυρα Hanning** μεγέθους 512 δειγμάτων ώστε να αναλύσουμε στη συνέχεια πλάισια.

Βήμα 1.1: Φασματική Ανάλυση

Σε αυτό το βήμα θέλαμε να εκφράσουμε το φάσμα του σήματος σε μονάδες SPL οι οποίες εκφράζουν τη **πίεση του αέρα του τυμπάνου του αυτιού**. Για να το πετύχουμε αυτό ορίζουμε τη **κλίμακα Bark** και υπολογίζουμε το **φάσμα ισχύος P(k)** του σήματος με N=512 δείγματα σύμφωνα με το πρότυπο MPEG Layer-1. Επειδή το φάσμα ισχύος P(k) έχει **συμμετρία αντίστοιχη του μέτρου του DFT**, τελικά κρατάμε το **μονόπλευρο φάσμα ισχύος,** που αντιστοιχεί σε k ∈ [0,N/2] σύμφωνα με τον δοσμένο τύπο με **PN=90.302dB**:

$$P(k) = PN + 10\log_{10}\left|\sum_{n=0}^{N-1} w(n)x(n)e^{-j\frac{2\pi kn}{N}}\right|^2, 0 \le k \le \frac{N}{2}.$$

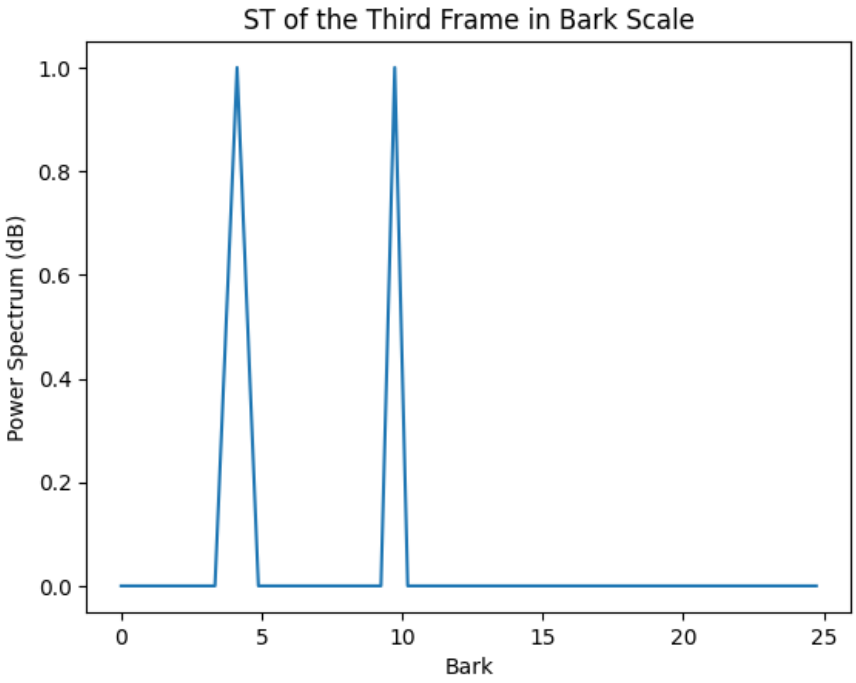


Επιλέξαμε να δουλεύουμε με το **τρίτο παράθυρο** για το οποίο πλοτάραμε το φάσμα ισχύος σε κλίμακα bark, όπως φαίνεται παραπάνω. Παρατηρούμε ότι το φάσμα ισχύος του παραθυρομμένου σήματος στις **χαμηλές και στις μεσαίες συχνότητες είναι σχετικά σταθερά υψηλό** σε αντίθεση με τις υψηλές συχνότητες όπου τείνει στο 0.

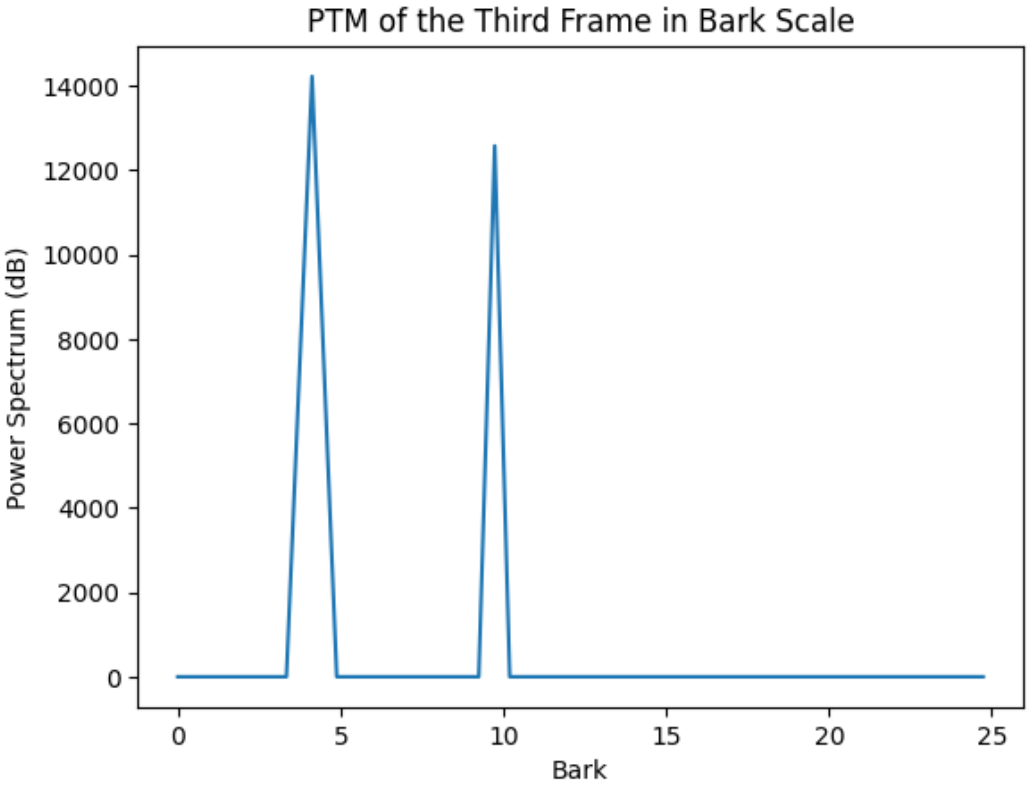
Βήμα 1.2: Εντοπισμός μασκών τόνων και θορύβου (Maskers)

Αφού υπολογίσαμε το φάσμα ισχύος, το θέλουμε να εντοπίσουμε τα τοπικά μέγιστα (μάσκες) ανά critical band. Τα τοπικά μέγιστα είναι τα μεγαλύτερα φάσματα ισχύος σε σύγκριση με τις γειτονικές τους συχνότητες κατά τουλάχιστον 7db. Η συνάρτηση **ST(k)** που παραθέτουμε παρακάτω **επιστρέφει 1 εάν στη θέση k υπάρχει τονική μάσκα και 0 αλλιώς**:

$$S_T(k) = \begin{cases} 0, & \text{αν } k \notin [2, 250) \\ P(k) > P(k \pm 1) \bigwedge P(k) > P(k \pm \Delta_k) + 7\text{dB}, & \text{αν } k \in [2, 250) \end{cases}$$



Παρακάτω βλέπουμε την ισχύ των τόνων των μασκών PTM(k) για το τρίτο πλαίσιο που διαλέξαμε:

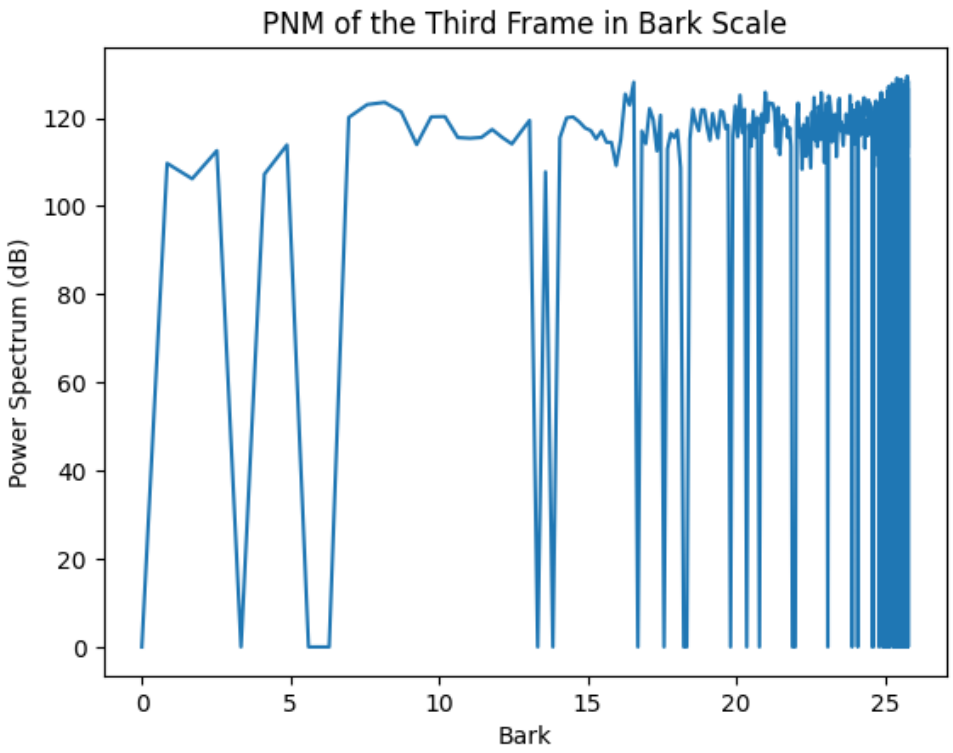


Η ισχύς δίνεται από τον παρακάτω τύπο:

$$P_{TM}(k) = \begin{cases} 10\log_{10}(10^{0.1(P(k-1))} + 10^{0.1(P(k))} + 10^{0.1(P(k+1))})(\text{dB}), & \alpha\nu\ S_T(k) = 1 \\ 0, & \alpha\nu\ S_T(k) = 0 \end{cases}$$

Αυτό που παρατηρούμε είναι ότι οι **ισχύς είναι μηδενική στο μεγαλύτερο μέρος εκτός από τα σημεία που έχουμε μάσκα** τα οποία και ξεχωρίζουν έναντι των γειτονικών τους.

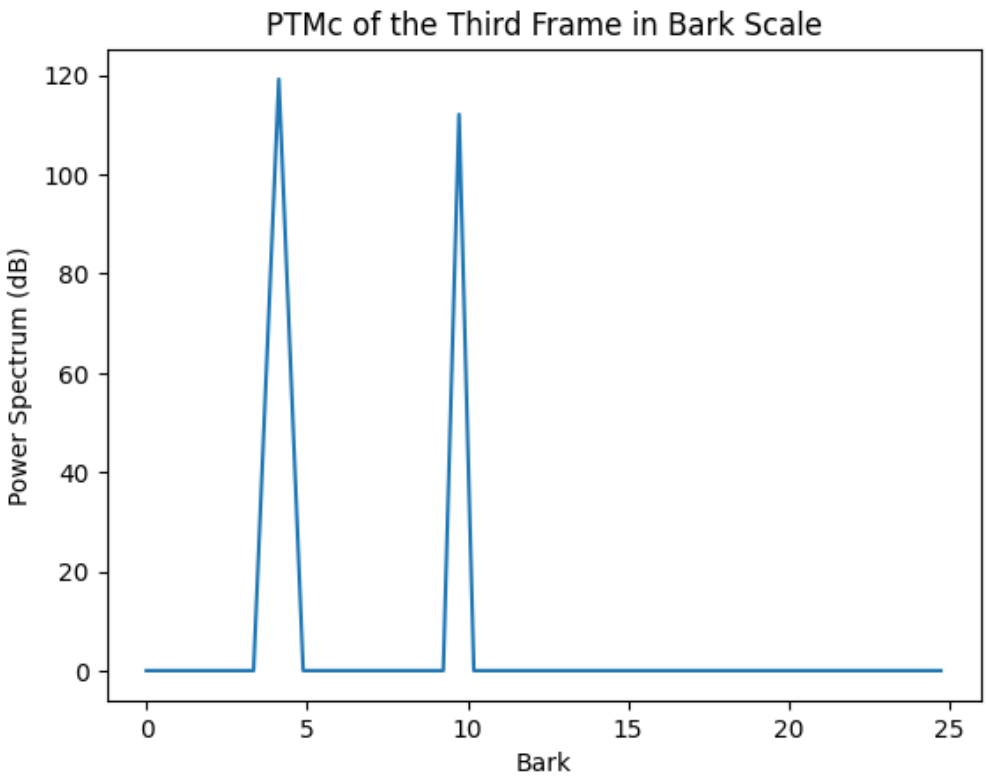
Τέλος, στο βήμα αυτό θα φορτώσουμε τον πίνακα P_NM που περιέχει μάσκες θορύβου για το κάθε παράθυρο του σήματος τις οποίες και αναπαριστούμε για το πλαίσιο που έχουμε διαλέξει:

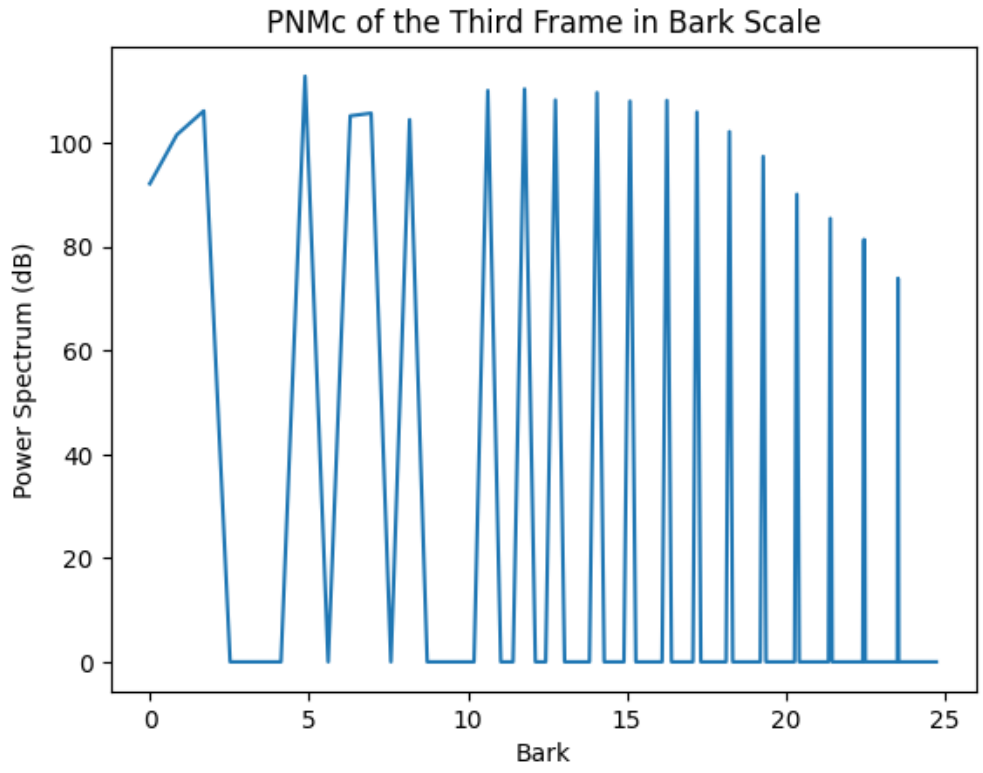


Βήμα 1.3: Μείωση και αναδιοργάνωση των μασκών

Στο βήμα αυτό θα μειώσουμε και θα αναδιοργανώσουμε τις μάσκες. Πιο συγκεκριμένα, κάθε τονική μάσκα και κάθε μάσκα θορύβου η οποία βρίσκεται **κάτω από το κατώφλι απόλυτης ακοής απορρίπτεται** ενώ επιπλέον, στα παράθυρα του 0.5 Bark βρίσκουμε τις μάσκες και τις **αντικαθιστούμε με την πιο δυνατή σε ένταση μάσκα**.

Αφού φορτώσουμε τους πίνακες των μειωμένων και αναδιοργανωμένων μασκών P_TMc, P_NMc παραθέτουμε τις γραφικές παραστάσεις για το επιλεχθέν πλαίσιο:


PTMc of the Third Frame in Bark Scale
Power Spectrum (dB)
120
100
80
60
40
20
0
0
5
10
15
20
25
Bark


PNMc of the Third Frame in Bark Scale
Power Spectrum (dB)
100
80
60
40
20
0
0
5
10
15
20
25
Bark

Βήμα 1.4: Υπολογισμός των δυο διαφορετικών κατωφλίων κάλυψης (Individual Masking Thresholds)

Σύμφωνα με τους παρακάτω τύπους υπολογίζουμε τα **κατώφλια κάλυψης** για τα δύο είδη μασκών τα οποία παριστούν το ποσοστό κάλυψης σε κάθε σημείο i, που προέρχεται από μάσκα στο σημείο j:

$$T_{TM}(i,j) = P_{TM}(j) - 0.275b(j) + SF(i,j) - 6.025(\text{dB SPL}),$$

$$T_{NM}(i,j) = P_{NM}(j) - 0.175b(j) + SF(i,j) - 2.025(\text{dB SPL})$$

Η συνάρτηση SF(i,j) που εμφανίζεται, δηλώνει το **ελάχιστο επίπεδο ισχύος που χρειάζονται οι γειτονικές συχνότητες για να γίνουν αντιληπτές από το ανθρώπινο ακουστικό σύστημα** και δίνεται από τη σχέση:

$$SF(i,j) = \begin{cases} 17\Delta_b - 0.4P_{TM}(j) + 11, & -3 \le \Delta_b < -1 \\ (0.4P_{TM}(j) + 6)\Delta_b, & -1 \le \Delta_b < 0 \\ -17\Delta_b, & 0 \le \Delta_b < 1 \\ (0.15P_{TM}(j) - 17)\Delta_b - 0.15P_{TM}(j), & 1 \le \Delta_b < 8 \end{cases}$$

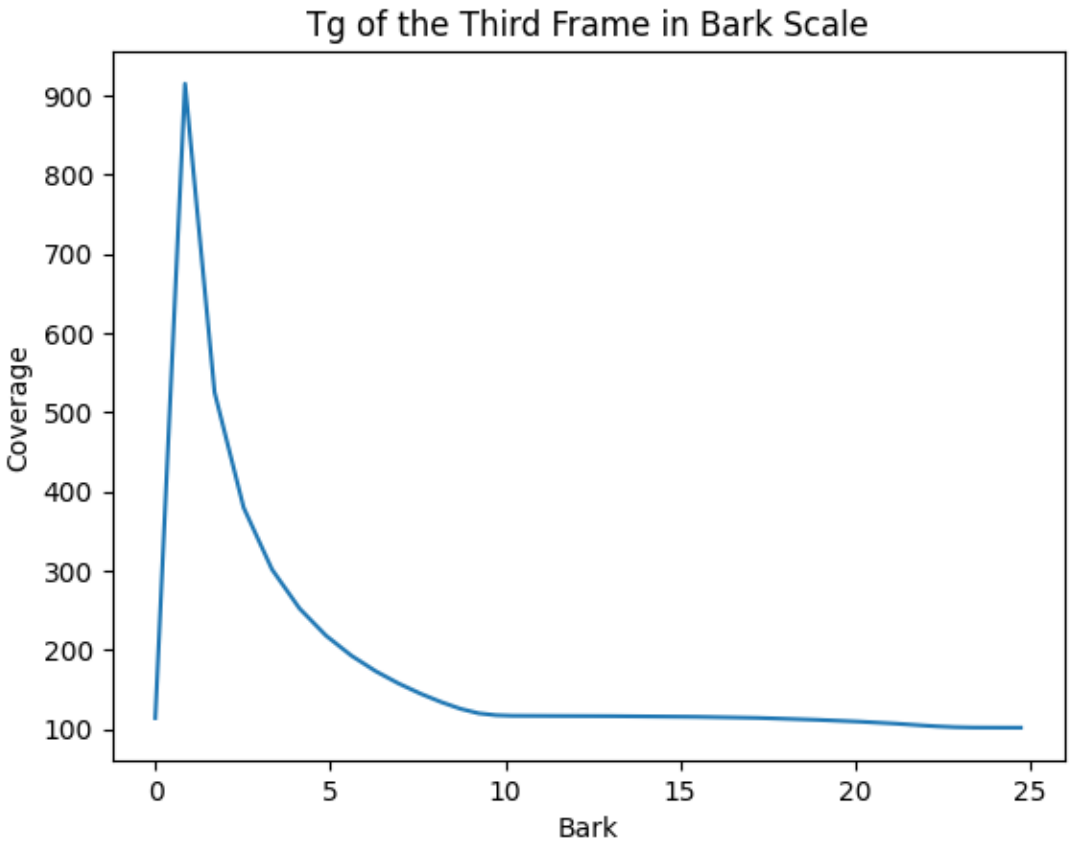
Στο μοντέλο αυτό θεωρήσαμε πως το κατώφλι TTM ορίζεται σε ένα διάστημα γειτονιάς των 12-Bark της μάσκας στο σημείο j, δηλαδή στις θέσεις i : b(i) ∈ [b(j) − 3, b(j) + 8].

Βήμα 1.5: Υπολογισμός του συνολικού κατωφλίου κάλυψης (Global Masking Threshold)

Από τα δύο κατώφλια που υπολογίσαμε, μπορεί να προκύψει τώρα το συνολικό για κάθε πλαίσιο σύμφωνα με τον τύπο:

$$T_g(i) = 10\log_{10}\left(10^{0.1T_q(i)} + \sum_{l=0}^{255} 10^{0.1T_{TM}(i,\ell)} + \sum_{m=0}^{255} 10^{0.1T_{NM}(i,m)}\right) \text{ dB SPL},$$

Παρακάτω φαίνεται η γραφική παράσταση κατωφλίου για το 3ο πλαίσιο σε κλίμακα Bark:

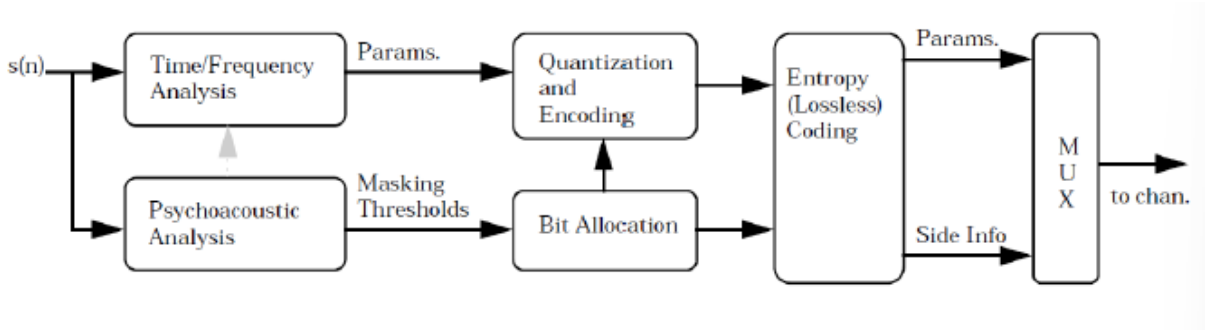


Παρατηρούμε πως για **υψηλές συχνότητες το κατώφλι είναι μηδενικό** ενώ για **χαμηλότερες εμφανίζει ένα peak που φθήνει προς τις χαμηλές**. Αυτό σημαίνει ότι στις υψηλές συχνότητες δεν υπάρχει κάλυψη από γειτονικές συχνότητες. Έτσι, στο ανθρώπινο ακουστικό σύστημα **οι χαμηλές συχνότητες είναι αυτές που αλληλοκαλύπτονται πιο εύκολα με αποτέλεσμα την δυσκολία διάκρισής τους**.

Μέρος 2. Χρονο-Συχνοτική Ανάλυση με Συστοιχία Ζωνοπερατών Φίλτρων

Σκοπός της άσκησης είναι η εξοικείωση με έννοιες όπως η **κβάντιση** και η **κωδικοποίηση** ενός ηχητικού σήματος. Για τον σκοπό αυτό θα χρησιμοποιηθούν κάποιες **συστοιχίες ζωνοπερατών φίλτρων** οι οποίες διαιρούν το φάσμα σε υποζώνες συχνοτήτων. Η χρησιμότητα αυτών φαίνεται στην ταυτοποίηση των αντιληπτικά περιττών σημείων του ηχητικού σήματος που αυτές μας παρέχουν.

Πιο συγκεκριμένα θα υλοποιηθεί η διαδικασία του παρακάτω σχήματος:



Βήμα 2.0: Συστοιχία Ζωνοπερατών Φίλτρων (Filterbank)

Ανά χρονικό πλαίσιο, χρησιμοποιούμε συστοιχίες ζωνοπερατών φίλτρων τα οποία σχεδιάζονται με βάση τον **διακριτό μετασχηματισμό συνημιτόνων** (MDCT). Χρησιμοποιήσαμε **M=32 φίλτρα** ανάλυσης και σύνθεσης, $\mathbf{h_k[n]}$ και $\mathbf{g_k[n]}$ αντίστοιχα, των οποίων οι κρουστικές αποκρίσεις φαίνονται παρακάτω:

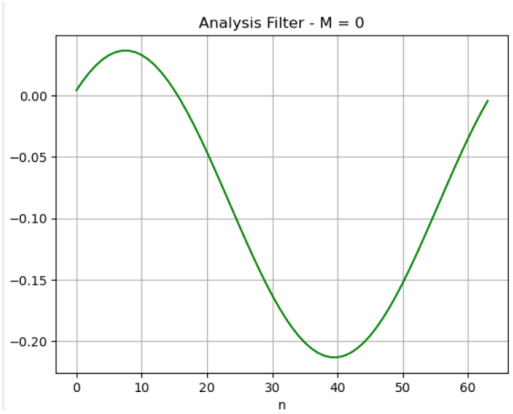
$$h_k(n) = \sin\left[\left(n+\frac{1}{2}\right)\frac{\pi}{2M}\right]\sqrt{\frac{2}{M}}\cos\left[\frac{(2n+M+1)(2k+1)\pi}{4M}\right]$$

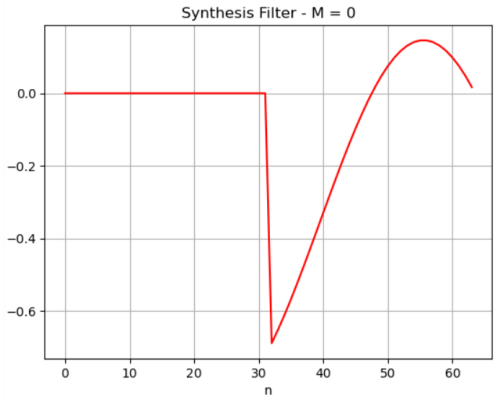
$$g_k(n) = h_k(2M-1-n)$$

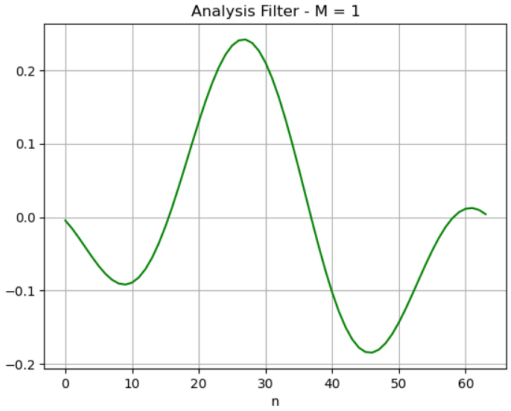
Σημειώνουμε ότι τα φίλτρα αυτά είναι μήκους L=2M με $0 \le n \le L-1$ και $0 \le k \le M-1$.

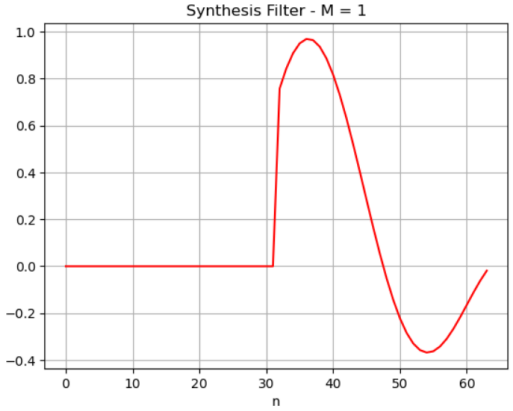
Ο κώδικας κατασκευής των φίλτρων φαίνεται στο αντίστοιχο αρχείο notebook.

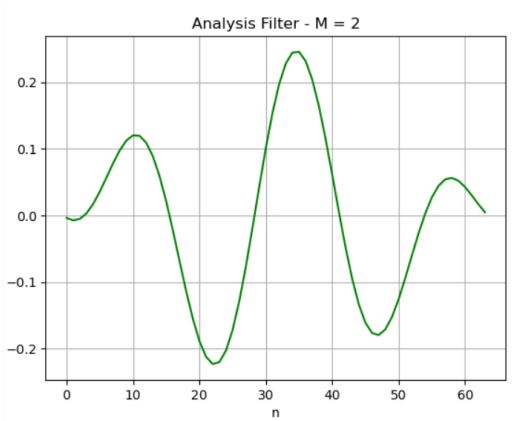
Για να έχουμε μία εικόνα των κρουστικών αποκρίσεων των φίλτρων κάνουμε plot τα πρώτα τέσσερα και τα τελευταία τέσσερα φίλτρα ανάλυσης και σύνθεσης (για M=0,1,...3 και M=28,29,...31) τα οποία και παραθέτουμε αμέσως παρακάτω:

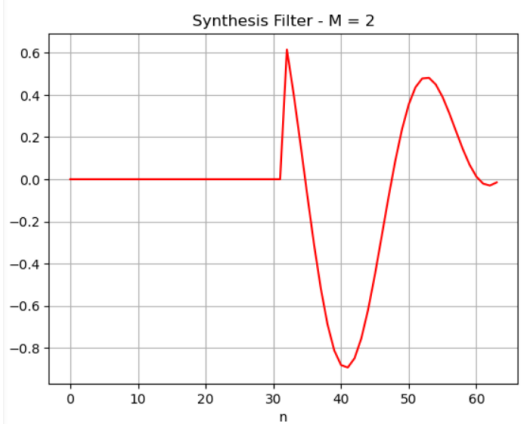

Analysis Filter - M = 0
0.00
−0.05
−0.10
−0.15
−0.20
0
10
20
30
40
50
60
n

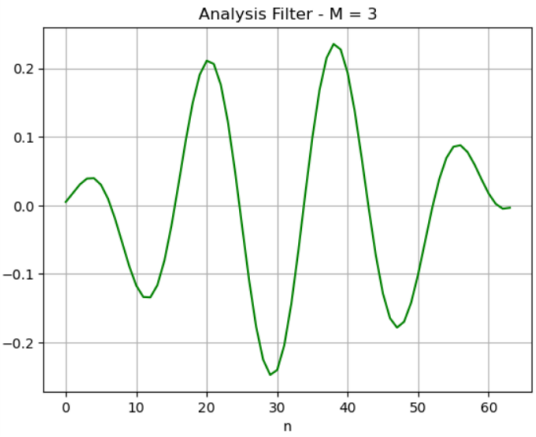

Synthesis Filter - M = 0
0.0
−0.2
−0.4
−0.6
0
10
20
30
40
50
60
n

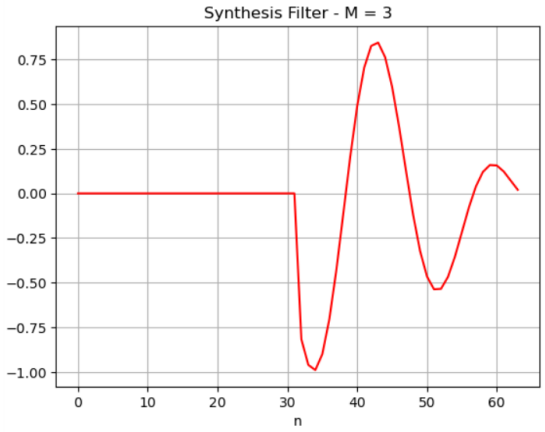

Analysis Filter - M = 1
0.2
0.1
0.0
−0.1
−0.2
0
10
20
30
40
50
60
n

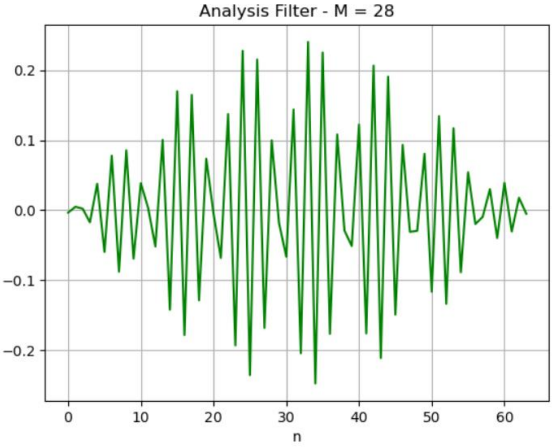

Synthesis Filter - M = 1
1.0
0.8
0.6
0.4
0.2
0.0
−0.2
−0.4
0
10
20
30
40
50
60
n

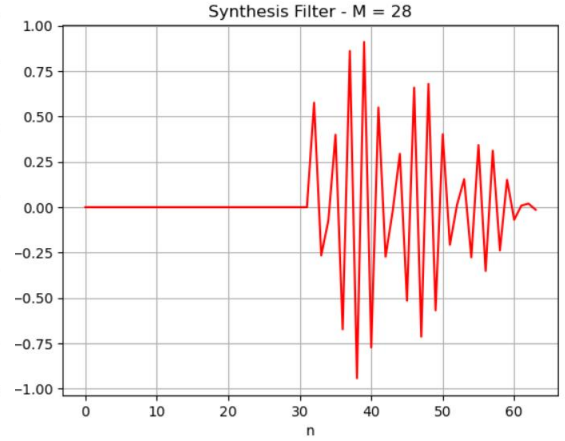

Analysis Filter - M = 2
0.2
0.1
0.0
−0.1
−0.2
0
10
20
30
40
50
60
n

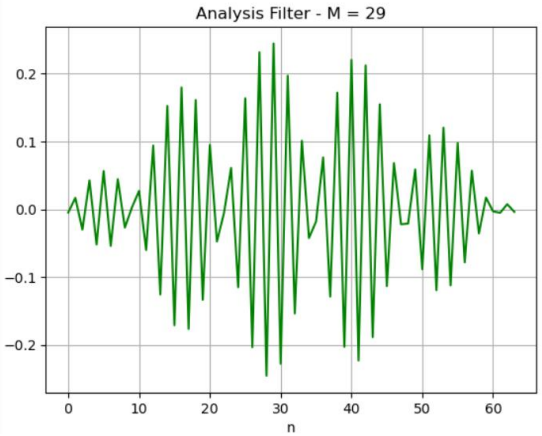

Synthesis Filter - M = 2
0.6
0.4
0.2
0.0
−0.2
−0.4
−0.6
−0.8
0
10
20
30
40
50
60
n

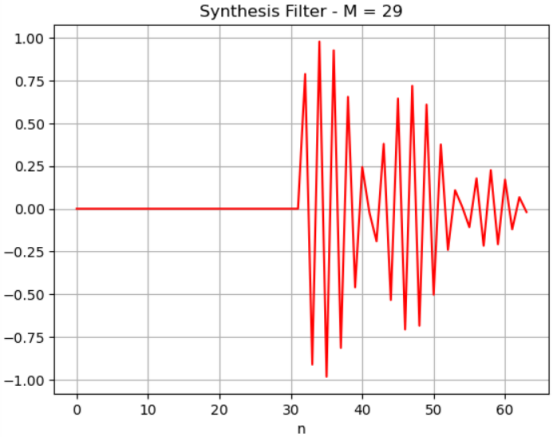

Analysis Filter - M = 3
n

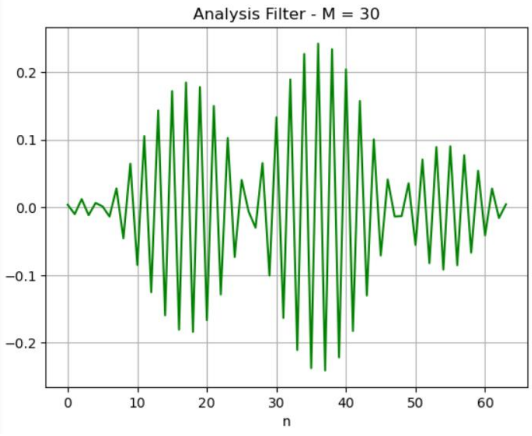

Synthesis Filter - M = 3
n

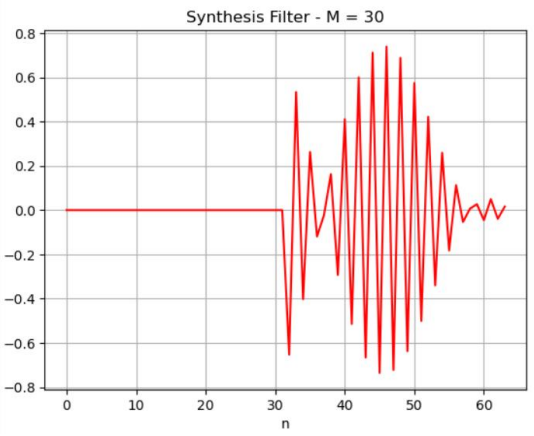

Analysis Filter - M = 28
n

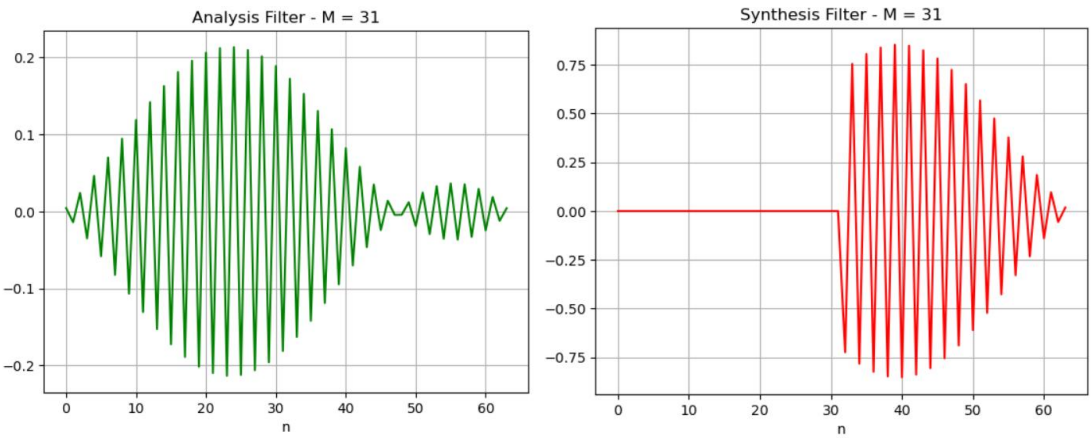

Synthesis Filter - M = 28
n


Analysis Filter - M = 29
n

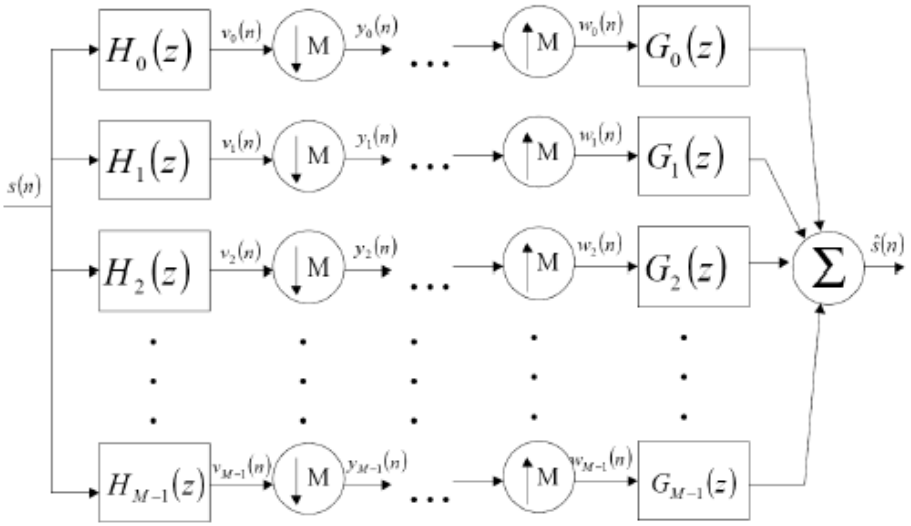

Synthesis Filter - M = 29
n


Analysis Filter - M = 30
n


Synthesis Filter - M = 30
n



Παραθέτουμε σχηματικά την διαδικασία που θα ακολουθήσει στα παρακάτω ερωτήματα:



Βήμα 2.1: Ανάλυση με Συστοιχία Φίλτρων

Στο αντίστοιχο αρχείο, έχει δημιουργηθεί κώδικας με τον οποίο **συνελίσσουμε το x[n] με τα φίλτρα ανάλυσης h_k[n]**, περνάμε δηλαδή το σήμα μέσα από τα φίλτρα και παίρνουμε εξόδους ίσες με τις συνελίξεις των σημάτος με την κρουστική απόκριση καθενός από τα φίλτρα. Παρακάτω φαίνεται ο μαθηματικός υπολογισμός των συνελίξεων **u_k[n]**:

$$v_k(n) = h_k(n) * x(n) = \sum_{m=0}^{L-1} x(n-m)h_k(m),\ k = 0, 1, \ldots, M-1$$

Αυτό επιτυγχάνεται με την χρήση την συνάρτησης **convolve()** της βιβλιοθήκης numpy.

Έπειτα, ακολουθεί μια **υποδειγματοληψία** κάθε ενός $u_k[n]$ **κατά παράγοντα M=32** έτσι ώστε το αρχικό σήμα να διαιρεθεί στις χρονικές συνιστώσες $\mathbf{y_k(n) = u_k(Mn)}$. Αυτό επιτυγχάνεται με τη μέθοδο slicing της python (y = u[::M]).

Αξίζει να σημειωθεί ότι θεωρούμε αμελητέες τις επικαλύψεις που εισάγουν τα μη ιδανικά φίλτρα.

Βήμα 2.2: Κβαντοποίηση

Η λειτουργία του **κβαντιστή** αφορά την αντιστοίχιση των τιμών των δειγμάτων μιας διακριτής ακολουθίας σε **αριθμημένα επίπεδα κβάντισης** (στάθμες). Θα κατασκευάσουμε δύο ειδών κβαντιστές, έναν **προσαρμοζόμενο ομοιόρφο κβαντιστή 2^{Bk} επιπέδων** ,όπου $\mathbf{B_k}$ είναι το πλήθος των bits κωδικοποίησης της ακολουθίας $y_k(n)$ στο τρέχον πλαίσιο x[n], και έναν **μη προσαρμοζόμενο κβαντιστή** με σταθερό **Bk=8 bits** . Το Bk υπολογίζεται μέσω του τύπου:

$$B_k = \text{int}\left(\log_2 \left(\frac{R}{\min(T_g(i))} \right) - 1 \right)$$

Το βήμα κβαντισμού Δ ρυθμίζεται βάση του B_k και του πεδίου τιμών $[x_{min}, x_{max}]$,

$$\Delta = \frac{x_{max} - x_{min}}{2^{B_k}}$$

ενώ $\mathbf{T_g(i)}$ είναι το συνολικό κατώφλι κάλυψης του ψυχοακουστικού μοντέλου όπως το υπολογίσαμε στο **Μέρος 1** και $\mathbf{R = 2^B}$ το πλήθος των βαθμίδων έντασης του

αρχικού σήματος s(n), κωδικοποιημένο στα B bits ανά δείγμα .

Βήμα 2.3: Σύνθεση

Στο βήμα αυτό, οι ακολουθίες που υπέστησαν κβαντισμό υπόκεινται **υπερδειγματοληψία κατά παράγοντα Μ**. Στην πραγματικότητα παίρνουμε το κβαντισμένο σήμα μας και κάθε 0,M,2M,… δείγματα **παρεμβάλλουμε μηδενικά** φτιάχνοντας έτσι το σήμα $\mathbf{w_k[n]}$ (Δηλαδή το w είναι ίδιο με το σήμα στις θέσεις 0,M,2M,… και αλλού μηδέν). Έπειτα το σήμα αυτό περνά από τα **φίλτρα σύνθεσης** $\mathbf{g_k[n]}$ που κατασκευάσαμε σε προηγούμενο ερώτημα, δηλαδή συνελίσσεται, και προκύπτει το σήμα εξόδου **x^[n]**. Τέλος εφαρμόζουμε τη μέθοδο **OverLap-Add** ώστε να προκύψει το σήμα μουσικής **s^[n]**.

ΕΦΑΡΜΟΓΗ ΤΩΝ ΠΑΡΑΠΑΝΩ ΣΤΟ ΣΗΜΑ ΜΟΥΣΙΚΗΣ

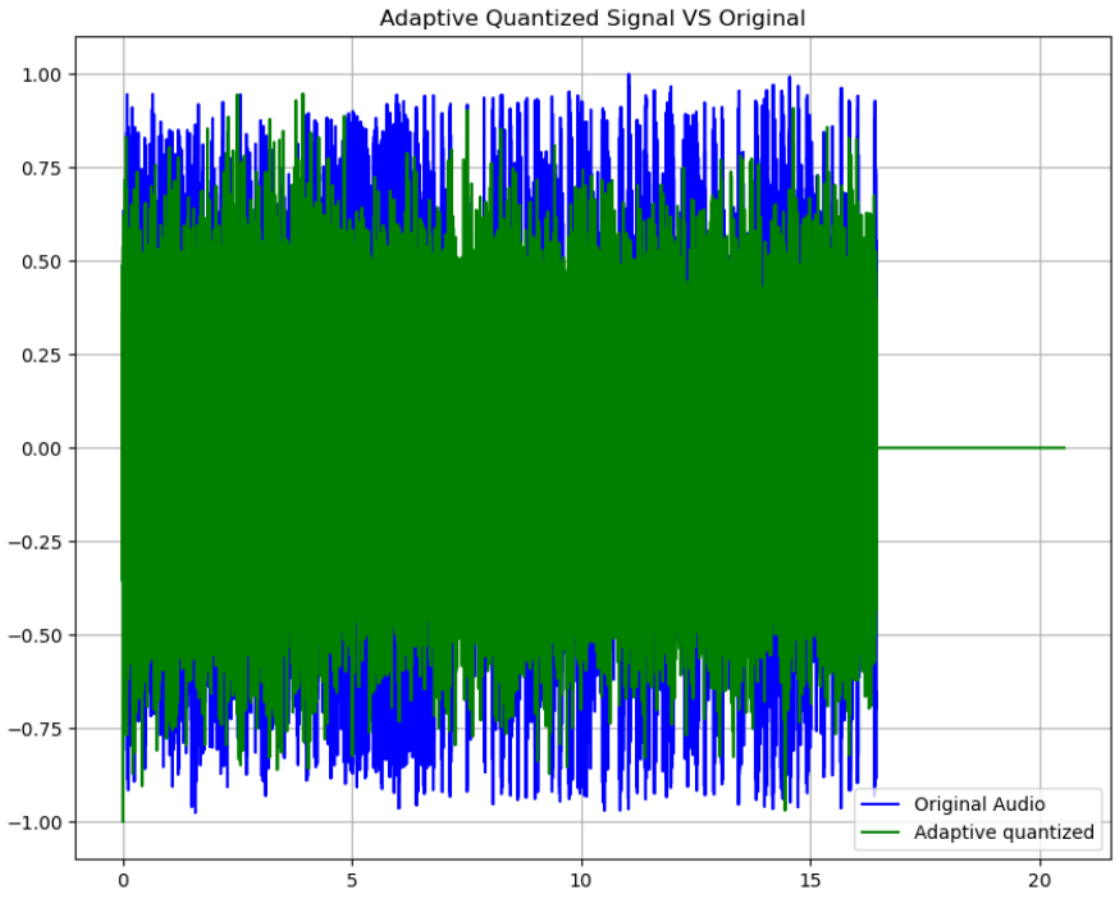
Για την εφαρμογή των παραπάνω, όπως φαίνεται και στο αρχείο κώδικα, παραθέσαμε και κάποιες χρήσιμες συναρτήσεις του μέρους ένα.

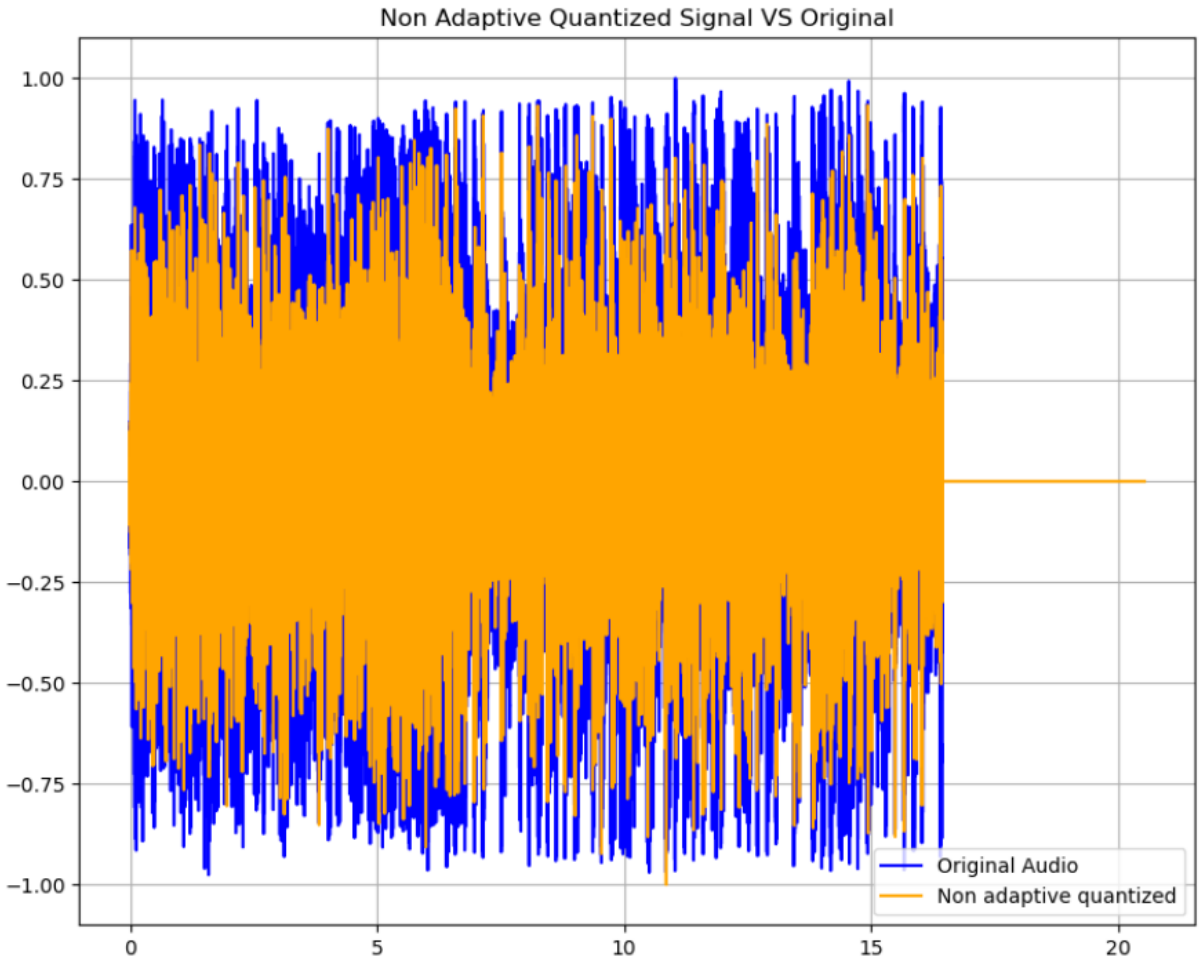
Αρχικά φορτώνουμε το σήμα μουσικής μέσω της **librosa.load()** και κανονικοποιούμε στο [-1,1]. Έπειτα χωρίζουμε το σήμα μας σε **N = 512 frames μήκους 512** μέσω της **librosa.util.frame()** και τέλος παραθυρώνουμε τα πλαίσια με παράθυρο **hanning** μήκους N = 512. Τ**υπώσαμε τα 5 πρώτα frames**.

```
Length of each frame=1416
Frame 0:
[-0. -0. -0. ... -0.  0. -0.]
Frame 1:
[-1.21434314e-07 -3.69348030e-06  2.58121242e-07 ... -2.99526235e-05
  1.18536319e-05 -7.80171451e-06]
Frame 2:
[-8.04838535e-07 -1.19857592e-05  9.72141696e-06 ... -1.28816409e-04
  4.19689301e-05 -2.40067157e-05]
Frame 3:
[-2.38620783e-06 -1.90632664e-05  3.44205200e-05 ... -2.85590012e-04
  5.98558068e-05 -4.93501535e-05]
Frame 4:
[-4.99252946e-06 -1.49212817e-05  6.80279048e-05 ... -4.70029737e-04
  7.91766334e-05 -8.74067991e-05]
```

Στην συνέχεια αφού φορτώσουμε τα αρχεία των μασκών, ακολουθεί η διαδικασία του **φιλτραρίσματος ανάλυσης**. Περνάμε τα πλαίσια του σήματος διαδοχικά από τα 32 φίλτρα ανάλυσης $\mathbf{h_k}$ και προκύπτουν τα σήματα $\mathbf{u_k}$ ακολουθεί μία **υποδειγματοληψία κατά παράγοντα M=32** (με slicing v = v[::32]). Έπειτα, περνάμε τα σήματα από έναν κβαντιστή που στη μία περίπτωση είναι **προσαρμοζόμενος ενώ στην άλλη μη προσαρμοζόμενος**. Ακολουθεί **υπερδειγματοληψία κατά M=32** δηλαδή παρεμβολή μηδενικών στο σήμα. Έπειτα τα πλαίσια θα περάσουν από τα **32 φίλτρα σύνθεσης** $\mathbf{g_k}$ και τα επιμέρους σήματα θα προστεθούν για την τελική δημιουργία του ανακατασκευασμένου σήματος.

Παραθέτουμε τα τελικά ανακατασκευασμένα σήματα μουσικής σε σχέση με το σήμα εισόδου μετά από χρήση προσαρμοζόμενου και μη προσαρμοζόμενου κβαντιστή παρακάτω:



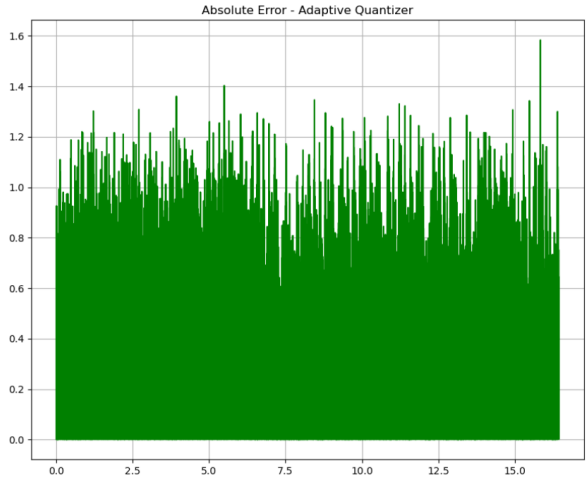


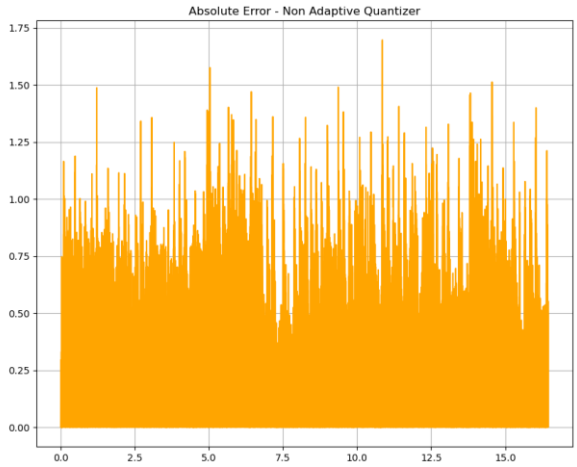
Αποθηκεύουμε σε **.wav** αρχεία τα δύο σήματα και παρατηρούμε ότι ο **θόρυβος στο προσαρμοζόμενα κβαντισμένο σήμα είναι αισθητά λιγότερος** όπως αναμέναμε. Επιπλέον, τα δύο σήματα είναι **μεγέθους 1769KB έναντι 2790KB του αρχικού** γεγονός που δηλώνει **επιτυχή συμπίεση του αρχείου music_dsp2024**.

Τέλος, υπολογίζουμε το μέσο τετραγωνικό σφάλμα για τους δύο κβαντιστές. **Τονίζουμε ότι λόγω κάποιου λάθους το οποίο δεν εντοπίστηκε, οι δύο κβαντιστές βγάζουν σχεδόν ίδιο σφάλμα με τον μη προσαρμοζόμενο να βγάζει ελαφρώς μικρότερο:**

```
Mean Square Error - Adaptive Quantizer= 0.09412908116619181
Mean Square Error -Non Adaptive Quantizer= 0.09120078425211521
```

Παραθέτουμε τα γραφήματα των απόλυτων τιμών των σφαλμάτων:





Να σημειωθεί ότι ο τελικός αριθμός των bits του προσαρμοζόμενου κβαντιστή υπολογίστηκε ίσος με 11328 bits.