

# Εθνικό Μετσόβιο Πολυτεχνείο

Σχολή Ηλεκτρολόγων Μηχανικών και Μηχανικών Υπολογιστών

Σήματα και Συστήματα, 2020-2021

## Αναφορά MATLAB

Ονοματεπώνυμο: Παναγιώτης Γιαδικιάρογλου

ΑΜ: 03119185

3<sup>ο</sup> Εξάμηνο

### 1 Σχεδίαση Φίλτρων

#### 1.1 Σχεδίαση φίλτρων ηχούς και αντήχησης (Echo, Reverb)

**1.1.α)** Η εξίσωση που περιγράφει το echo effect με  $c = 0.55$  είναι:

- Για  $P = 2$ ,  $y[n] = 0.55x[n] + 0.45x[n-2]$  (1.1.1)

- Για  $P = 5$ ,  $y[n] = 0.55x[n] + 0.45x[n-5]$  (1.1.2)

Από την εξίσωση διαφορών ενός αιτιατού φίλτρου σε διακριτό χρόνο προκύπτει άμεσα παρατηρώντας τις εξισώσεις 1.1.1 και 1.1.2 ότι και για τις δύο περιπτώσεις ισχύει ότι:

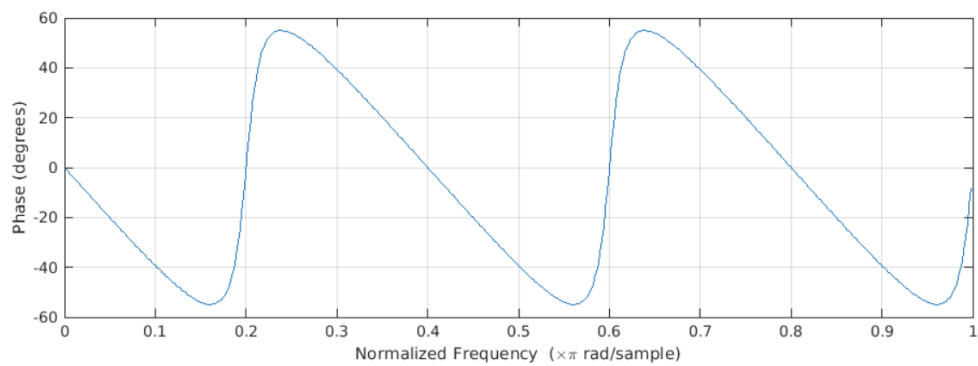
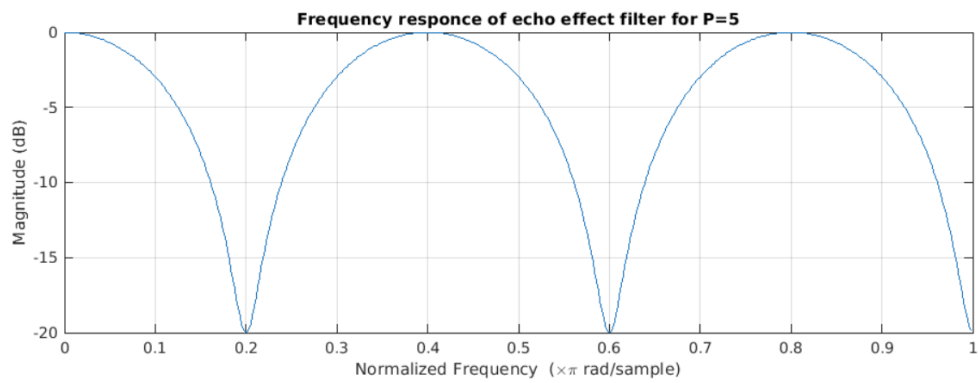
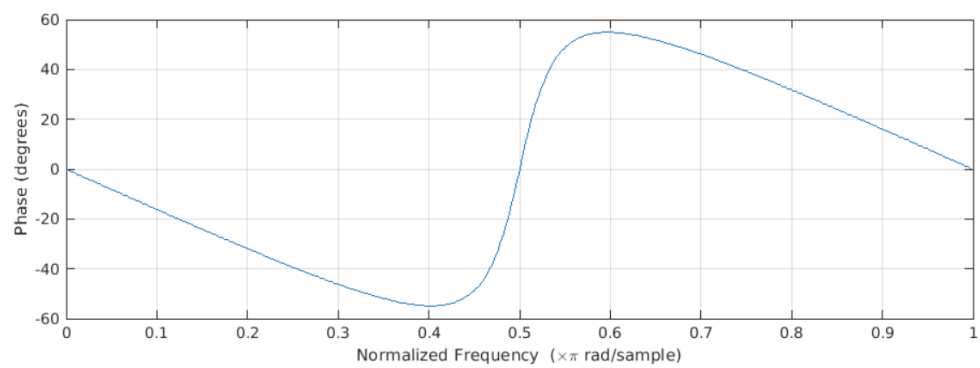
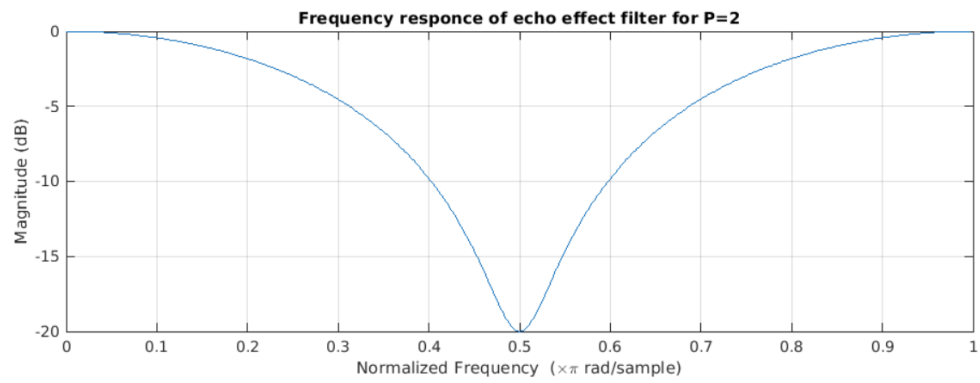
$$a_0 = 1 \text{ και } a_n = 0 \text{ για κάθε } n > 0$$

Για τα  $b_n$  προκύπτει και πάλι από παρατήρηση της εξίσωσης διαφορών ότι:

- Για  $P = 2$ ,  $b = [0.55, 0, 0.45]$

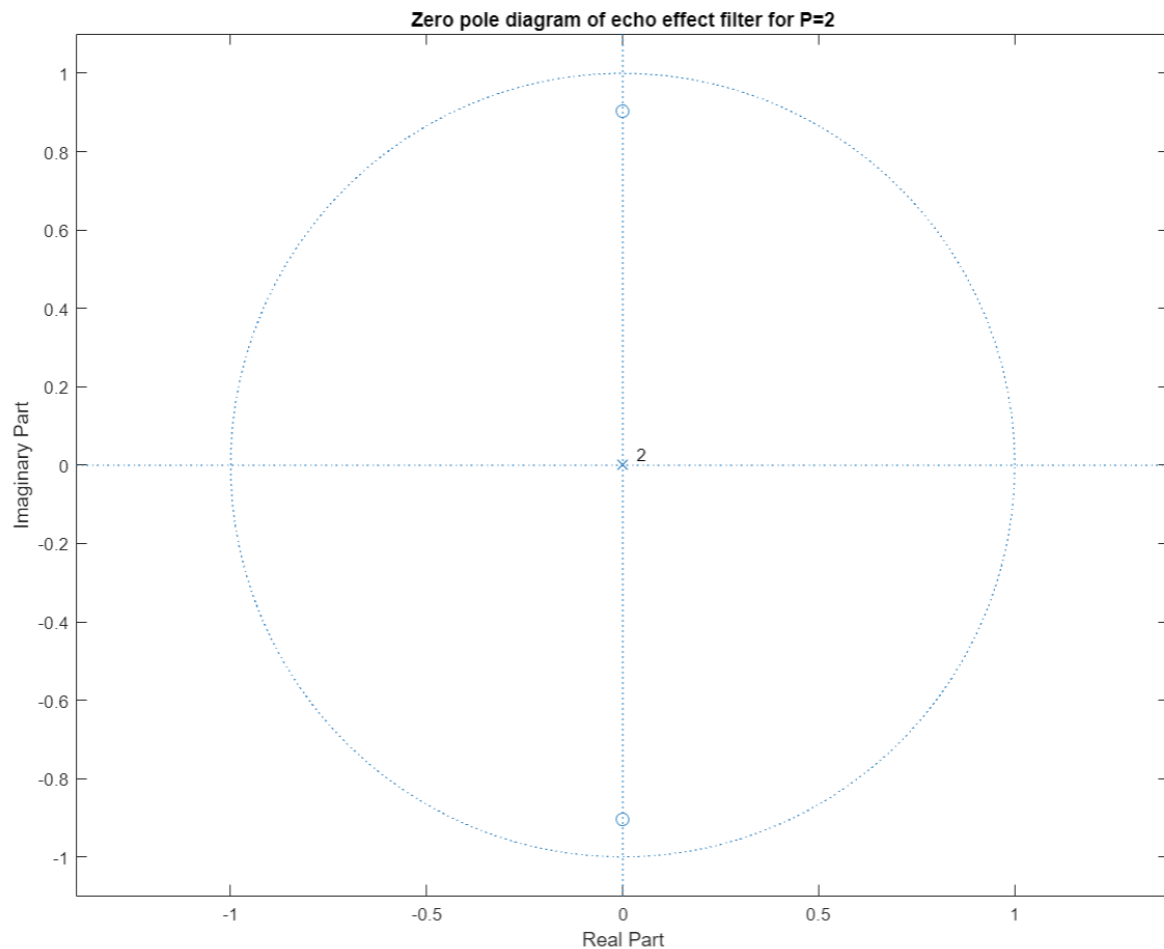
- Για  $P = 5$ ,  $b = [0.55, 0, 0, 0, 0, 0.45]$

1.1.β)

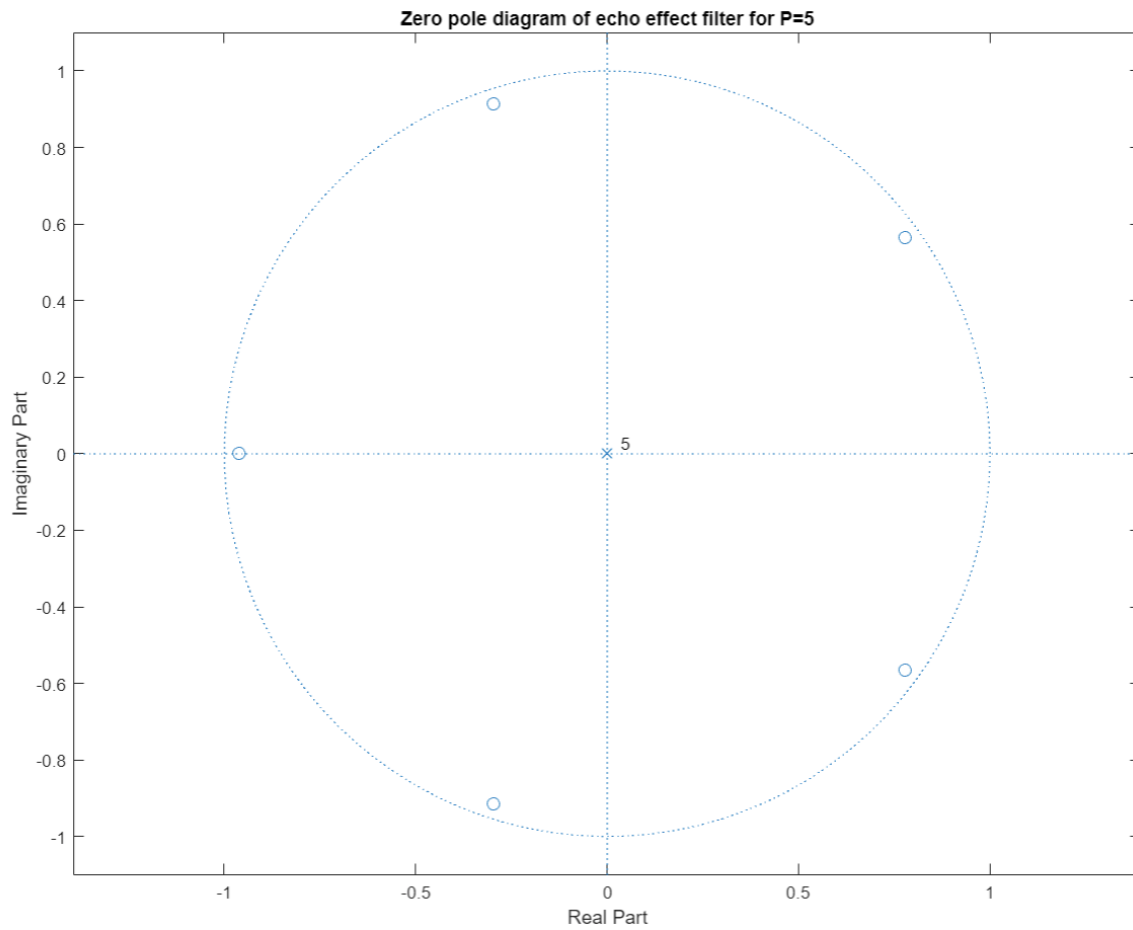


Από τα παραπάνω διαγράμματα παρατηρούμε ότι για  $P = 5$  η συχνότητα της απόκρισης πλάτους και φάσης είναι μεγαλύτερη από την αντίστοιχη για  $P = 2$ . Συγκεκριμένα, παρατηρούμε ότι η συχνότητα για  $P = 5$  είναι κατά  $5/2$  μεγαλύτερη της αντίστοιχης για  $P = 2$ , κάτι το οποίο αναμενόταν αφού η παράμετρος  $P$  καθορίζει και την καθυστέρηση του φίλτρου. Το μέγιστο πλάτος και η μέγιστη φάση είναι τα ίδια και για τις δύο περιπτώσεις.

### 1.1.γ)

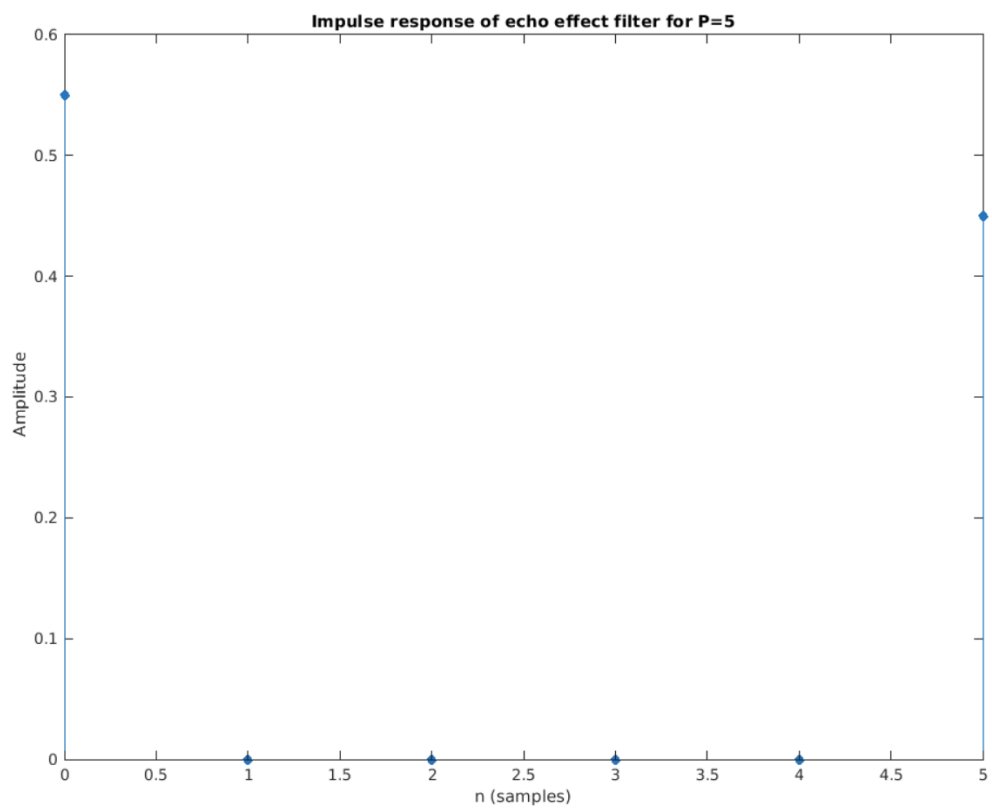
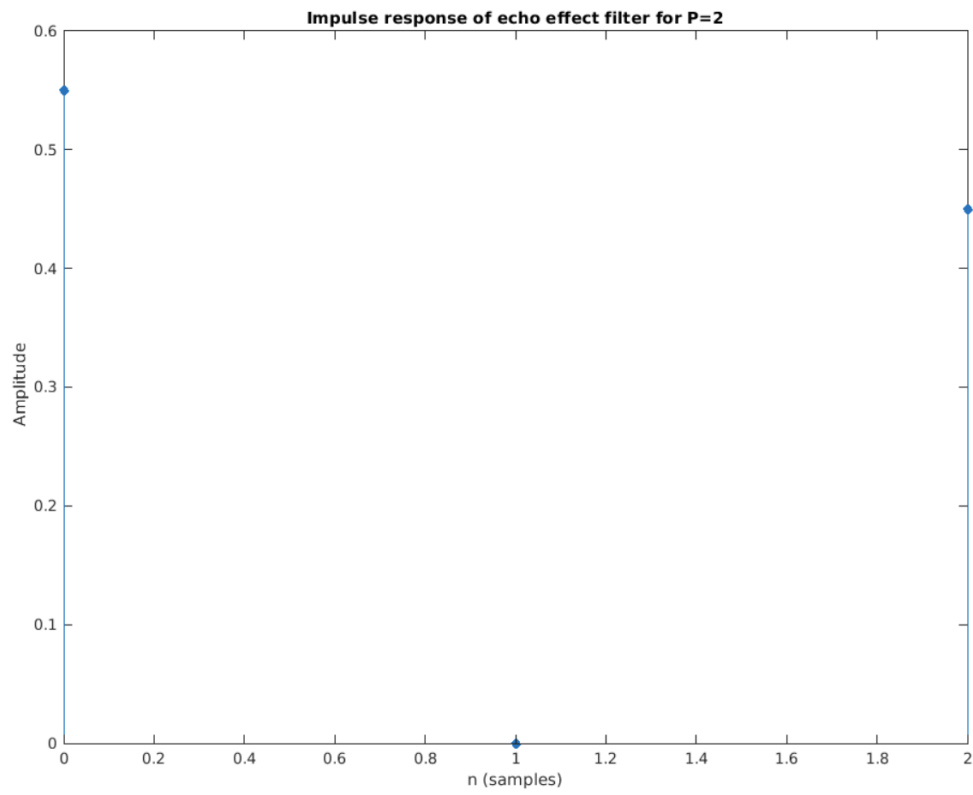


Παρατηρούμε αρχικά όπως αναμενόταν ότι τόσο οι πόλοι όσο και τα μηδενικά είναι 2 στο πλήθος όπως και το  $P$ . Οι πόλοι είναι ουσιαστικά ένας πόλος 2<sup>ου</sup> βαθμού στο σημείο  $(0, 0)$ , ενώ τα μηδενικά αποτελούν ζεύγος συζυγών μιγαδικών ριζών στα σημεία  $(0, 0.9045)$  και  $(0, -0.9045)$ . Επομένως, αφού οι πόλοι είναι το  $(0, 0)$  τότε το φίλτρο είναι ευσταθές.



Στην περίπτωση για  $P = 5$  παρατηρούμε αντίστοιχα ότι το πλήθος των πόλων αλλά και των μηδενικών είναι 5 με έναν πόλο πολλαπλότητας 5 στο  $(0, 0)$  και τα μηδενικά σε συμμετρικές θέσεις στα σημεία  $(-0.2969, 0.9136)$ ,  $(0.7772, 0.5647)$ ,  $(0.7772, -0.5647)$ ,  $(-0.2969, -0.9136)$  και  $(-0.9607, 0)$ . Και πάλι αφού έχουμε τους πόλους στο  $(0, 0)$  τότε ο μοναδιαίος κύκλος θα περιέχεται στην περιοχή σύγκλισης άρα το φίλτρο είναι και πάλι ευσταθές.

1.1.δ)



Στο (α) ερώτημα είδαμε ότι προέκυψε το διάνυσμα  $a$  να είναι 1 για  $n = 0$  και 0 για κάθε άλλο  $n$ . Από την παρακάτω σχέση:

$$\begin{aligned} h[0] &= b_0 \\ h[1] &= b_1 - a_1 h[0] \\ h[2] &= b_2 - a_1 h[1] - a_2 h[0] \\ &\vdots \\ h[q] &= b_q - a_1 h[q-1] - \dots - a_p h[q-p] \end{aligned} \quad (1.1.3)$$

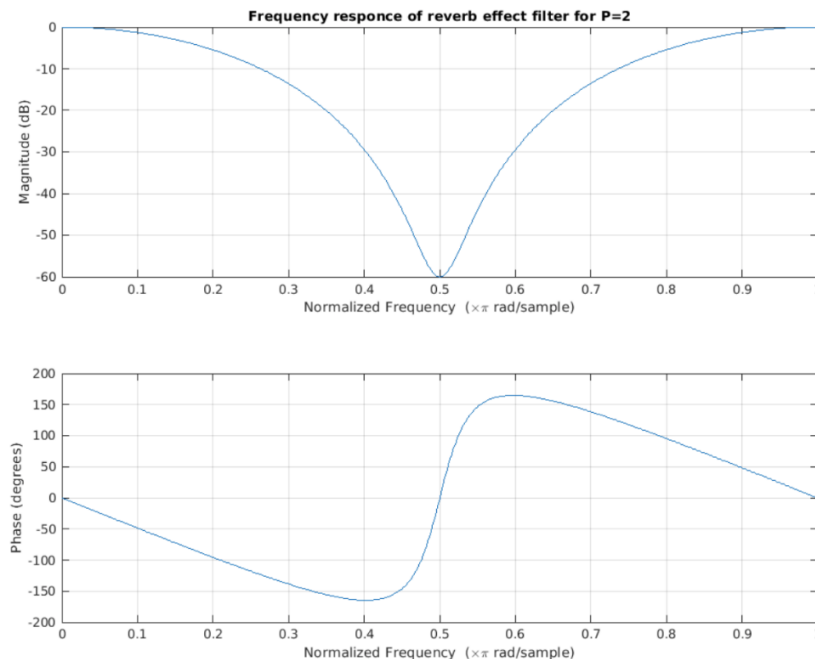
προκύπτει ότι  $h[n] = b_n$ .

Άρα από τα παραπάνω διαγράμματα επιβεβαιώνεται και η παραπάνω σχέση και για τις δύο περιπτώσεις, δηλαδή προέκυψαν τα αναμενόμενα αποτελέσματα.

**1.1.ε)** Για την υλοποίηση του φίλτρου αντήχησης 3<sup>ου</sup> βαθμού χρησιμοποιούμε 3 διαδοχικά φίλτρα ηχούς. Για την εύρεση των διανυσμάτων  $a$  και  $b$ , αφού έχουμε εντοπίσει την κρουστική απόκριση του φίλτρου αντήχησης μέσω 2 διαδοχικών συνελίξεων της κρουστικής απόκρισης του φίλτρου ηχούς με τον εαυτό της ( $h_{\text{reverb}} = h_{\text{echo}} * h_{\text{echo}} * h_{\text{echo}}$ ), χρησιμοποιήθηκε αρχικά η σχέση:

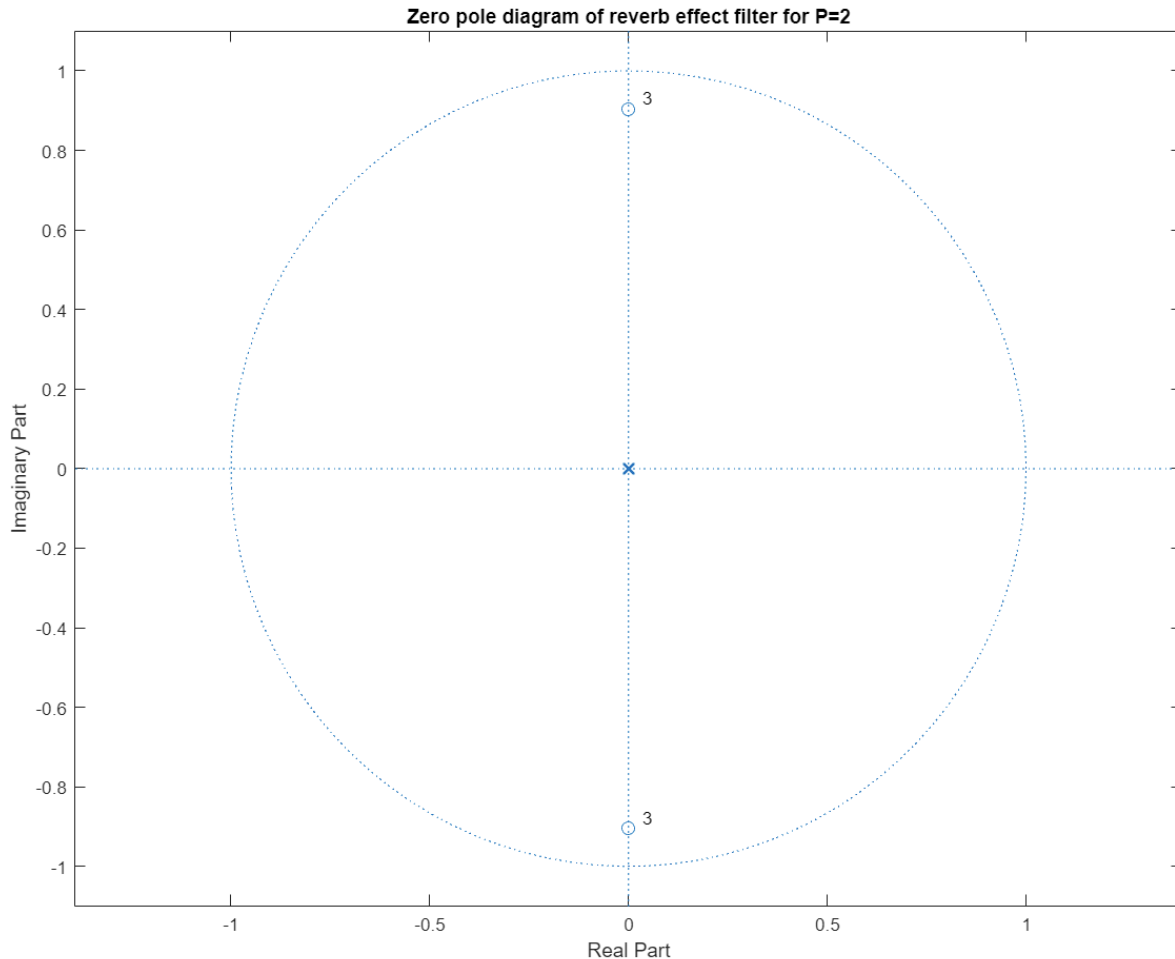
$$a = T_{\text{hn}}^{-1} \underline{h}[n] \quad (1.1.4)$$

όπου  $T_{\text{hn}}$  ο πίνακας Toeplitz και βρήκαμε ότι  $a = [1, 0, 0, 0, 0]$  ενώ στη συνέχεια προέκυψε άμεσα από τη σχέση 1.1.3 ότι  $b = [0.1664, 0, 0.4084, 0, 0.3341, 0, 0.0911]$ . Επομένως, για την απόκριση πλάτους και φάσης έχουμε:



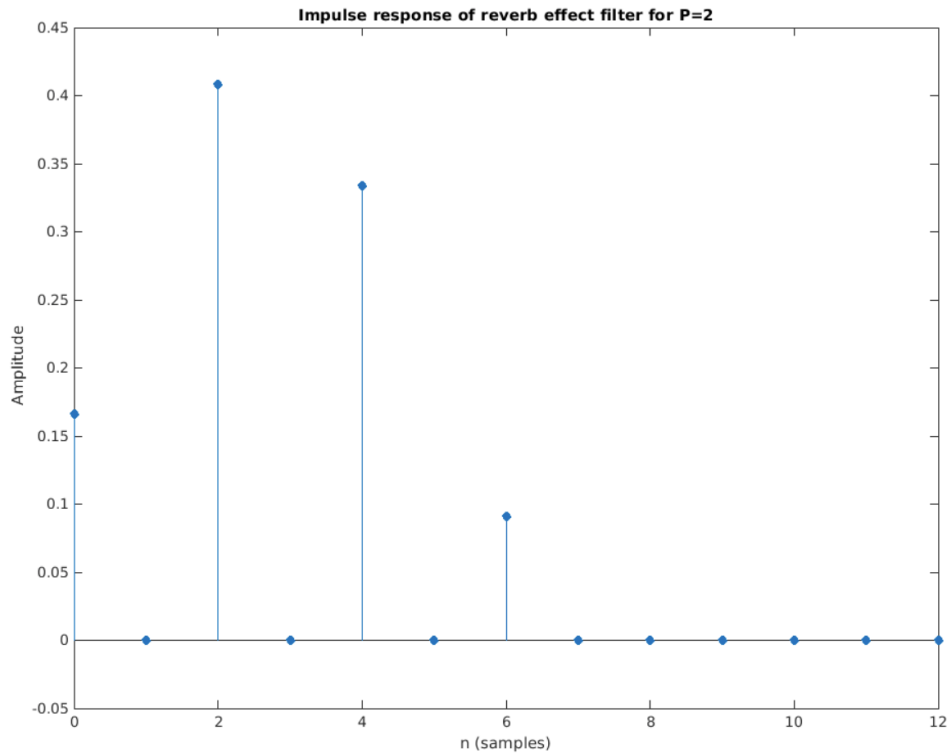
Όπως αναμενόταν, αφού το φίλτρο αντίχησης είναι 3 φίλτρα ηχούς εν σειρά, τότε το πλάτος και η φάση που προέκυψαν είναι τα τριπλάσια των αντίστοιχων για το φίλτρο ηχούς. Ωστόσο, η συχνότητα παρέμεινε προφανώς η ίδια με αυτή για το φίλτρο ηχούς με  $P = 2$  αφού η παράμετρος  $P$  είναι αυτή που την καθορίζει όπως είδαμε και προηγουμένως.

Για το διάγραμμα πόλων μηδενικών έχουμε:



Όπως αναμενόταν οι πόλοι και τα μηδενικά είναι 6 στο πλήθος, δηλαδή ο τριπλάσιος αριθμός από αυτόν που προέκυψε για το φίλτρο ηχούς. Τα μηδενικά αποτελούν ένα ζεύγος τριπλών συζυγών μιγαδικών ριζών στα σημεία  $(0, 0.904537)$  και  $(0, -0.904537)$ . Για τους πόλους αν και αρχικά φαίνεται πως έχει προκύψει μόνο ένας παρατηρούμε μετά από αρκετή μεγέθυνση ότι είναι 6 απλά όλοι βρίσκονται σε πολύ κοντινή απόσταση από το  $(0, 0)$ . Το φίλτρο είναι ευσταθές.

Τέλος, για την κρουστική απόκριση του φίλτρου αντήχησης προέκυψε και πάλι ένα διάγραμμα το οποίο ταυτίζεται με αυτό για τους συντελεστές  $b$  του φίλτρου και είναι το παρακάτω:



**1.1.στ)** Για τον υπολογισμό των διανυσμάτων  $b$ , α για το φίλτρο απαλοιφής της αντήχησης ακολουθήθηκε η παρακάτω διαδικασία:

Γνωρίζουμε ότι για τις κρουστικές αποκρίσεις των φίλτρων αντήχησης και απαλοιφής της αντήχησης ισχύει η σχέση:

$$h_{\text{reverb}}[n] * h_{\text{dereverb}}[n] = \delta[n] \quad (1.1.5)$$

Μετά από τον μετασχηματισμό  $Z$  της σχέσης 1.1.5 προκύπτει ότι:

$$H_{\text{dereverb}}(Z) = \frac{1}{H_{\text{reverb}}(Z)} \quad (1.1.6)$$

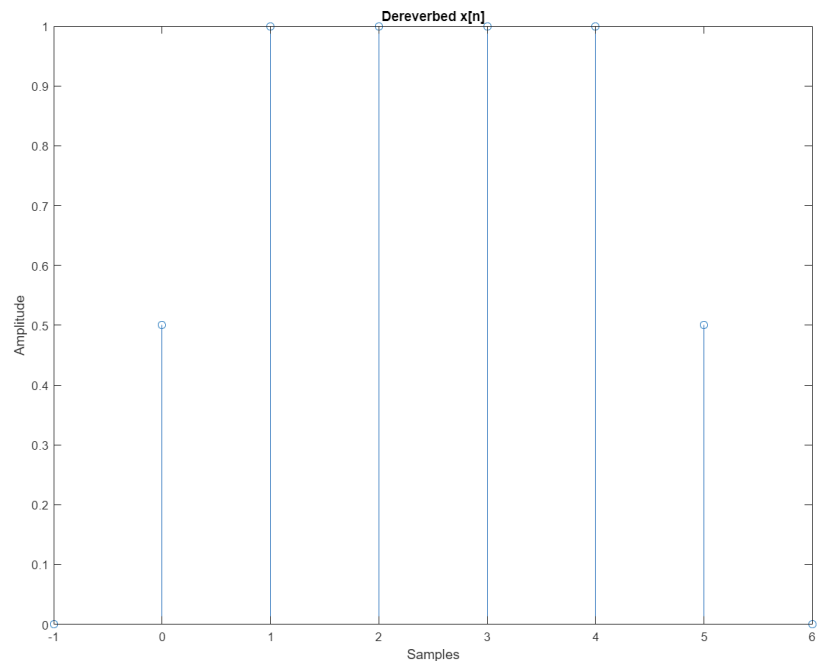
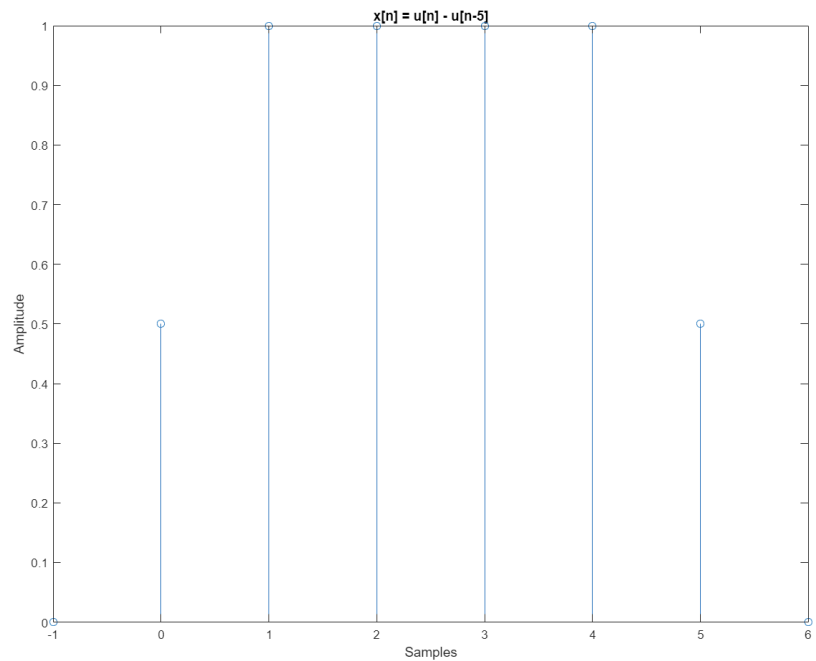
Γνωρίζουμε επίσης ότι:

$$H(Z) = \frac{\sum_{i=0}^q b_i Z^{-i}}{\sum_{i=0}^p a_i Z^{-i}} \quad (1.1.7)$$



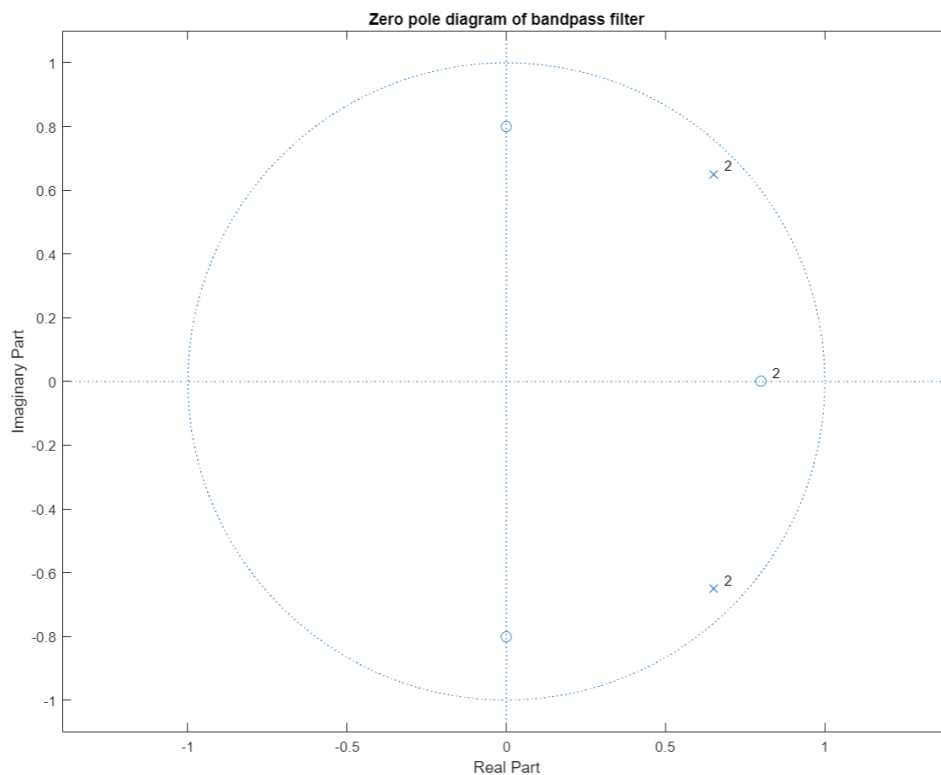
Άρα από τις σχέσεις 1.1.6 και 1.1.7 μπορεί να προκύψει άμεσα ότι το διάνυσμα  $a$  του dereverb φίλτρου ταυτίζεται με το διάνυσμα  $b$  του reverb φίλτρου και αντίστροφα το διάνυσμα  $b$  του dereverb φίλτρου είναι ίσο με το διάνυσμα  $a$  του φίλτρου αντήχησης.

Για να επαληθεύσουμε το παραπάνω χρησιμοποιήθηκε το σήμα  $x[n] = u[n] - u[n-5]$  για το οποίο εφαρμόστηκε αρχικά το φίλτρο αντήχησης και στη συνέχεια το φίλτρο απαλοιφής της αντήχησης. Το αναμενόμενο αποτέλεσμα θα ήταν το αρχικό σήμα να ταυτίζεται με το dereverbed όπως και έγινε. Παρακάτω δίνεται το αρχικό σήμα  $x[n]$  και το σήμα μετά την εφαρμογή των δύο φίλτρων:



## 1.2 Σχεδίαση Ζωνοπερατών Φίλτρων

**1.2.α)** Το διάγραμμα πόλων και μηδενικών που προέκυψε για ζεύγος διπλών μιγαδικών πόλων στις θέσεις  $(0.65, 0.65i)$ ,  $(0.65, -0.65i)$  και διπλό μηδενικό στη θέση  $(0.8, 0)$  και ζεύγος συζυγών μιγαδικών μηδενικών στις θέσεις  $(0, 0.8i)$ ,  $(0, -0.8i)$  μέσω της εντολής `zplane()` είναι το παρακάτω:



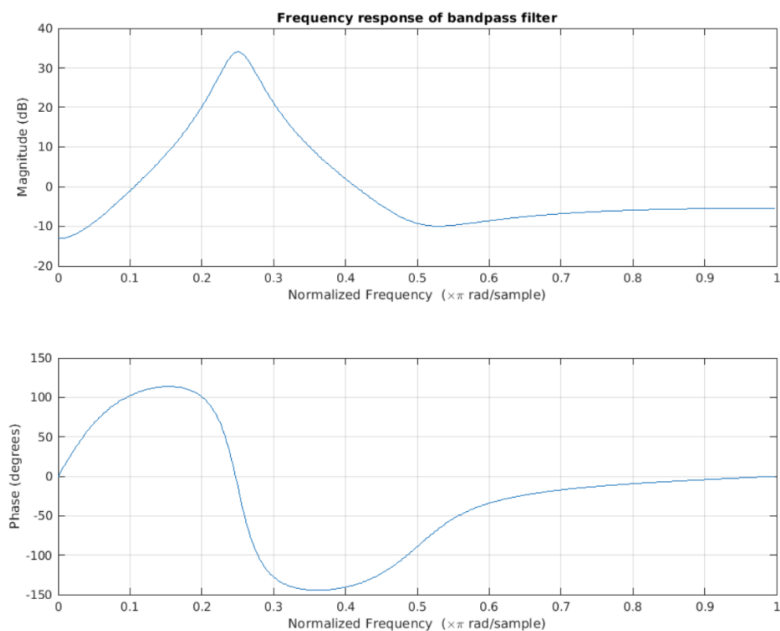
Το φίλτρο είναι ευσταθές αφού οι πόλοι ανήκουν στον μοναδιαίο κύκλο.

Τα διανύσματα  $a$ ,  $b$  που προέκυψαν μέσω της συνάρτησης `zp2tf()` για  $k = 1$  (gain) είναι τα παρακάτω:

$$a = [1, -2.6, 3.38, -2.197, 0.714]$$

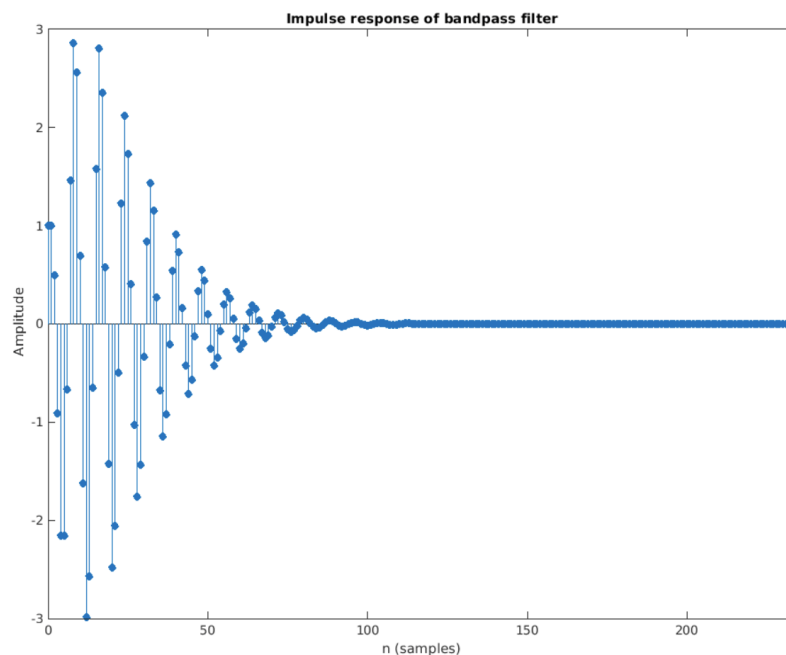
$$b = [1, -1.6, 1.28, -1.024, 0.4096]$$

**1.2.β)** Μέσω της εντολής `freqz()` προέκυψε η παρακάτω απόκριση πλάτους και φάσης του ζωνοπερατού φίλτρου:

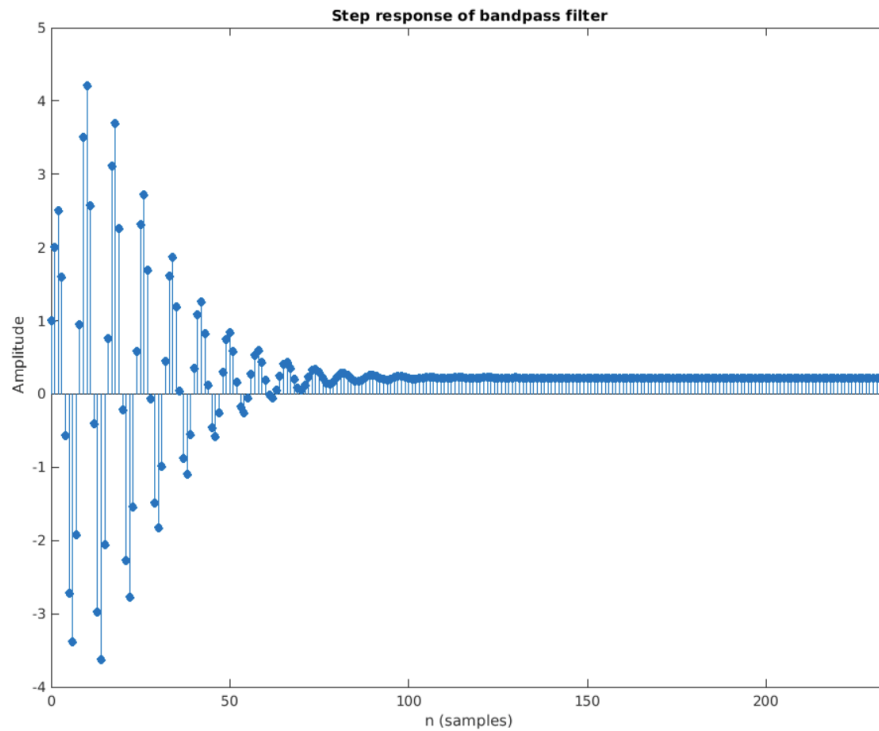


Το παραπάνω είναι όπως αναμενόταν ένα bandpass φίλτρο. Όπως προέκυψε και από πριν το παραπάνω bandpass φίλτρο είναι ευσταθές. Επίσης, όσο αυξάνουμε του  $k$  (gain) τόσο μετατοπίζεται και η απόκριση πλάτους προς τα πάνω. Στη συγκεκριμένη περίπτωση έχει επιλεγθεί αυθαίρετα  $k = 1$ .

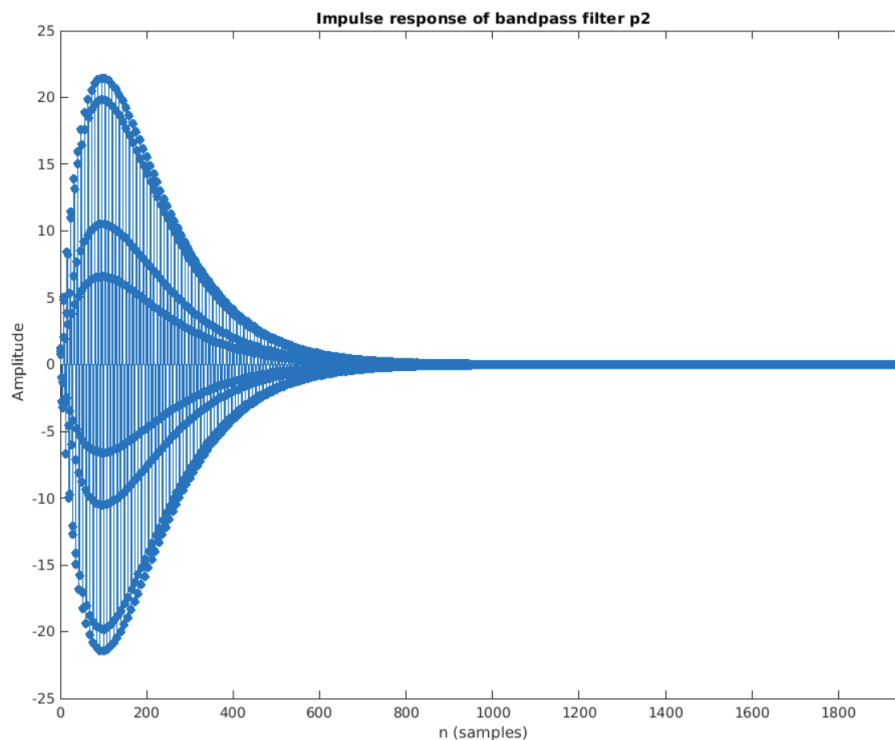
**1.2.γ)** Η κρουστική απόκριση του φίλτρου που βρέθηκε με τη συνάρτηση `impz()` είναι η εξής:



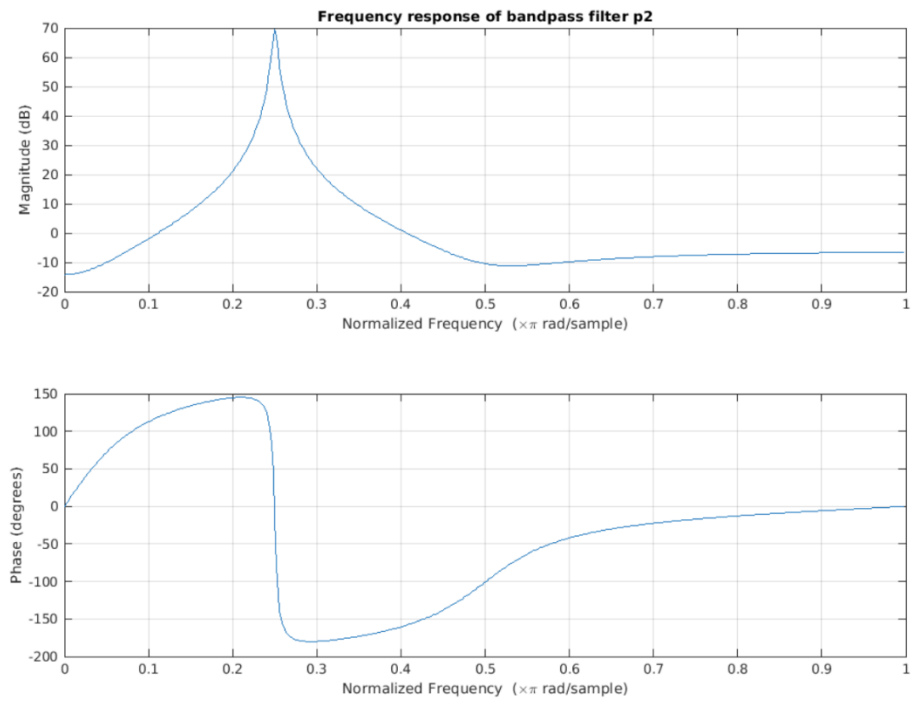
Η βηματική απόκριση είναι η παρακάτω:



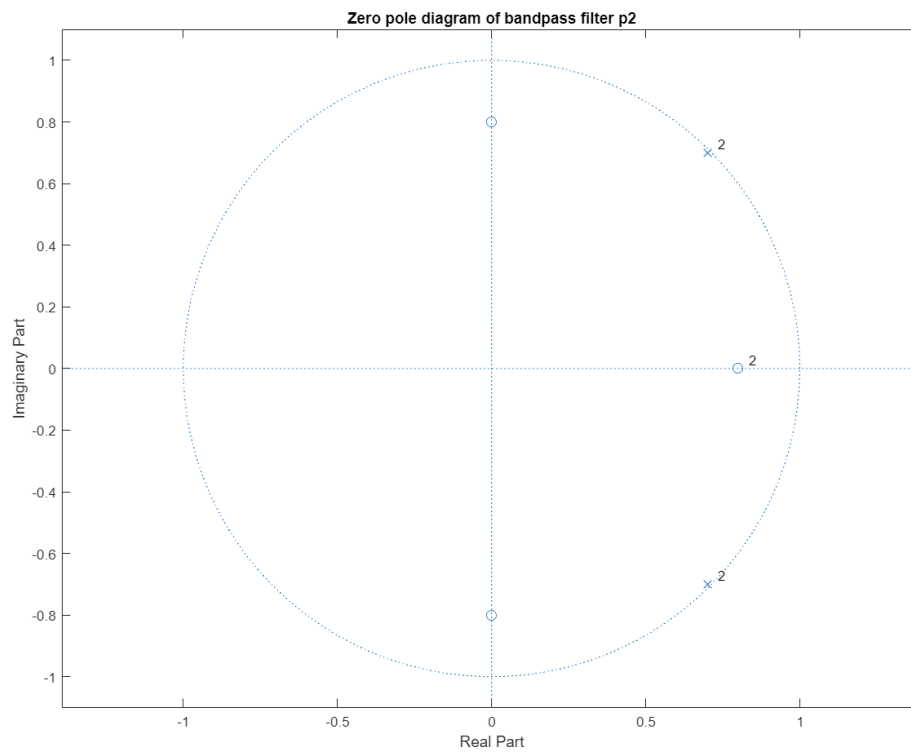
**1.2.δ)** Για την περίπτωση όπου οι πόλοι έχουν μετακινηθεί στις θέσεις  $(0.7, 0.7i)$ ,  $(0.7, -0.7i)$  έχουμε την παρακάτω κρουστική απόκριση:



Ενώ η απόκριση πλάτους και φάσης είναι:

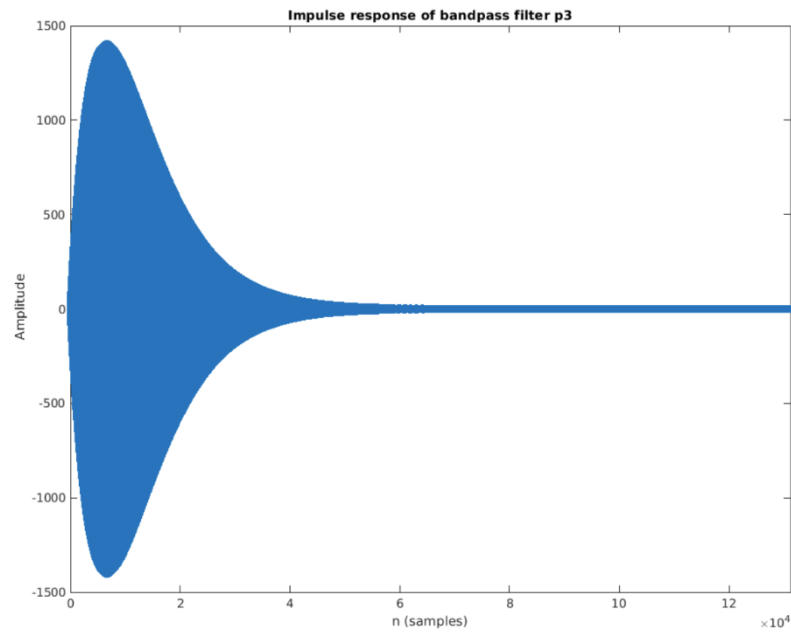


Το διάγραμμα πόλων και μηδενικών:

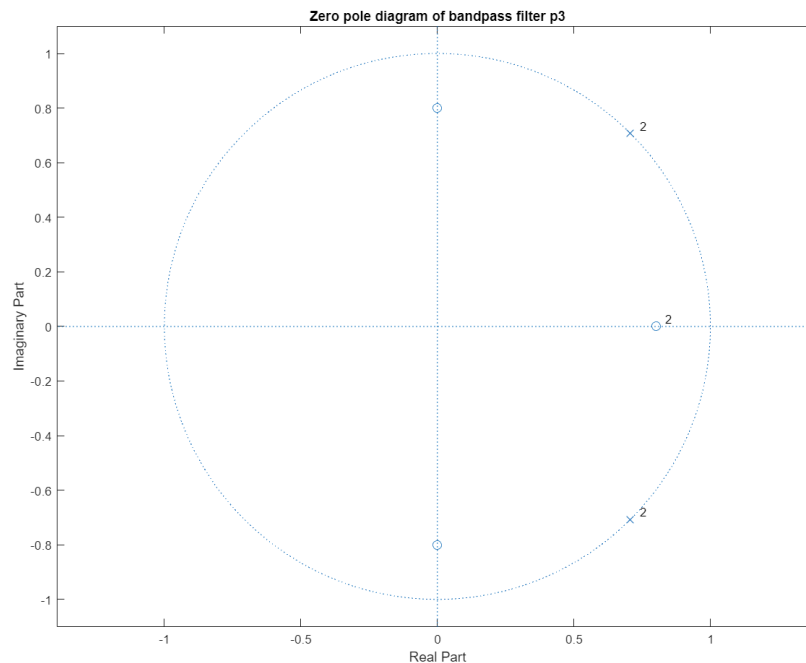


Τόσο από το παραπάνω διάγραμμα της κρουστικής απόκρισης, όπου για  $h[n] \xrightarrow{n \rightarrow \infty} 0$ , όσο και από το ότι η απόσταση των πόλων από την αρχή των αξόνων στο διάγραμμα πόλων μηδενικών είναι:  $|0.7 \pm 0.7i| \cong 0.99$  προκύπτει ότι το φίλτρο αυτό είναι οριακά ευσταθές.

Για την περίπτωση όπου οι πόλοι έχουν μετακινηθεί στις θέσεις  $(0.707, 0.707i)$ ,  $(0.707, -0.707i)$  έχουμε την παρακάτω κρουστική απόκριση:

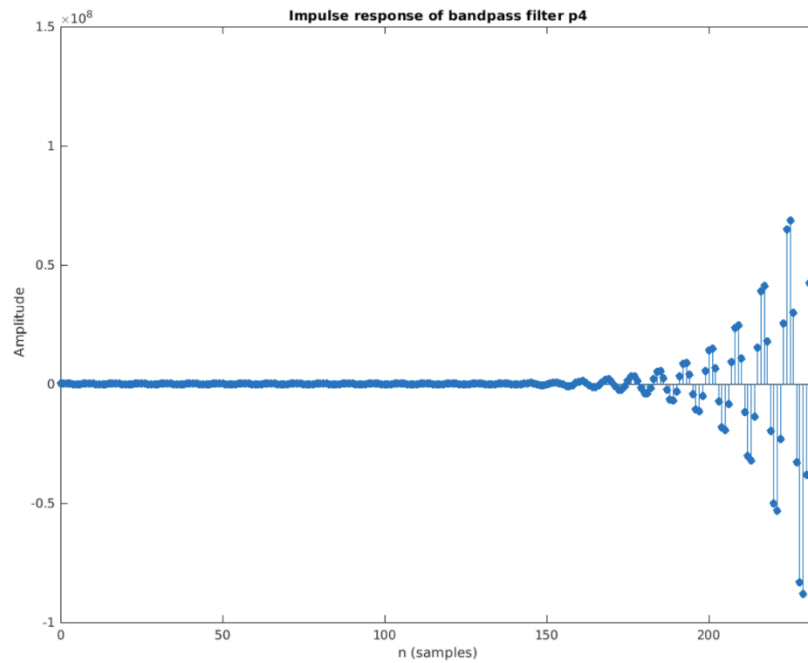


Το διάγραμμα πόλων και μηδενικών του φίλτρου είναι:

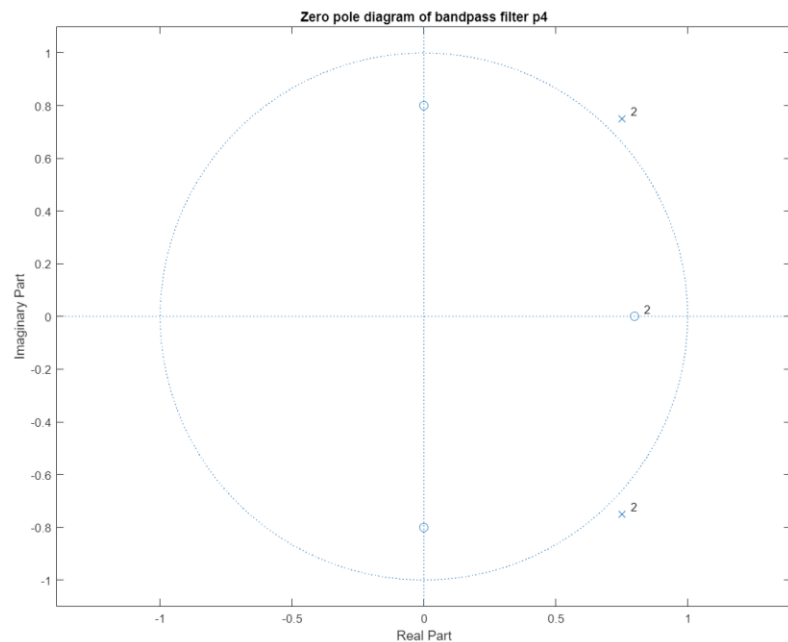


Από την κρουστική απόκριση η οποία μηδενίζεται για  $n \rightarrow \infty$ , από το διάγραμμα πόλων-μηδενικών αλλά και από τον υπολογισμό της απόστασης των πόλων από το  $(0, 0)$ :  $|0.707 \pm 0.707i| \cong 0.999$  έχουμε ότι το παραπάνω φίλτρο είναι οριακά ευσταθές.

Τέλος, για την περίπτωση όπου οι πόλοι έχουν μετατοπιστεί στις θέσεις  $(0.75, 0.75i)$ ,  $(0.75, -0.75i)$  έχουμε την εξής κρουστική απόκριση:



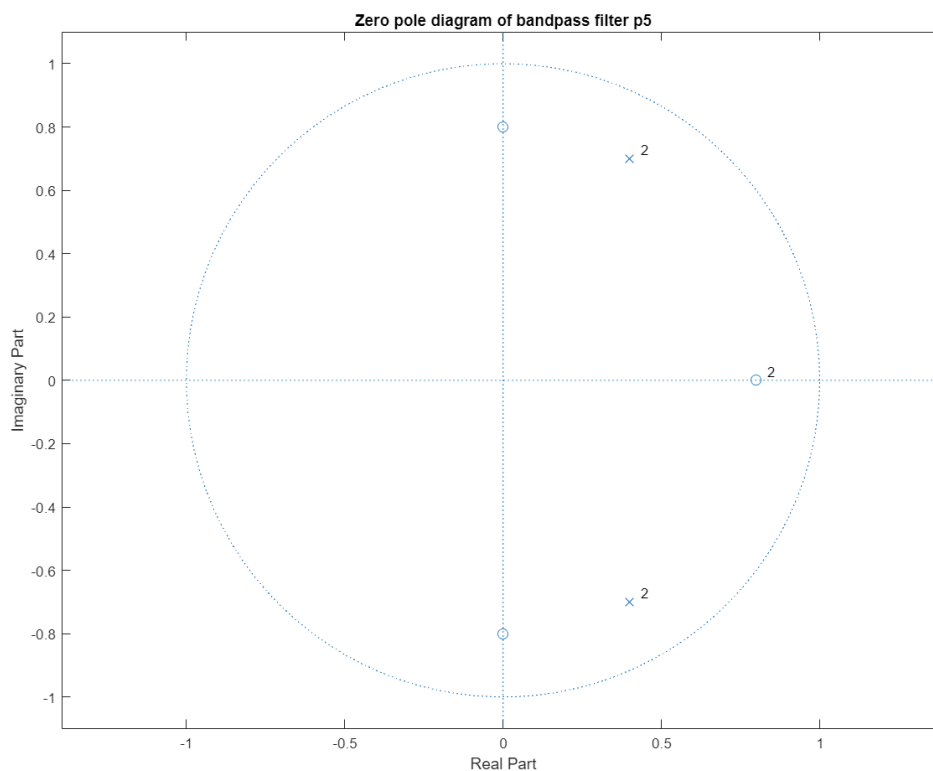
Και το παρακάτω διάγραμμα πόλων-μηδενικών:



Τόσο από την μορφή της κρουστικής απόκρισης που απειρίζεται καθώς  $n \rightarrow \infty$ , όσο και από το γεγονός ότι οι πόλοι βρίσκονται εκτός του μοναδιαίου κύκλου σύμφωνα με το διάγραμμα πόλων και μηδενικών καταλήγουμε στο συμπέρασμα ότι το φίλτρο στην περίπτωση αυτή είναι ασταθές.

**1.2.ε)** Στην περίπτωση αυτή έχουμε ότι οι πόλοι αποτελούν ένα διπλό ζεύγος στις θέσεις  $(0.4, 0.7i)$  και  $(0.4, -0.7i)$  και τα μηδενικά αυτά του ερωτήματος (α).

Το διάγραμμα πόλων και μηδενικών που προέκυψε μετά από χρήση της συνάρτησης `zplane()` είναι:



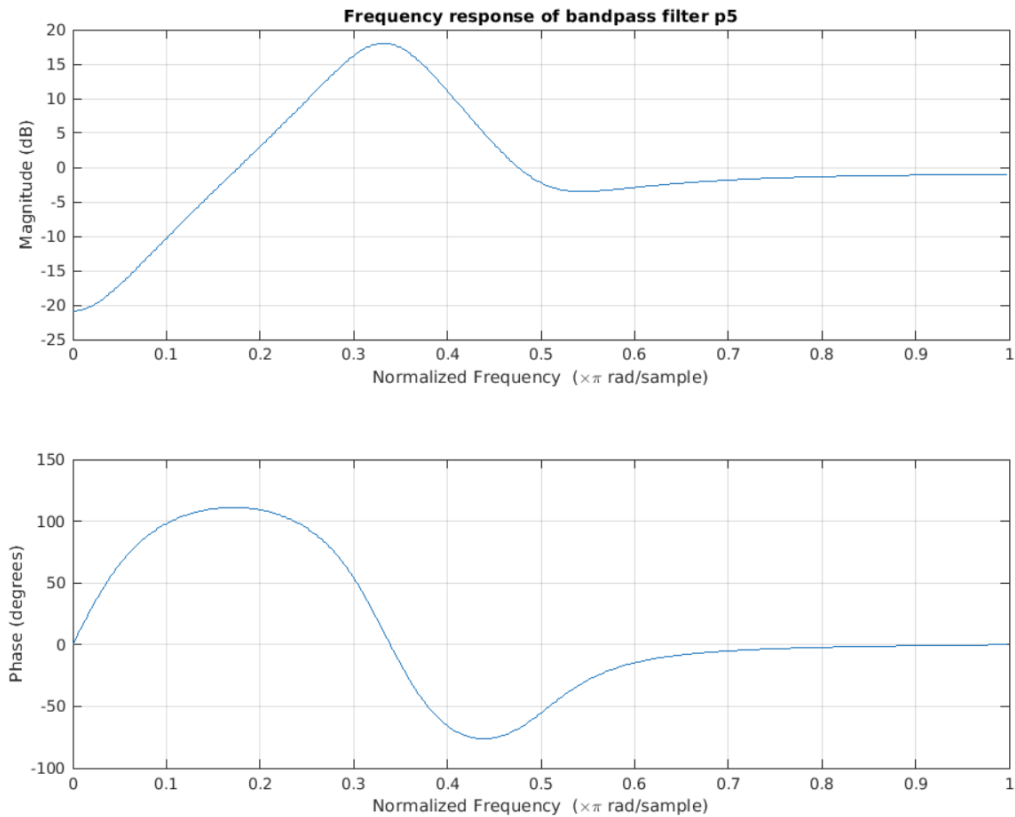
Με χρήση της συνάρτησης `zp2tf()` βρέθηκε ότι τα διανύσματα  $a$ ,  $b$  για το φίλτρο είναι:

$a = [1, -1.6, 1.94, -1.04, 0.4225]$

$b = [1, -1.6, 1.28, -1.024, 0.4096]$



Η απόκριση πλάτους και φάσης του φίλτρου σχεδιάστηκε μέσω της `freqz()` και είναι:



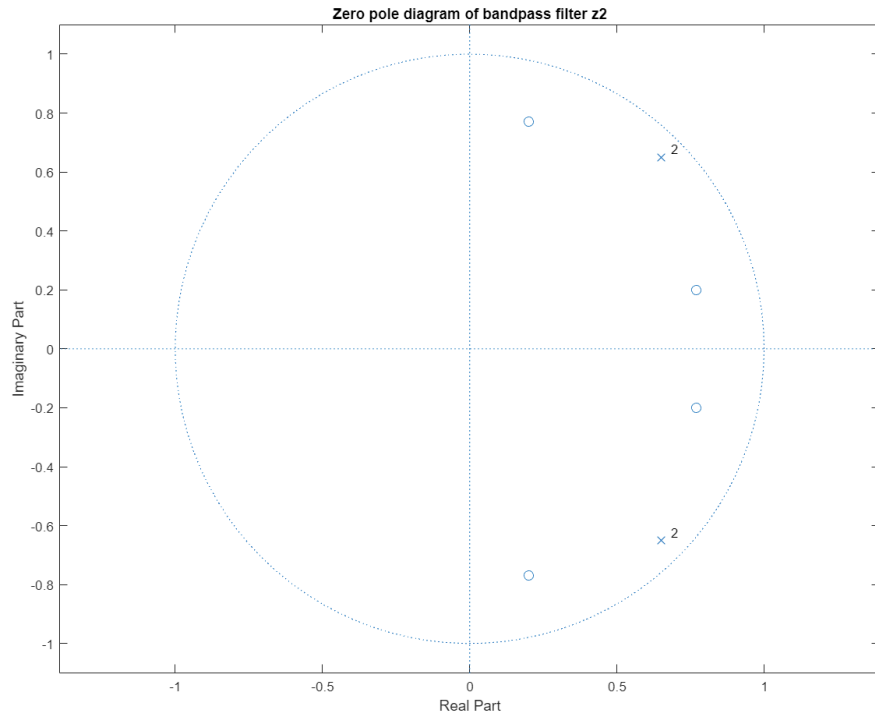
Για τη ζώνη διέλευσης του φίλτρου μπορούμε να παρατηρήσουμε από την απόκριση πλάτους ότι οι συχνότητες που είναι μεγαλύτερες από αυτή που αντιστοιχεί στο μέγιστο του γραφήματος δεν αποκόπτονται σε ικανοποιητικό βαθμό και άρα έχουμε μεγαλύτερο εύρος ζώνης διέλευσης σε σχέση με το αρχικό φίλτρο.

Γενικά, παρατηρήθηκε μετά την δοκιμή και άλλων συνδιασμών πόλων πως όσο πλησιέστερα βρίσκονται οι πόλοι στο  $(0, 0)$  τόσο μεγαλύτερο είναι και το εύρος της ζώνης διέλευσης του φίλτρου.

Το ότι στο παραπάνω φίλτρο περνούν συχνότητες μεγαλύτερες από αυτή που αντιστοιχεί στο μέγιστο οφείλεται στην ασύμμετρη απόσταση που απέχουν τα ζεύγη των πόλων από τα δύο ζεύγη μηδενικών.

**1.2.στ)** Στην πρώτη περίπτωση έχουμε ότι οι πόλοι παραμένουν στην ίδια θέση όπως και στο ερώτημα (α), ενώ τα μηδενικά βρίσκονται στις θέσεις:  $(0.77, 0.2i)$ ,  $(0.77, -0.2i)$ ,  $(0.2, 0.77i)$  και  $(0.2, -0.77i)$ .

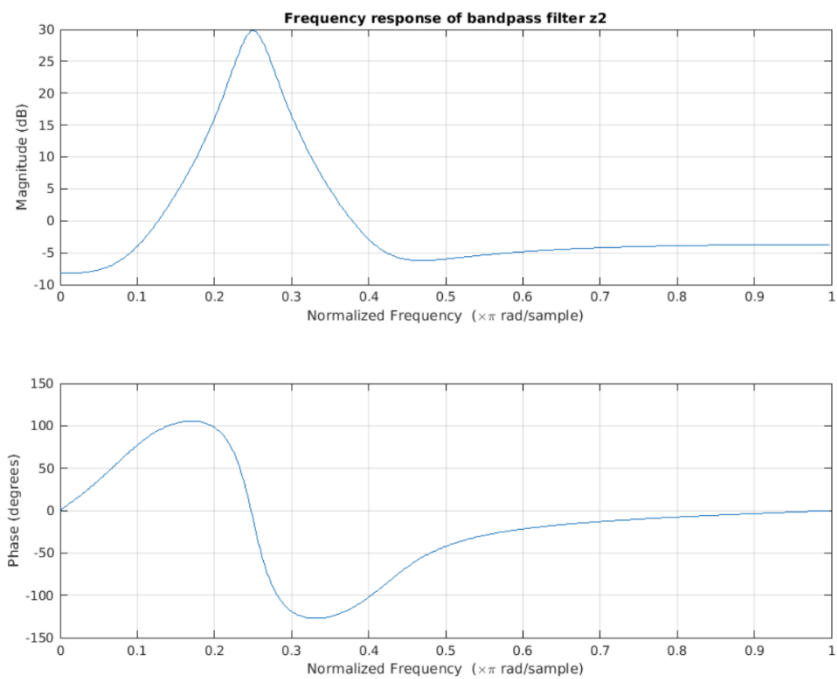
Το διάγραμμα πόλων και μηδενικών είναι:



Τα  $a$ ,  $b$  είναι:

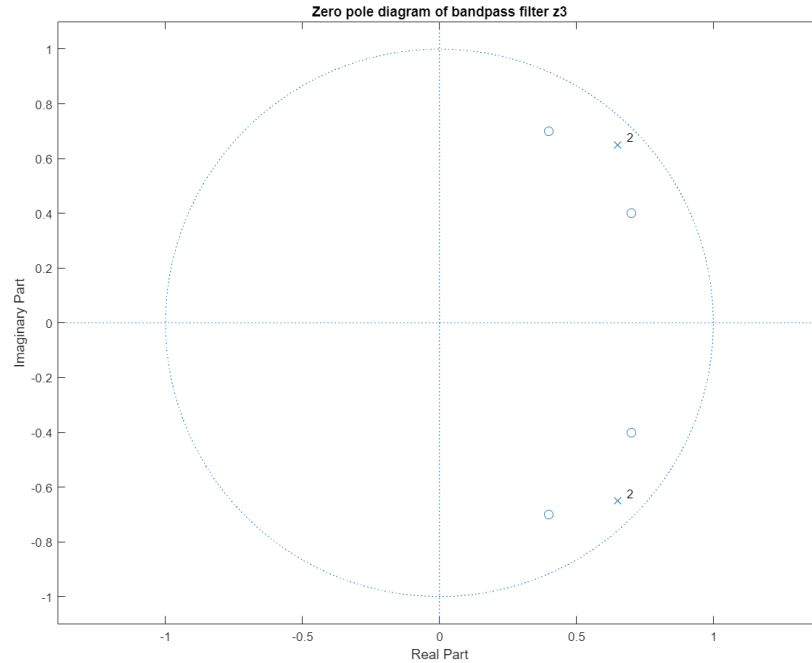
$$a = [1, -2.6, 3.38, -2.197, 0.714] \quad b = [1, -1.94, 1.8818, -1.2278, 0.4006]$$

Η απόκριση πλάτους και φάσης είναι:



Στη δεύτερη περίπτωση έχουμε και πάλι τους αρχικούς πόλους αλλά τα μηδενικά βρίσκονται στις θέσεις:  $(0.4, 0.7i)$ ,  $(0.4, -0.7i)$ ,  $(0.7, 0.4i)$  και  $(0.7, -0.4i)$ .

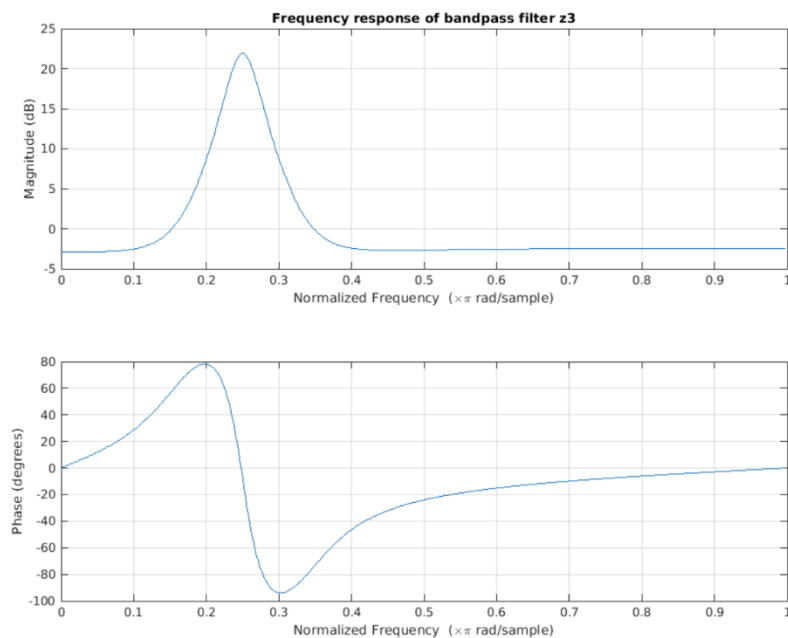
Το διάγραμμα πόλων-μηδενικών είναι:



Τα  $a$ ,  $b$  είναι:

$$a = [1, -2.6, 3.38, -2.197, 0.714] \quad b = [1, -2.2, 2.42, -1.43, 0.4225]$$

Η απόκριση πλάτους και φάσης είναι:

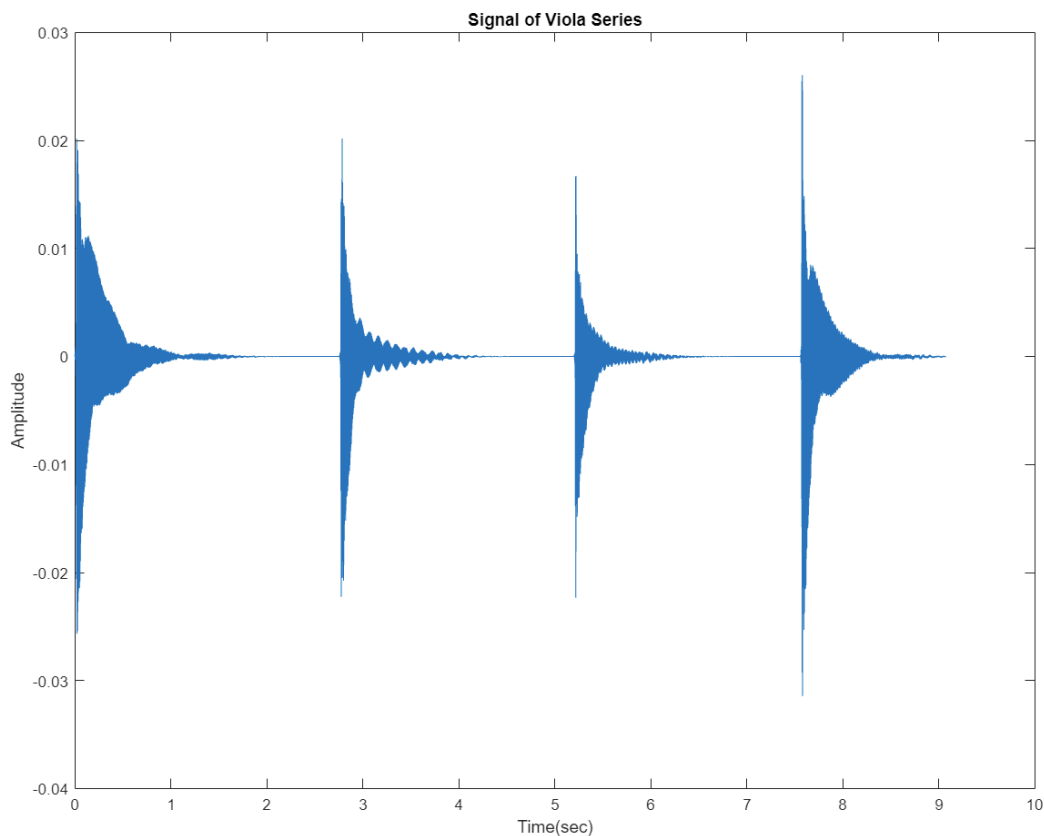


Παρατηρώντας και συγκρίνοντας τις αποκρίσεις πλάτους των παραπάνω φίλτρων σε σχέση με τα αντίστοιχα διαγράμματα πόλων και μηδενικών συμπεραίνουμε ότι όσο πλησιάζουν τα ζεύγη των μηδενικών τους πόλους τόσο μειώνεται το gain όπως και η ζώνη διέλευσης του φίλτρου.

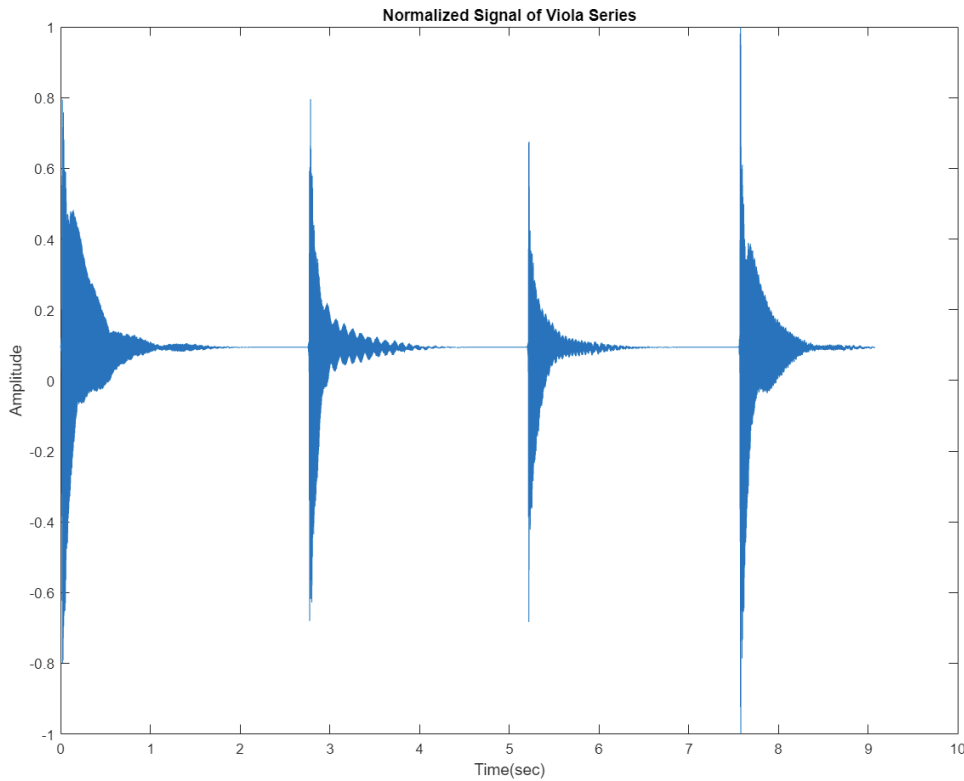
## 2 Ανάλυση Μουσικών Σημάτων και Εφαρμογή Φίλτρων

### 2.1 Ανάλυση Μουσικών Σημάτων

**2.1.α)** Μέσω της εντολής `audioread()` φορτώνουμε το αρχείο *viola\_series.wav* στο Matlab με συχνότητα δειγματοληψίας 44.1 kHz. Στη συνέχεια, ακούστηκε το σήμα μέσω της εντολής `sound()` όπου ακούστηκαν 4 νότες βιόλας. Τέλος, μέσω της `plot()` σχεδιάστηκε το σήμα και είναι το παρακάτω:



**2.1.β)** Αρχικά κανονικοποιούμε το σήμα μας στο διάστημα  $[-1, 1]$  μέσω της εντολής `normalize()` και προκύπτει το εξής αποτέλεσμα:



Για τον υπολογισμό της ενέργειας γνωρίζουμε ότι ισχύει ο παρακάτω συνελκτικός τύπος:

$$E[n] = \sum_{m=0}^M x^2[m] w[n - m] \quad (2.1.1)$$

Όπου  $w[n]$  είναι το παράθυρο Hamming που δίνεται από την εξίσωση:

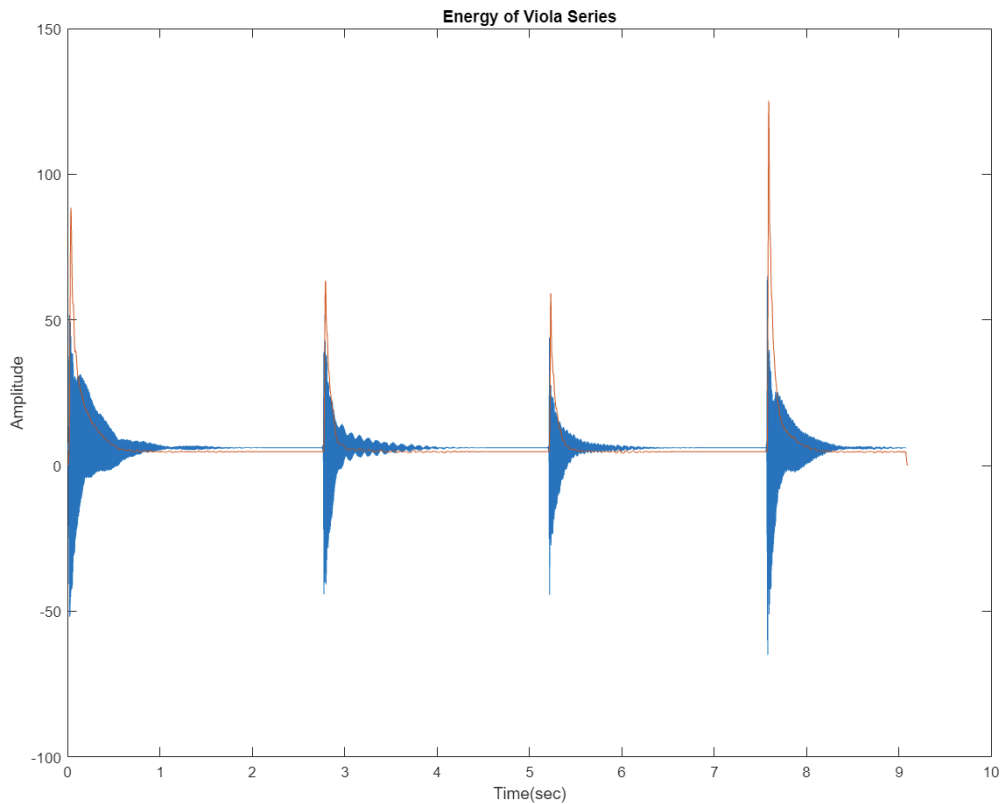
$$w[n] = 0.54 + 0.46 \cos\left(\frac{2\pi n}{N}\right) \quad (2.1.2)$$

και χρησιμοποιούμε  $N = 1000$  δείγματα.

Ωστόσο υπάρχει και κατάλληλη συνάρτηση `hamming()` στο Matlab η οποία μας επιτρέπει τον υπολογισμό της ταχύτερα.

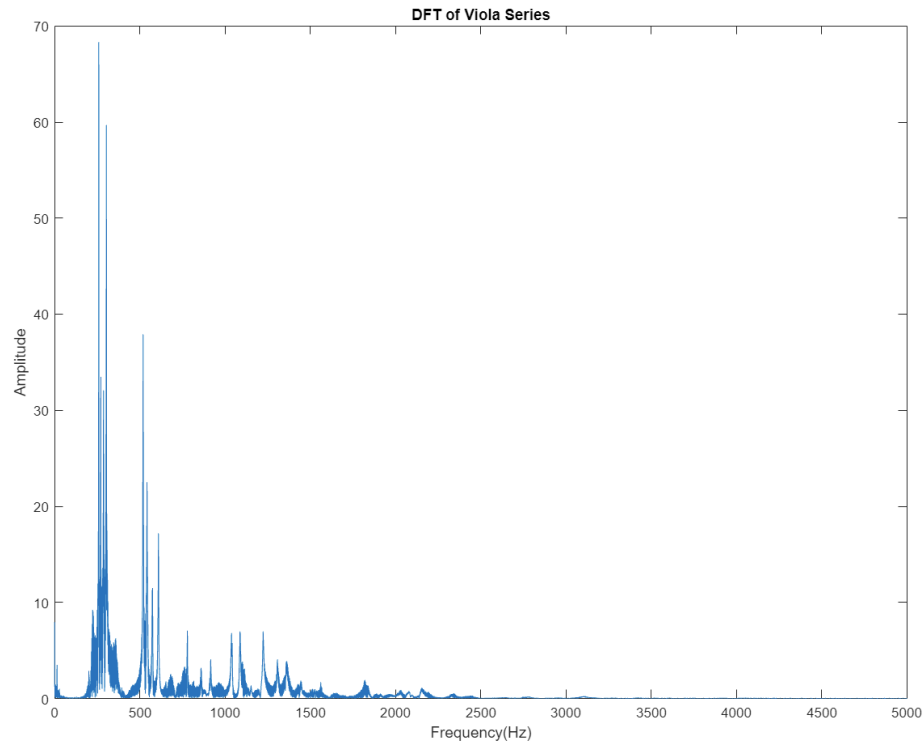
Έτσι, αφού πρώτα υπολογίσαμε το παράθυρο Hamming, υπολογίστηκε η ενέργεια του σήματος πραγματοποιώντας τη συνέλιξη με το κανονικοποιημένο σήμα στο  $[-1, 1]$  υψωμένο στο τετράγωνο και στη συνέχεια κανονικοποιήθηκε και πάλι το σήμα “Viola Series” στο διάστημα  $[-65, 65]$  ώστε να είναι ορατό όταν σχεδιαστεί μαζί με την ενέργεια του σήματος στο ίδιο διάγραμμα.

Η ενέργεια του σήματος μαζί με το κανονικοποιημένο σήμα δίνονται από το παρακάτω διάγραμμα:

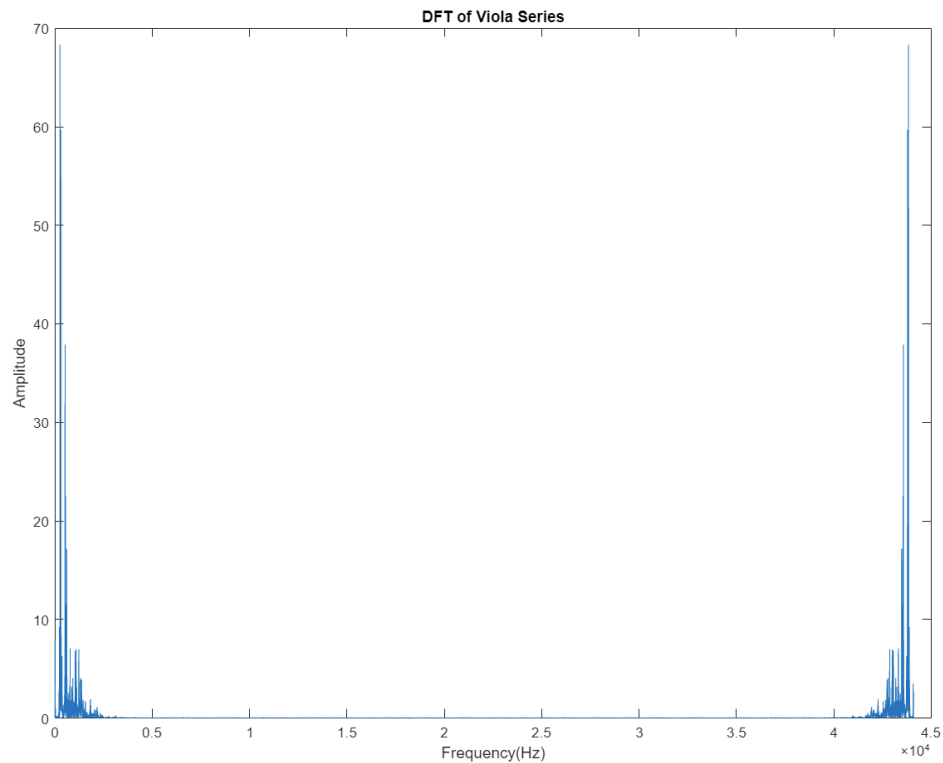


Σύμφωνα με το διάγραμμα μπορούμε να δούμε όπως είναι αναμενόμενο ότι η ενέργεια είναι μη αρνητική και πως στα σημεία όπου το σήμα έχει μεγαλύτερη ένταση εκεί είναι και μεγαλύτερη η ενέργεια. Πέρα από τη διαισθητική επιβεβαίωση, με αυτόν τον τρόπο επιβεβαιώνεται και η συνελκτική σχέση μεταξύ της ενέργειας και του σήματός, αφού όσο μεγαλύτερες είναι οι τιμές του σήματος, τόσο μεγαλύτερες και οι τιμές της ενέργειας.

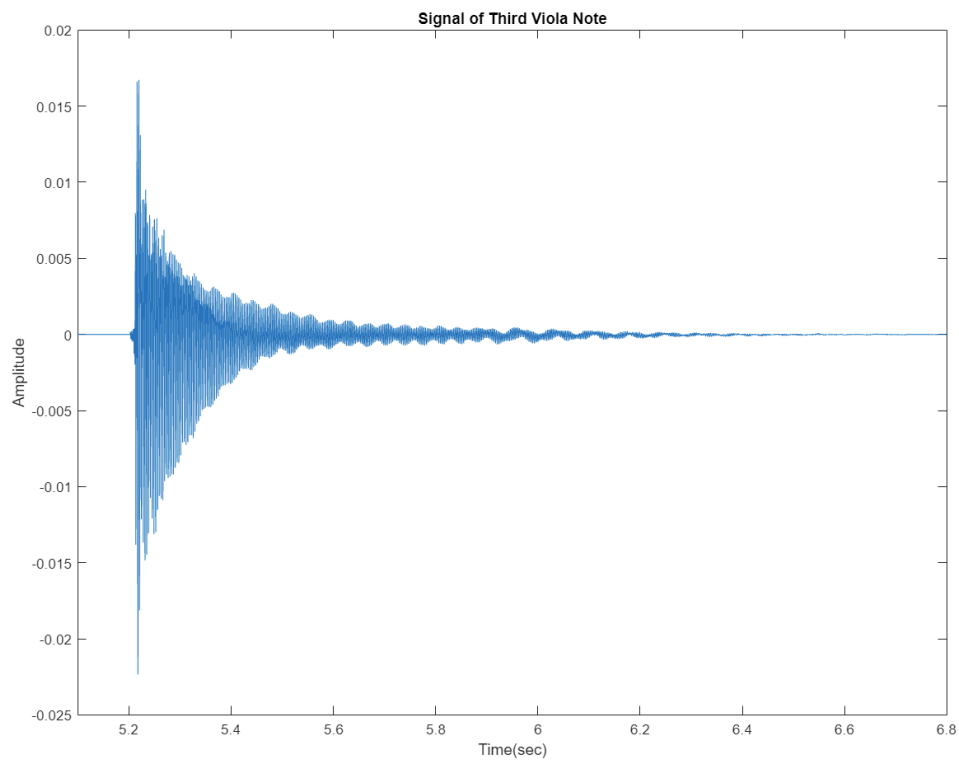
**2.1.γ)** Για να υπολογίσουμε τον DFT του σήματος “Viola Series” χρησιμοποιούμε την εντολή `fft()`. Για το σχεδιασμό του μέτρου του φάσματος χρησιμοποιήθηκε η `plot()` και η `abs()` και το αποτέλεσμα είναι το παρακάτω:



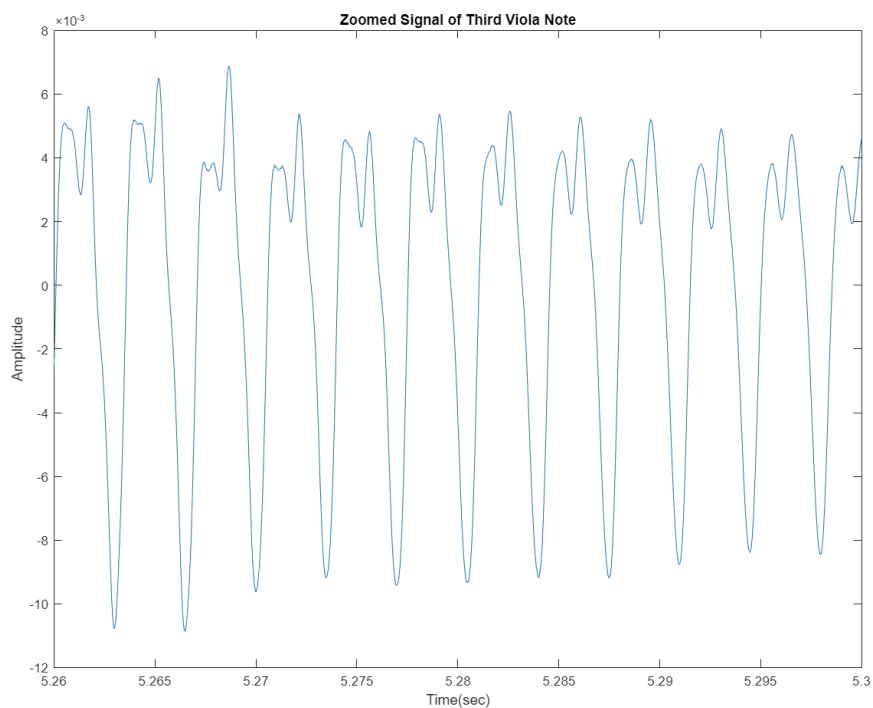
Γνωρίζουμε ότι ο DFT είναι συμμετρικός και για αυτό το λόγο έχουμε απομονώσει μέσω της εντολής `xlim([0 5000])` μόνο ένα κομμάτι του αριστερού τμήματος αυτού. Χωρίς την προηγούμενη εντολή εμφανίζεται ολόκληρο το συμμετρικό φάσμα και είναι:



**2.1.δ)** Στο ερώτημα αυτό απομονώθηκε η 3<sup>η</sup> νότα μεταξύ των 5.1 - 6.8 sec μέσω της εντολής xlim(). Η απομονωμένη νότα σχεδιάστηκε μέσω της εντολής plot() και είναι η εξής:



Παρατίθεται και μεγεθυμένη απεικόνιση της τρίτης νότας στο διάστημα (5.63, 5.3) μέσω της οποίας υπολογίστηκε εποπτικά η περίοδος:

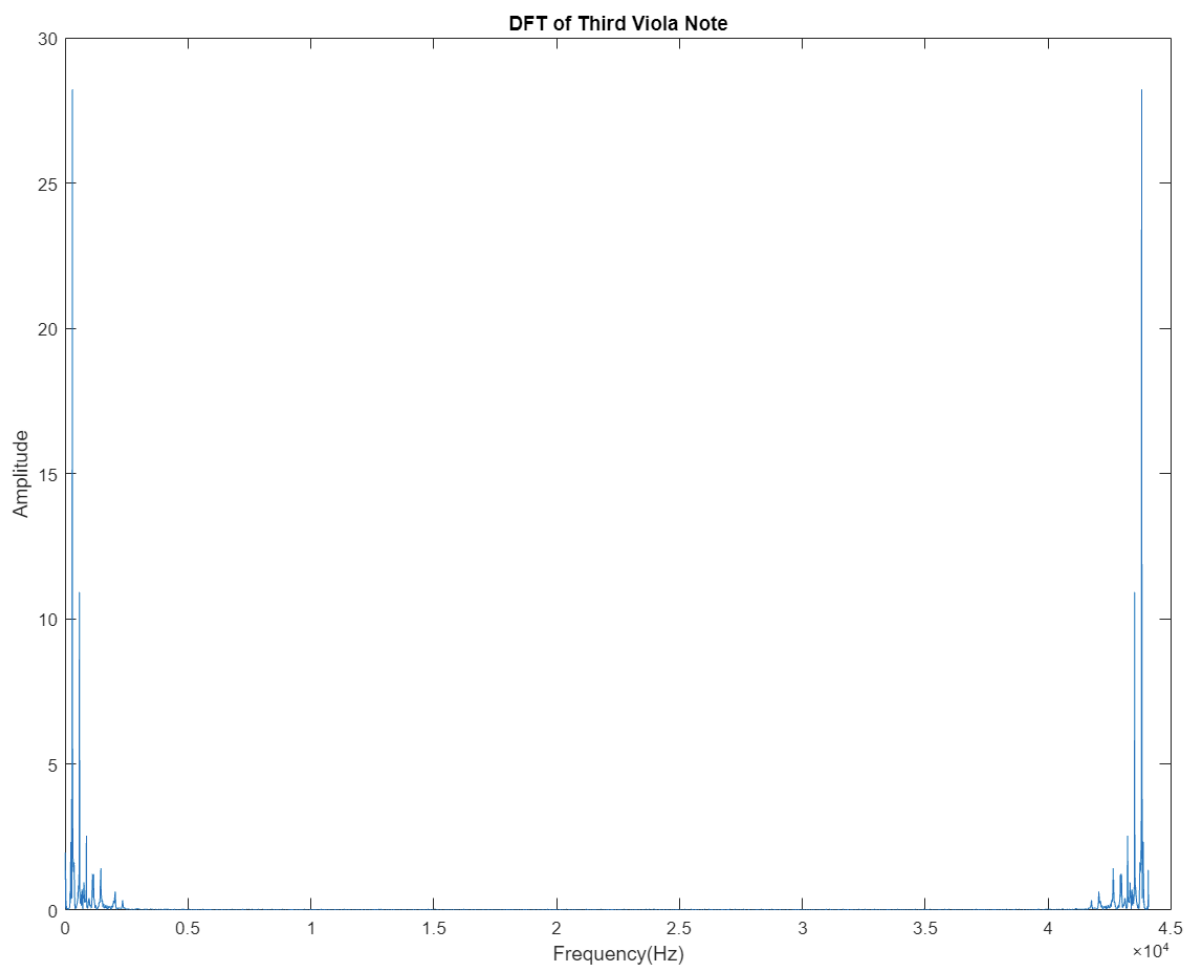


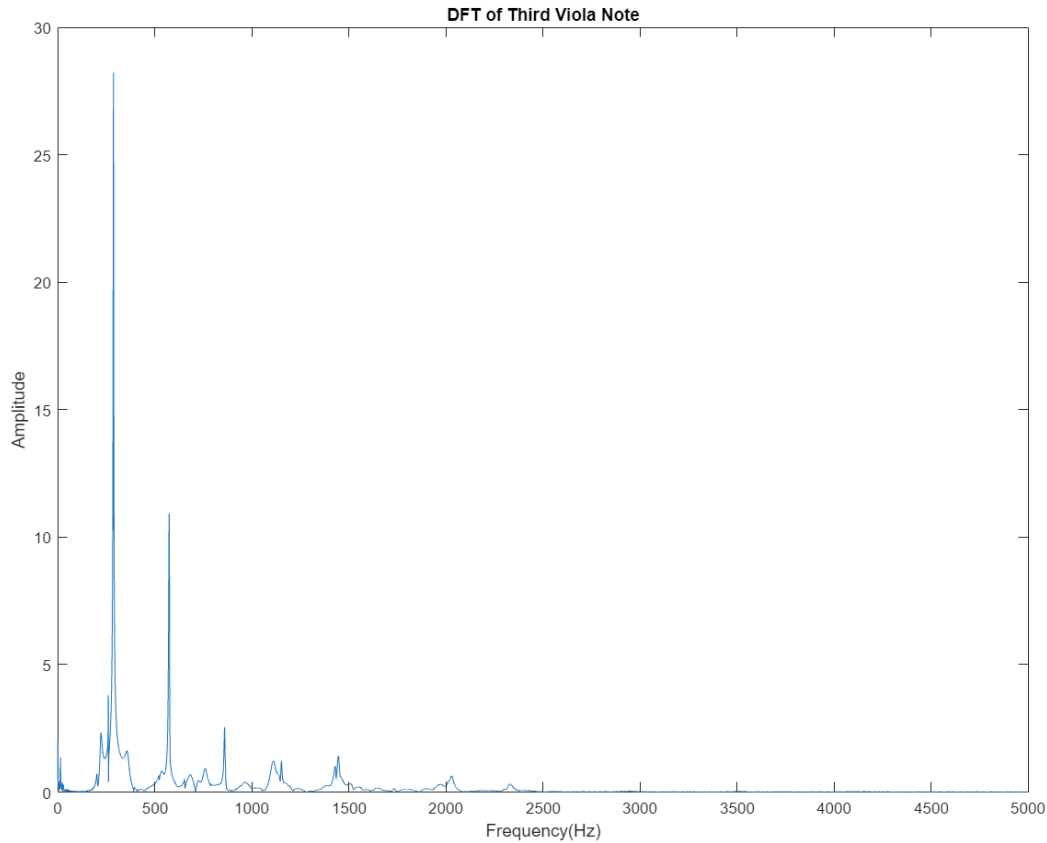


Η περίοδος σε ms υπολογίστηκε **3.49ms** ενώ σε δείγματα  $3.49ms \times 44.1kHz = 153.909 =$   
**154 δείγματα.**

**2.1.ε)** Για τον υπολογισμό του DFT της 3<sup>ης</sup> νότας και τον σχεδιασμό του μέτρου του φάσματος χρησιμοποιήθηκαν όπως και στο ερώτημα (γ) οι εντολές `fft()`, `plot()` και `abs()`, αφού όμως πρώτα απομονώσαμε το αρχικό σήμα στο διάστημα (5.1, 6.8).

Παρατίθενται τόσο ολόκληρο το συμμετρικό φάσμα όσο και η μεγεθυμένη αριστερή πλευρά του στο διάστημα (0, 5000):

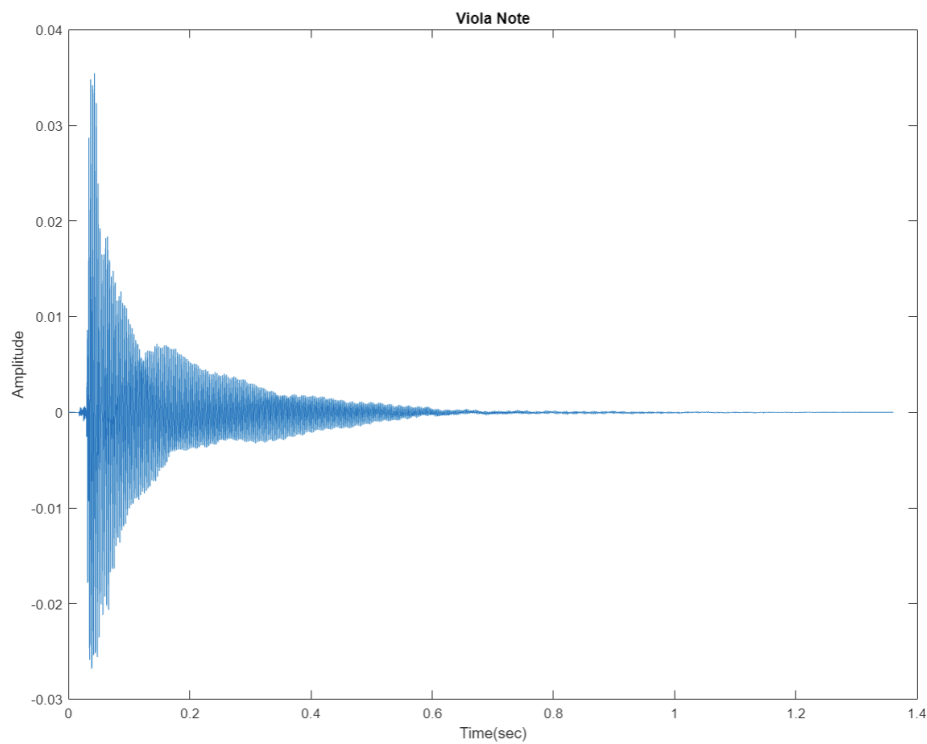




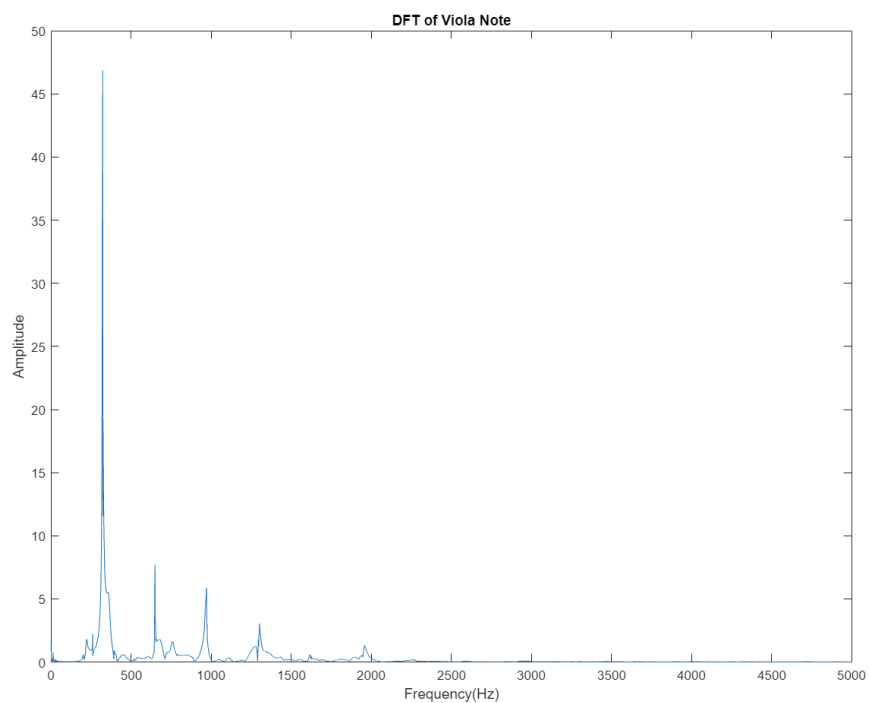
Η θεμελιώδης συχνότητα του φάσματος ταυτίζεται με την συχνότητα που αντιστοιχεί στην πρώτη αρμονική και είναι **287 Hz**. Για να επιβεβαιώσουμε τη σχέση μεταξύ θεμελιώδους συχνότητας με την περίοδο που υπολογίσαμε στο προηγούμενο ερώτημα θα πρέπει να ισχύει:  $287 = \frac{1}{3.49 \times 10^{-3}}$   
 $\Rightarrow 287 \cong 286.5$  το οποίο ισχύει και η μικρή απόκλιση οφείλεται στην εποπτική εύρεση της περιόδου. Άρα το παραπάνω φάσμα είναι ορθό και η θεμελιώδης συχνότητα αντιστοιχεί στην περίοδο που βρήκαμε προηγουμένως.

Σχετικά με τη μορφή του φάσματος, μπορούμε να παρατηρήσουμε συγκρίνοντας και το φάσμα του αρχικού σήματος πως αποτελεί ένα μέρος αυτού. Συγκεκριμένα, το αρχικό φάσμα είναι το άθροισμα των φασμάτων για κάθε νότα, και το φάσμα της τρίτης νότας είναι το αρχικό φάσμα έχοντας όμως απομονώσει μόνο το κομμάτι που αντιστοιχεί στη νότα αυτή. Υπάρχουν και αρμονικές υψηλότερης τάξης από την πρώτη, ωστόσο το πλάτος τους μειώνεται εκθετικά.

**2.1.στ)** Αρχικά φορτώσαμε το αρχείο “*viola\_note.wav*” μέσω της εντολής `audioread()` και στη συνέχεια κάναμε το γράφημα στο πεδίο του χρόνου μέσω της `plot()` και είναι το παρακάτω:



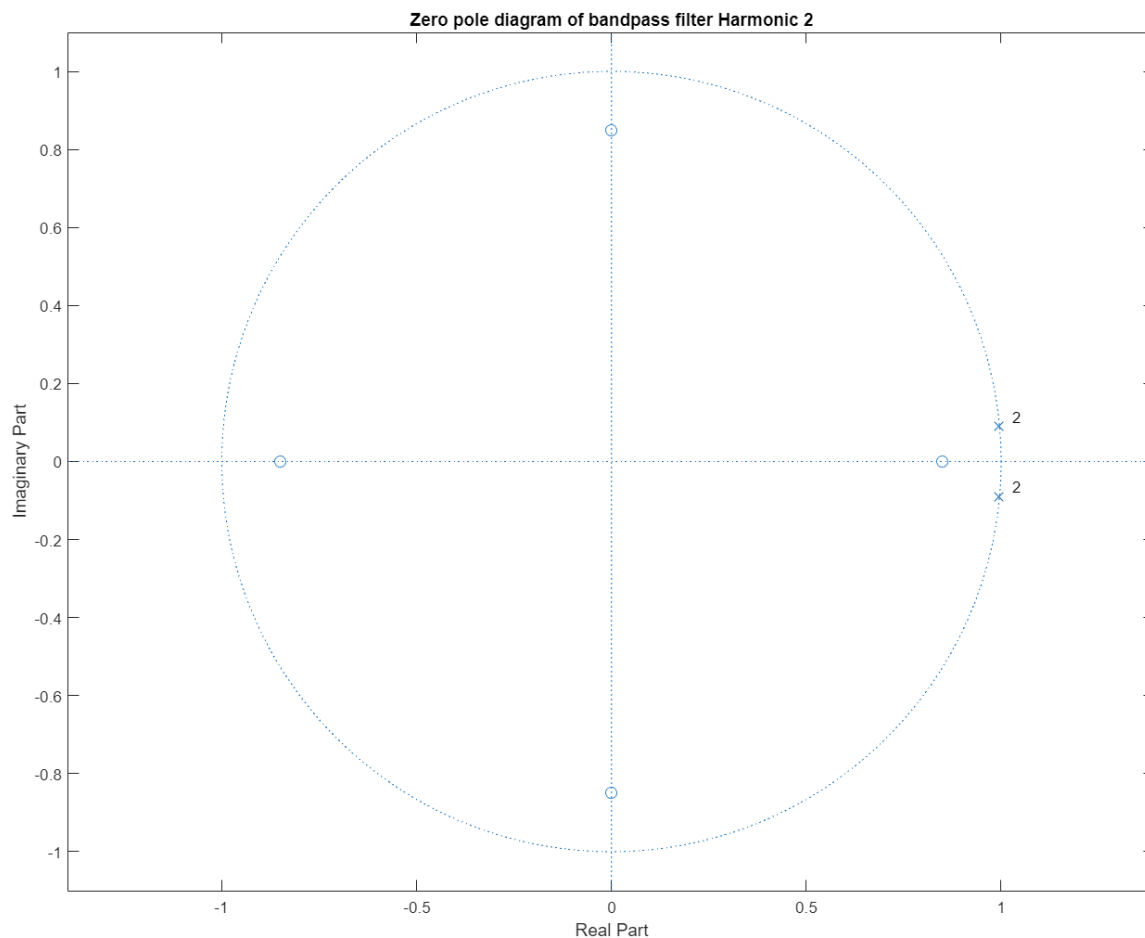
Μέσω του `fft()` εντοπίσαμε τον DFT της νότας το μέτρο του οποίου στο διάστημα (0, 5000) είναι το παρακάτω:



Από το παραπάνω φάσμα εντοπίζουμε την θεμελιώδη συχνότητα η οποία είναι 321.93 Hz. Θέλουμε να απομονώσουμε την 2<sup>η</sup> και την 4<sup>η</sup> αρμονική οι οποίες είναι στα 648.27 Hz και 1031.68 Hz αντίστοιχα. Για να το καταφέρουμε αυτό θα ακολουθήσουμε τη μεθοδολογία της 1<sup>ης</sup> άσκησης ώστε να κατασκευάσουμε δύο κατάλληλα bandpass φίλτρα.

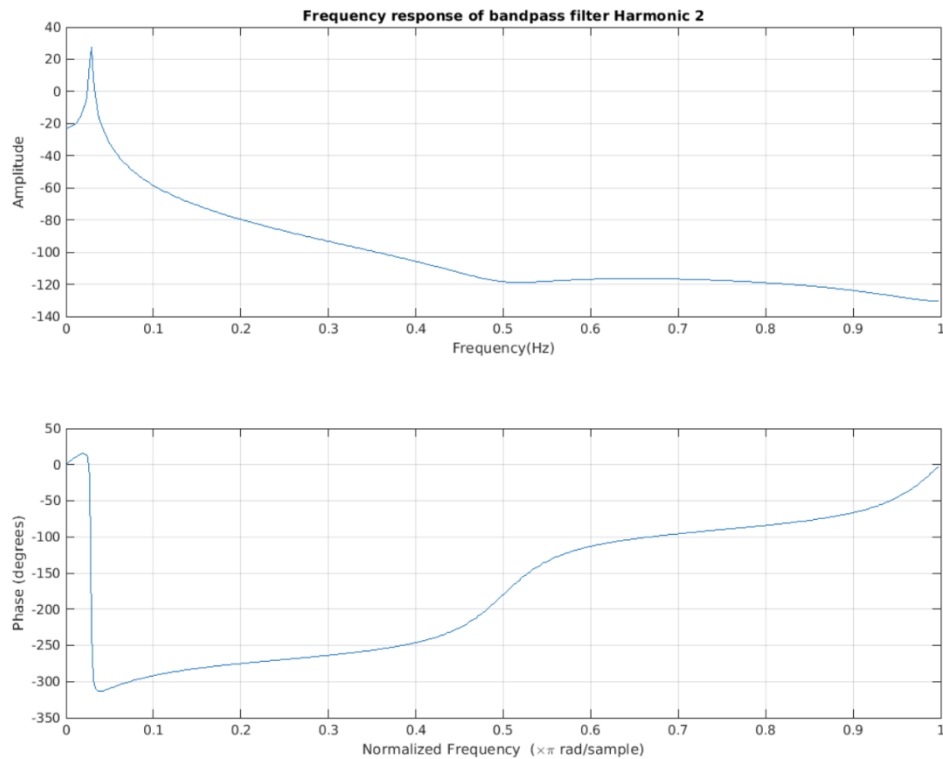
Για το φίλτρο απομόνωσης της 2<sup>ης</sup> αρμονικής έχουμε:

Μετά από δοκιμές και εκμεταλλεόμενοι τα αποτελέσματα της άσκησης 1 για το πως διαμορφώνεται το bandpass φίλτρο ανάλογα με τους πόλους και τα μηδενικά καταλήξαμε στο παρακάτω διάγραμμα πόλων-μηδενικών:



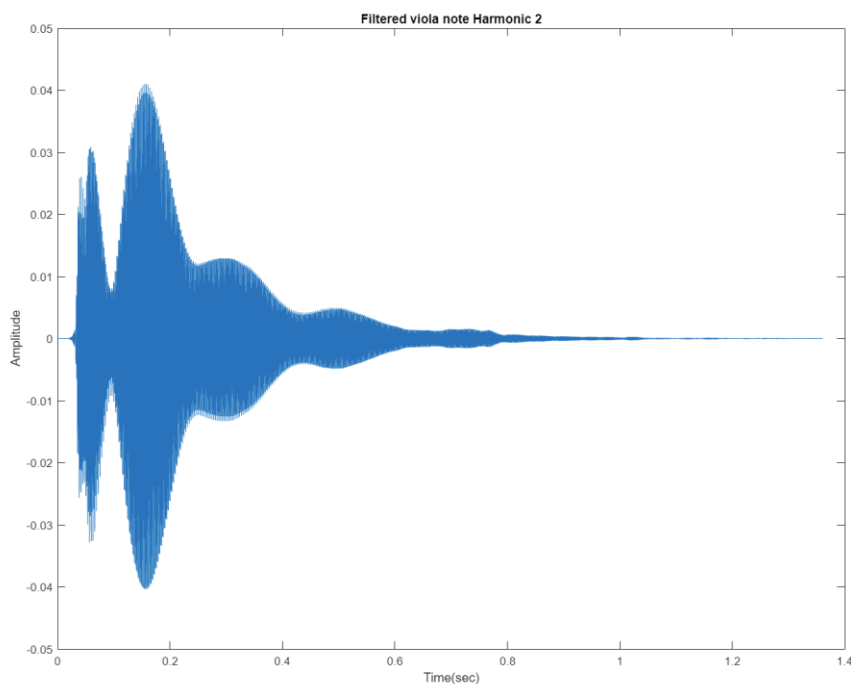
Οι πόλοι που επιλέχθηκαν είναι ένα διπλό ζεύγος συζυγών πόλων στα σημεία (0.994, 0.09), (0.994, -0.09) ενώ τα μηδενικά είναι ένα ζεύγος συζυγών στα σημεία (0.85, 0), (-0.85, 0) και ένα ακόμα ζεύγος συζυγών στα σημεία (0, 0.85) και (0, -0.85).

Η απόκριση πλάτους και φάσης του φίλτρου σχεδιάστηκε μέσω της συνάρτησης freqz(), αφού πρώτα υπολογίστηκαν τα διανύσματα a, b από την συνάρτηση zp2tf(), και είναι η παρακάτω:

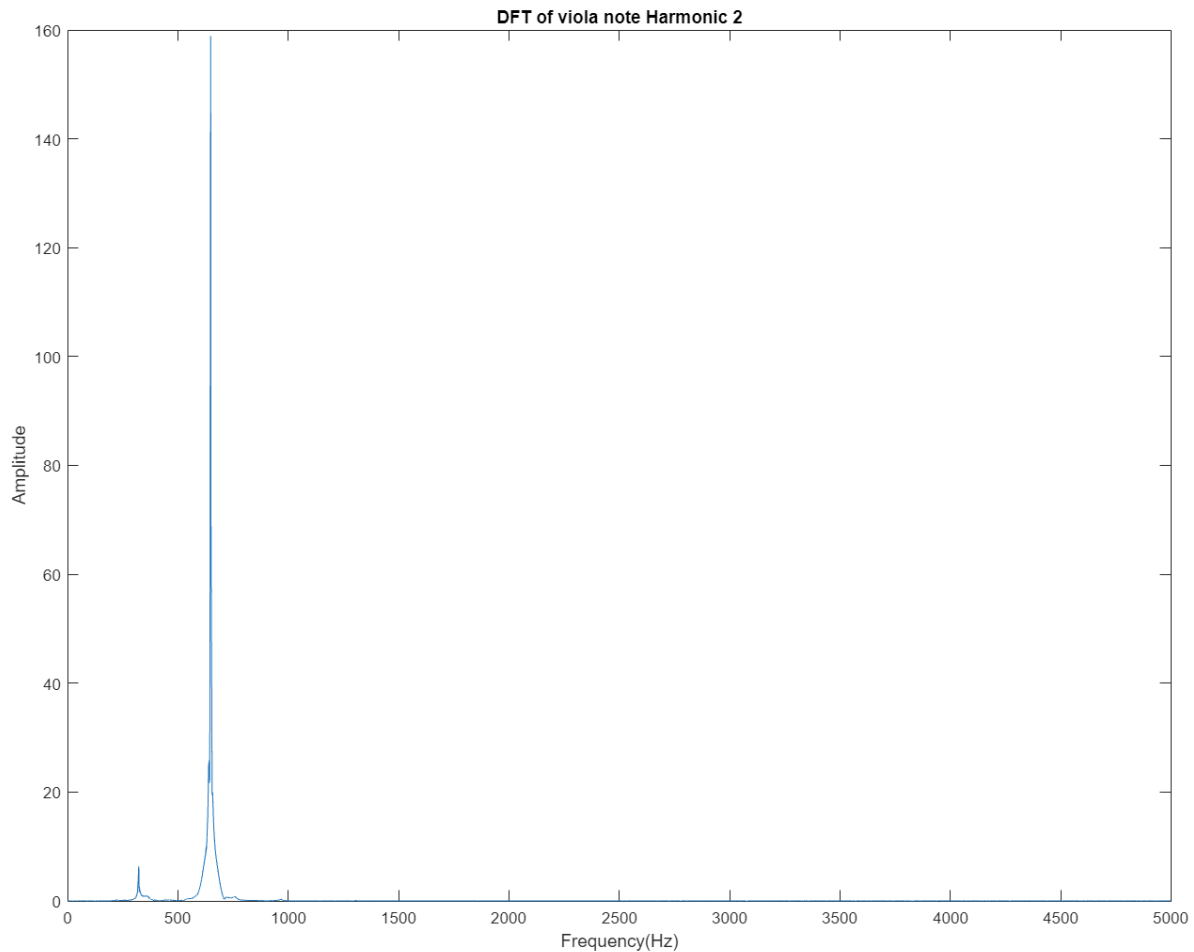


Όπως φαίνεται τόσο από το οριακό διάγραμμα της απόκρισης πλάτους και φάσης του bandpass φίλτρου που απεικονίζεται, όσο και από το ότι οι πόλοι βρίσκονται οριακά πάνω στον μοναδιαίο κύκλο συμπεραίνουμε ότι το φίλτρο που κατασκευάσαμε είναι οριακά ευσταθές.

Έπειτα μέσω της συνάρτησης filter() βρήκαμε την έξοδο του σήματος “Viola Note” έχοντας εφαρμόσει το παραπάνω bandpass φίλτρο. Η έξοδος έχει την εξής μορφή στο πεδίο του χρόνου:



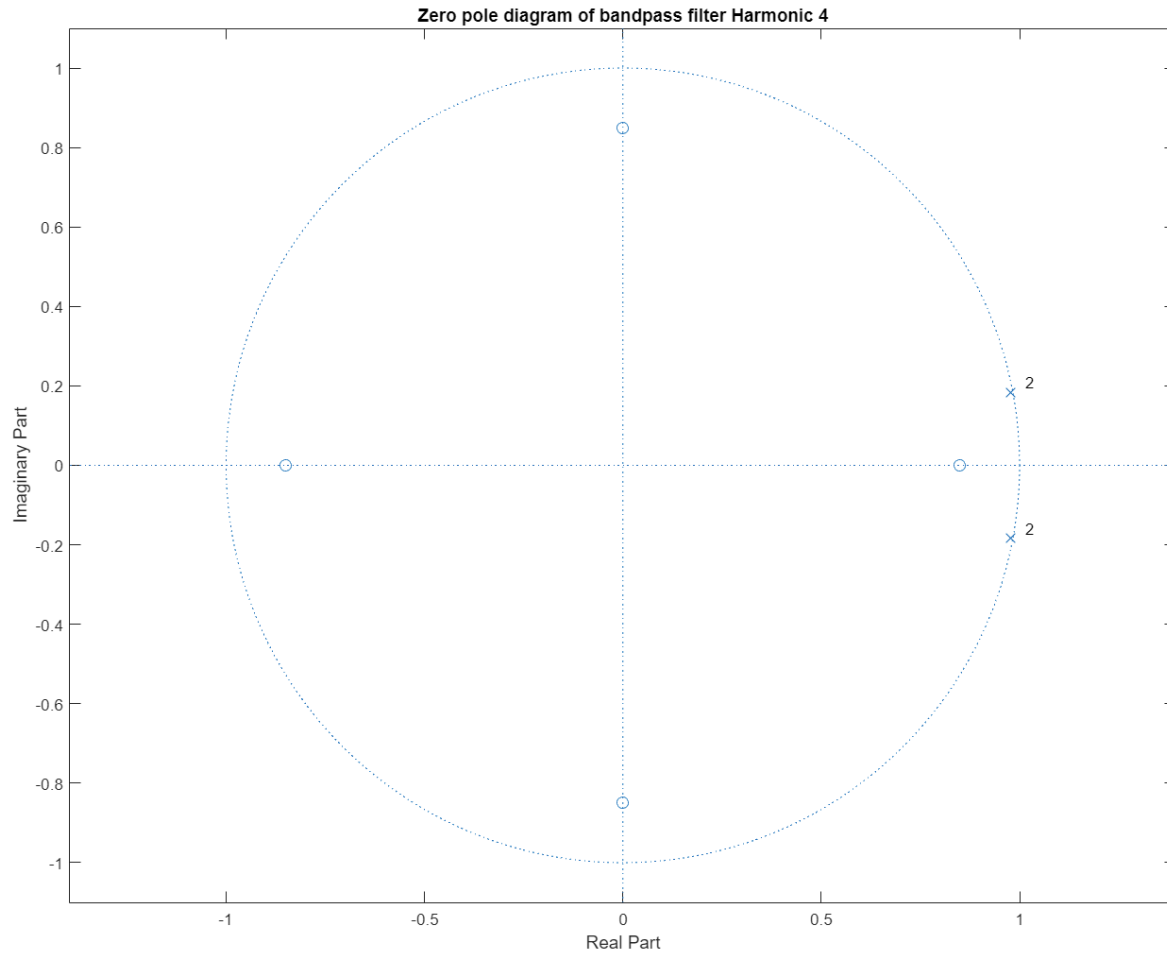
Ο DFT του παραπάνω φιλτραρισμένου σήματος εντοπίστηκε με τη βοήθεια της συνάρτησης `fft()` και δίνεται το μέτρο του φάσματος:



Επομένως, το αποτέλεσμα της εφαρμογής του φίλτρου ήταν να απομονώσει την συχνότητα των 648.27 Hz, δηλαδή επιτύχαμε να απομονώσουμε ακριβώς τη ζητούμενη συχνότητα. Το πλάτος της 2<sup>ης</sup> αρμονικής μετά την εφαρμογή του φίλτρου φτάνει στα 158.886 κάτι το οποίο μπορεί βέβαια να αυξομειωθεί ανάλογα με το `gain` που θα βάλουμε κατά την εκτέλεση της εντολής `zp2tf()` (στην συγκεκριμένη περίπτωση χρησιμοποιήθηκε `k = 0.00001`). Παρατηρούμε ότι περνούν σε ελάχιστο βαθμό και κάποιες άλλες συχνότητες οι οποίες βρίσκονται στην περιοχή της 1<sup>ης</sup> αρμονικής.

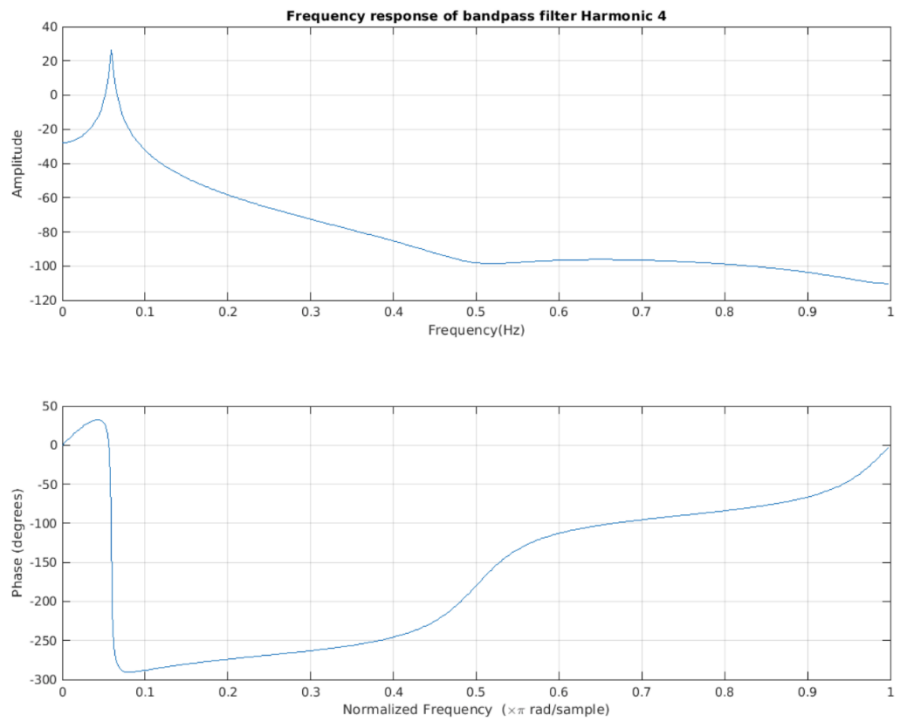
Για την περίπτωση όπου θέλουμε να απομονώσουμε την 4<sup>η</sup> αρμονική θα ακολουθήσουμε την αντίστοιχη διαδικασία.

Και πάλι ύστερα από δοκιμές καταλήξαμε στο εξής διάγραμμα πόλων-μηδενικών:



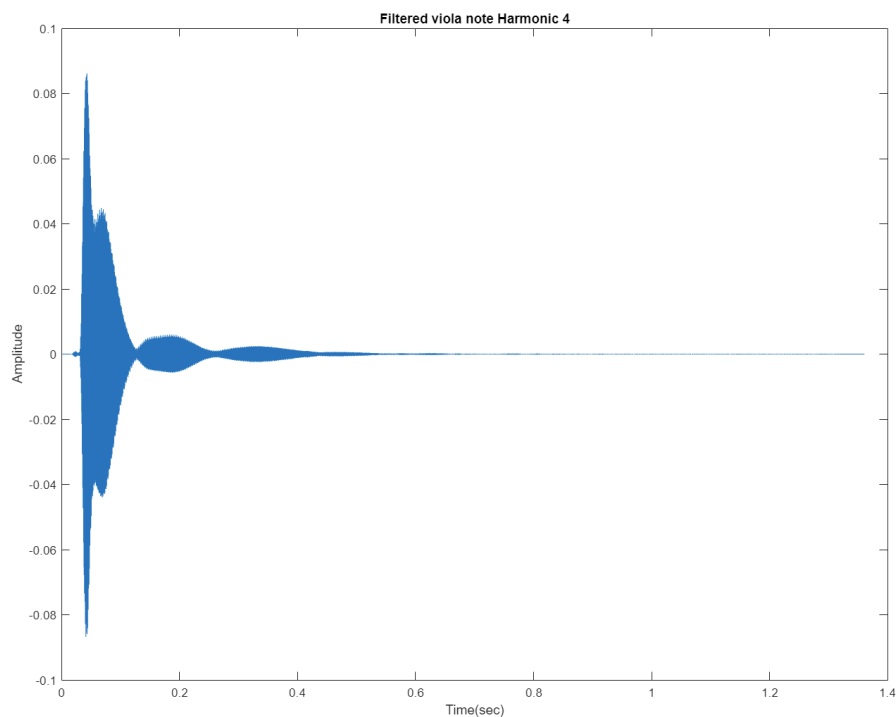
Οι πόλοι αυτή τη φορά είναι δύο ζεύγη συζυγών στα σημεία  $(0.9782, 0.184)$  και  $(0.9782, -0.184)$  ενώ τα μηδενικά είναι ένα ζεύγος συζυγών στα σημεία  $(0.85, 0)$ ,  $(-0.85, 0)$  και ένα ζεύγος στα σημεία  $(0, 0.85)$  και  $(0, -0.85)$ .

Η απόκριση πλάτους και φάσης είναι:



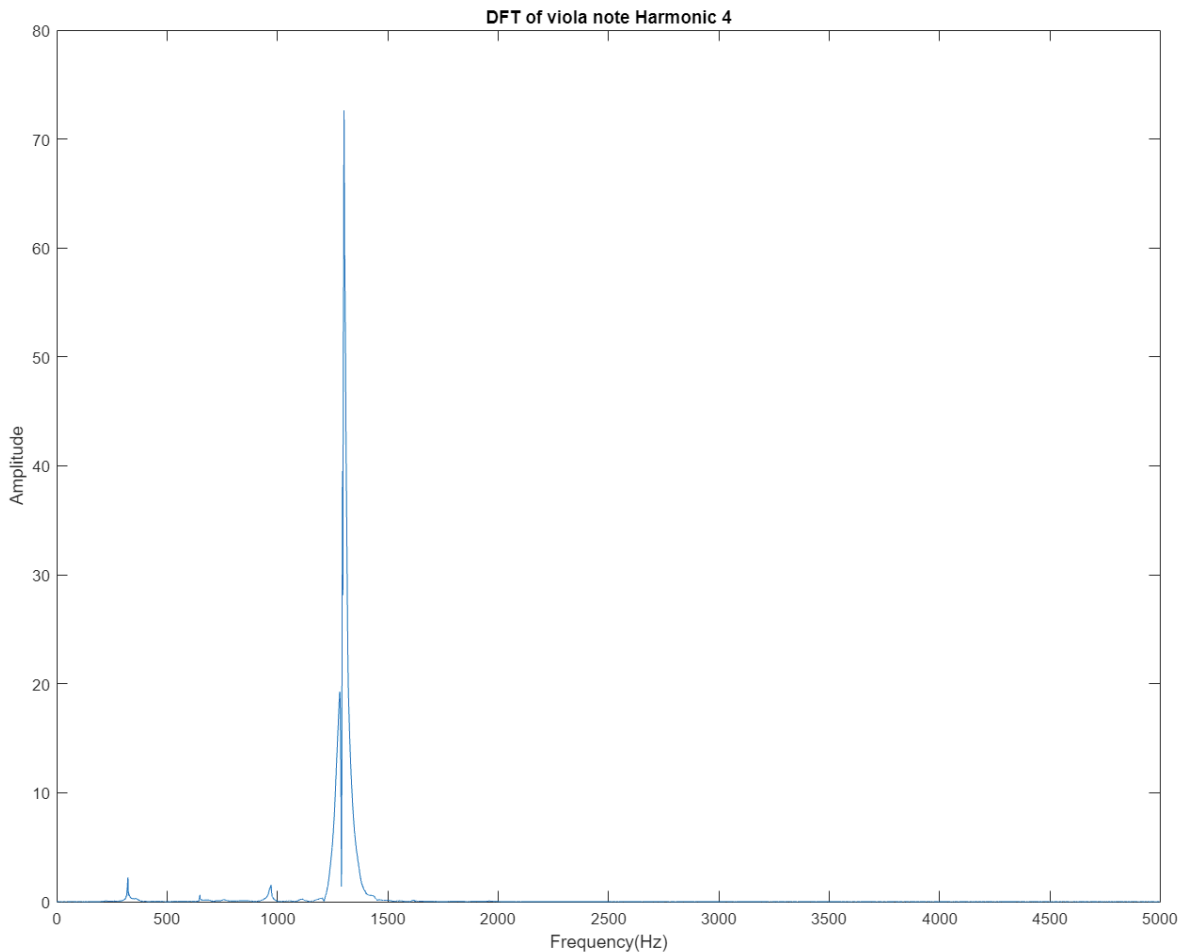
Για τους ίδιους λόγους που αναφέρθηκαν και στην υλοποίηση του προηγούμενου φίλτρου καταλήγουμε και πάλι ότι το φίλτρο για την αποκοπή της 4<sup>ης</sup> αρμονικής είναι οριακά ευσταθές.

Η έξοδος του αρχικού σήματος μετά την εφαρμογή του φίλτρου στο πεδίο του χρόνου είναι η παρακάτω:





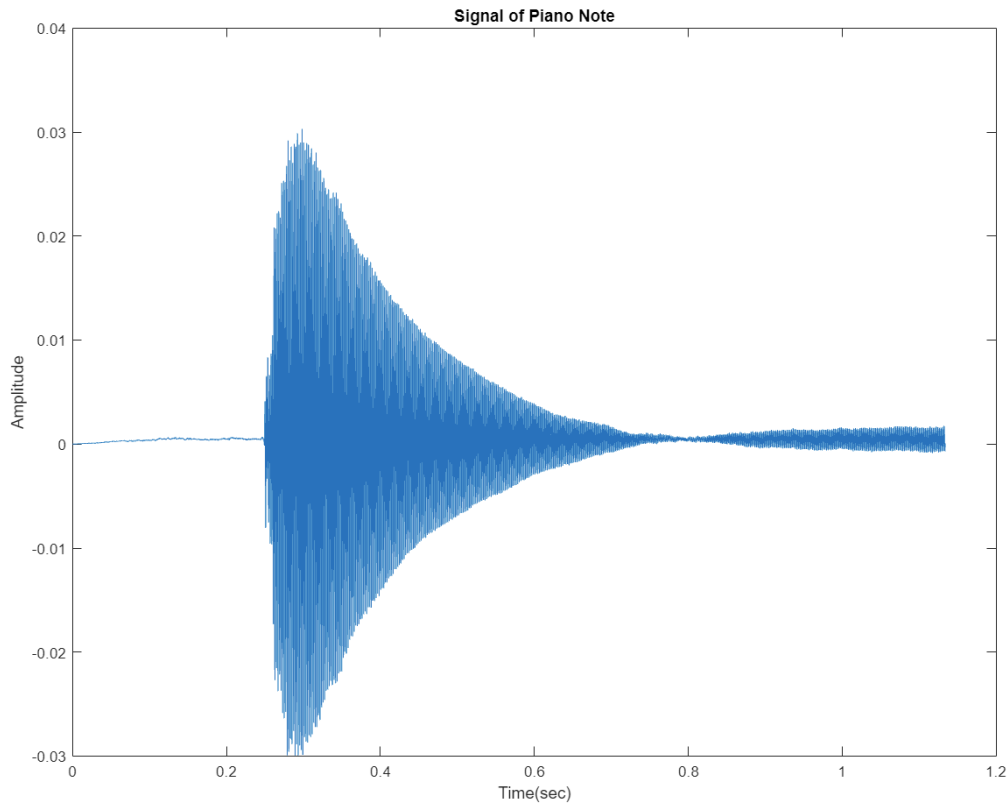
Το μέτρο του Διακριτού Μετασχηματισμού Fourier της παραπάνω εξόδου σχεδιάστηκε στο Matlab και είναι:



Η βασική συχνότητα η οποία περνά μετά την εφαρμογή του φίλτρου είναι τα 1301.68 Hz επομένως και πάλι επιτεύχθηκε να απομονωθεί μόνο η ζητούμενη αρμονική. Το μέγιστο πλάτος είναι 72.62 και αντιστοιχεί σε  $k = 0.0001$  (μπορούμε να το αλλάξουμε με τον ίδιο τρόπο όπως και στο προηγούμενο φίλτρο). Τέλος, παρατηρούμε πως το φίλτρο μας έχει αποκόψει τελείως κάθε άλλη συχνότητα εκτός της ζητούμενης και της κοντινής της περιοχής που αντιστοιχεί στη ζώνη διέλευσης του ζωνοπερατού φίλτρου. Επίσης, φαίνεται πως η συχνότητα της 1<sup>ης</sup> και 3<sup>ης</sup> αρμονικής δεν έχουν αποκοπεί τελείως και περνούν σε ελάχιστο βαθμό.

## 2.2 Εφαρμογή Φίλτρων για τη Δημιουργία Ηχούς και Αντήρησης εφέ σε Μουσικά Σήματα

**2.2.α)** Μέσω της εντολής `audioread()` φορτώνουμε το αρχείο *piano\_note.wav* το οποίο έχει συχνότητα δειγματοληψίας 44.1 kHz. Για τον σχεδιασμό του σήματος χρησιμοποιήθηκε η εντολή `plot()` και το αποτέλεσμα ήταν το παρακάτω:



Για να ακούσουμε τη νότα του πιάνο χρησιμοποιήθηκε η εντολή `sound()`.

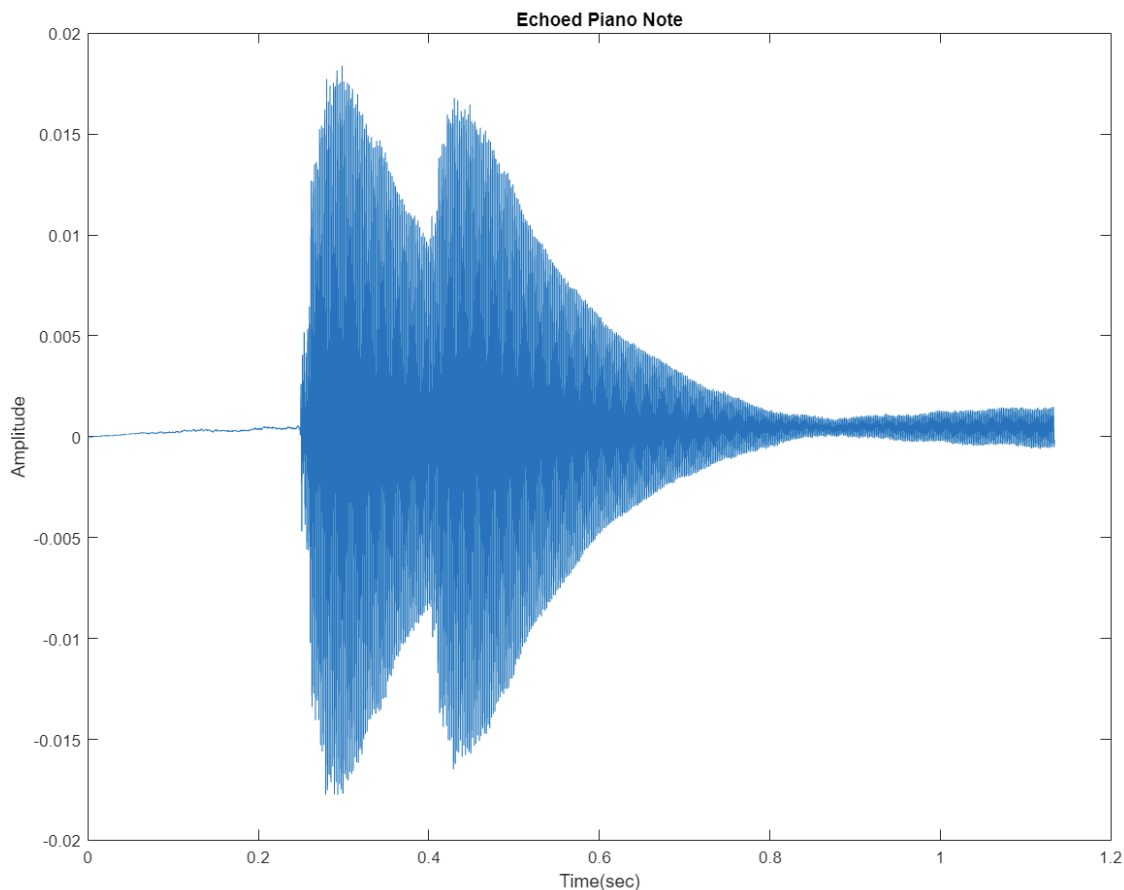
**2.2.β)** Θέλουμε η παράμετρος καθυστέρησης  $P$  του φίλτρου ηχούς να αντιστοιχεί σε καθυστέρηση 0.15 sec. Έχουμε ότι η απόσταση κάθε δείγματος από το επόμενο αντιστοιχεί σε  $\frac{1}{44.1 \times 10^3} = 2.2676 \times 10^{-5} \text{ sec}$  άρα για συνολική καθυστέρηση 0.15 sec έχουμε ότι:

$$P = 44.1 \times 10^3 \times 0.15 \Rightarrow \mathbf{P = 6615}$$

Από την εξίσωση διαφορών και την εξίσωση που περιγράφει το echo effect για παραμέτρους  $c = 0.6$  και  $P = 6615$ :  $y[n] = 0.6x[n] + 0.4x[n-6615]$ , προκύπτει ότι τα διανύσματα  $a$ ,  $b$  του φίλτρου ηχούς είναι:

- $a_0 = 1$  και  $a_n = 0$  για κάθε  $n > 0$
- $b_0 = 0.6$ ,  $b_{6615} = 0.4$  και  $b_n = 0$  για κάθε άλλο  $n$

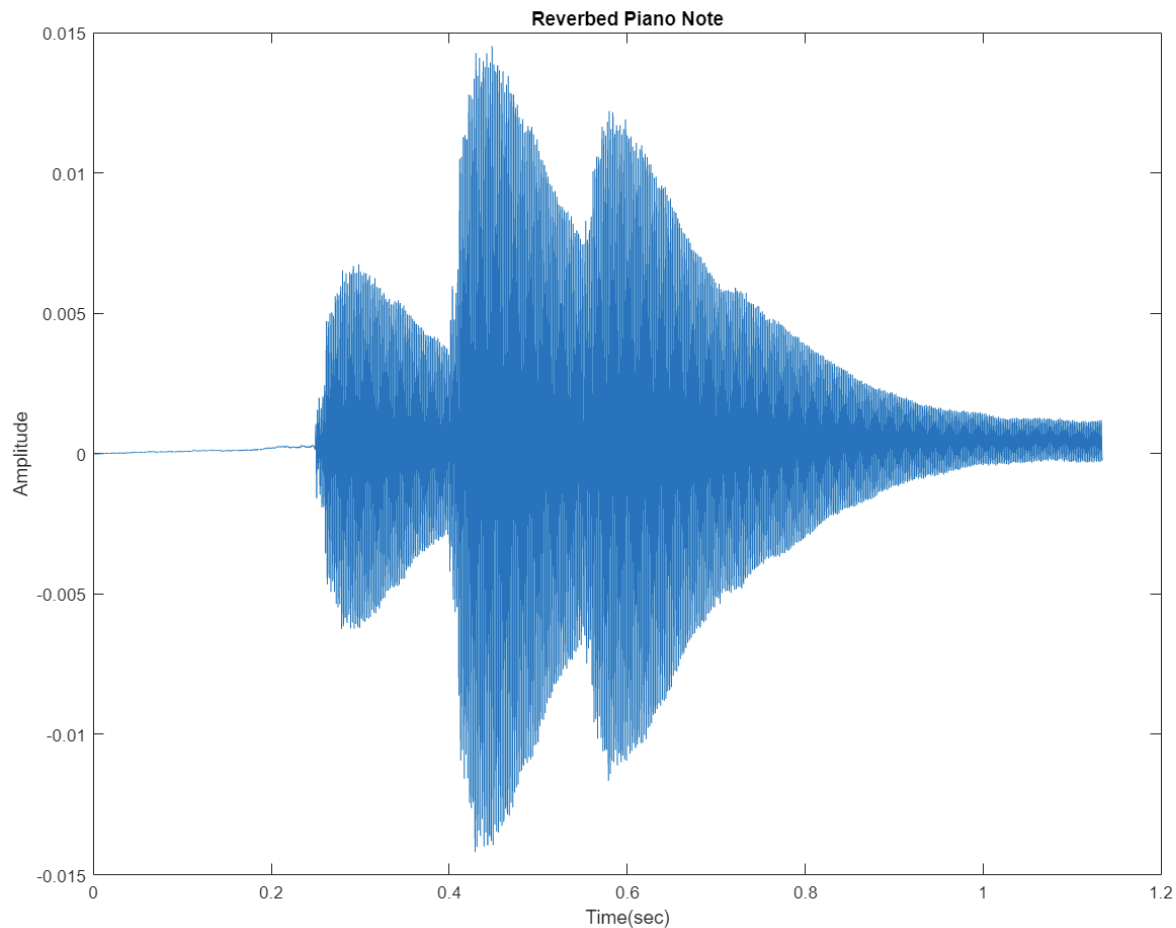
Για την εύρεση της κρουστικής απόκρισης του φίλτρου ηχούς χρησιμοποιήθηκε η συνάρτηση `impz()`, ενώ για τον υπολογισμό της εξόδου του φίλτρου χρησιμοποιήθηκε η `filter()`. Τέλος, για να ακουστεί το εφέ της ηχούς χρησιμοποιήθηκε η συνάρτηση `sound()` και σχεδιάστηκε με την `plot()`:



Παρατηρούμε από το παραπάνω γράφημα πως το echo effect είναι μια επικάλυψη του αρχικού σήματος με τον εαυτό του με καθυστέρηση 0.15 s όπως θα αναμέναμε και εμπειρικά.

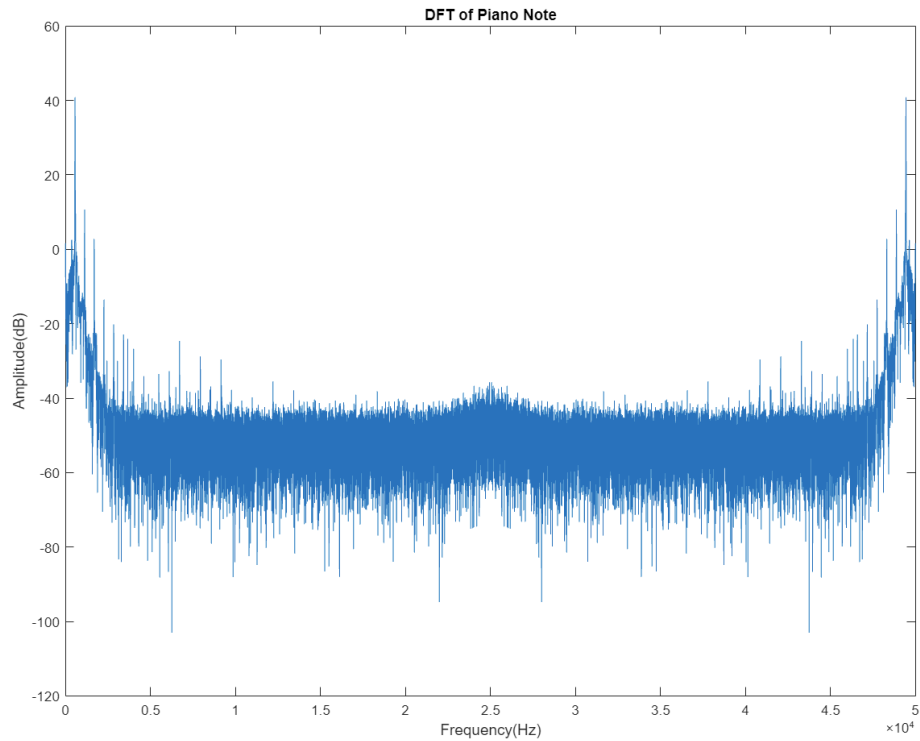
Για την υλοποίηση του φίλτρου αντήχησης θα χρησιμοποιήσουμε όπως είδαμε και στην πρώτη άσκηση 3 διαδοχικά φίλτρα ηχούς. Για να βρούμε δηλαδή την κρουστική απόκριση του φίλτρου αντήχησης θα κάνουμε δύο διαδοχικές συνελίξεις της κρουστικής απόκρισης του φίλτρου ηχούς με τον εαυτό της.

Από τη σχέση 1.1.4 υπολογίζουμε και πάλι όπως στο ερώτημα 1.1.ε το διάνυσμα  $a$  το οποίο είναι 1 για  $n = 0$  και 0 για κάθε άλλο  $n$ . Επίσης, από τη σχέση 1.1.3 προκύπτει ξανά ότι το διάνυσμα  $b$  ταυτίζεται με την κρουστική απόκριση του φίλτρου αντήχησης. Η έξοδος του φίλτρου βρίσκεται και πάλι μέσω της εντολής `filter()`, ακούγεται μέσω της `sound()` και σχεδιάζεται με την `plot()` όπως παρακάτω:

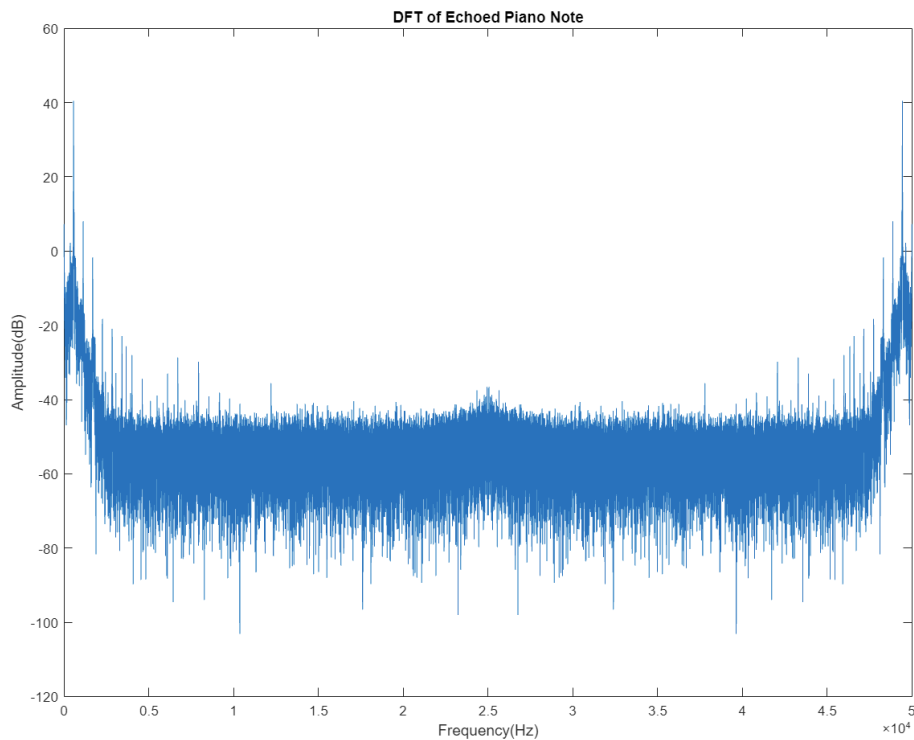


Παρατηρούμε από το γράφημα ότι το reverb effect αποτελεί μια τριπλή αλληλουχία του αρχικού σήματος αλλά η κάθε επανάληψη έχει διαφορετική ένταση. Αυτό επιβεβαιώνει και αυτό που ακούστηκε με την εντολή `sound()`.

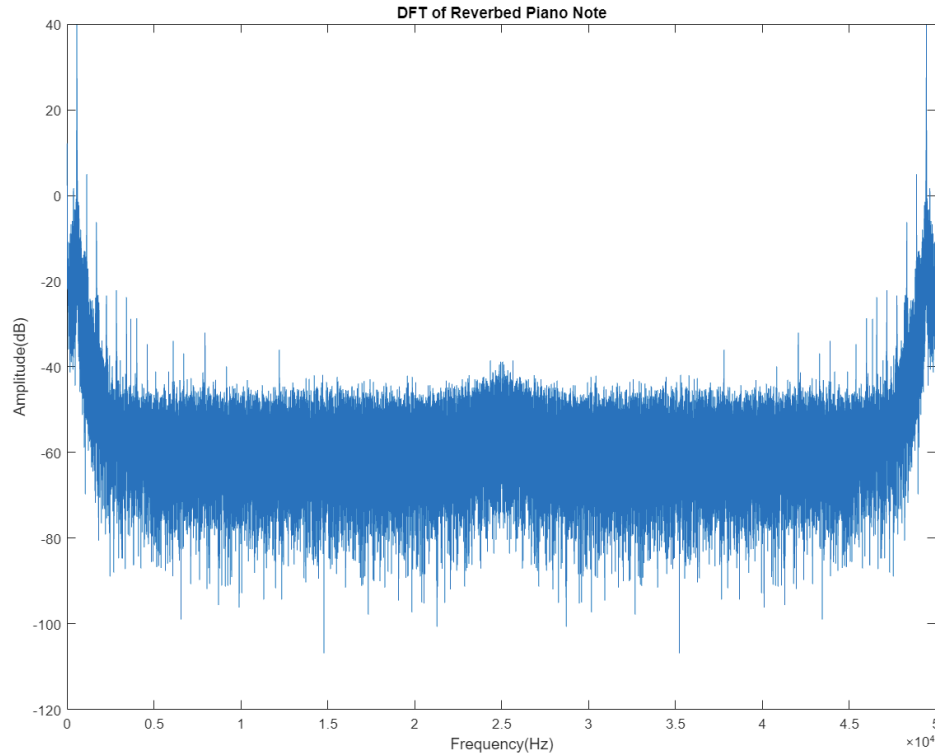
**2.2.γ)** Στο ερώτημα αυτό ζητείται το μέτρο του DFT μέσω της εντολής `fft()` σε λογαριθμική κλίμακα για κάθε ένα από τα 3 προηγούμενα σήματα. Έτσι για το αρχικό σήμα της νότας του πιάνο έχουμε το παρακάτω διάγραμμα:



Για το μέτρο φάσματος του σήματος ηχούς έχουμε:



Τέλος για το σήμα αντήχησης έχουμε το παρακάτω μέτρο φάσματος:



Παρατηρούμε αρχικά πως και τα 3 φάσματα είναι συμμετρικά όπως και αναμενόταν.

Επιπλέον, μπορούμε να δούμε ότι το μέτρο φάσματος του σήματος ηχούς μοιάζει με μια πιο πυκνή απεικόνιση του μέτρου φάσματος του αρχικού σήματος. Αυτό συμβαίνει γιατί το φάσμα του echoed σήματος είναι μια επικάλυψη δύο αρχικών σημάτων.

Για το μέτρο φάσματος του σήματος αντήχησης σε σύγκριση με αυτό του αρχικού σήματος προκύπτουν αντίστοιχες παρατηρήσεις με αυτές για το σήμα ηχούς με τη διαφορά πως στο σήμα αντήχησης έχουμε 3 επικαλύψεις.

**1.1.δ)** Για μικρότερα διαστήματα καθυστέρησης της ηχούς αρκεί να μειώσουμε την παράμετρο καθυστέρησης  $P$ . Για να εντοπίσουμε ένα σημείο στο οποίο σταματάει να γίνεται αντιληπτό το echo effect ακολουθήσαμε την εξής διαδικασία:

- Εντοπίζουμε το σημείο που αντιστοιχεί στη δεύτερη κορυφή του echo effect και είναι το (0.429909, 0.0167847).
- Εντοπίζουμε το αμέσως προηγούμενο σημείο που αντιστοιχεί περίπου σε αυτήν την τιμή και το οποίο προφανώς βρίσκεται δεξιά της πρώτης κορυφής. Το σημείο αυτό είναι το (0.316531, 0.016687).

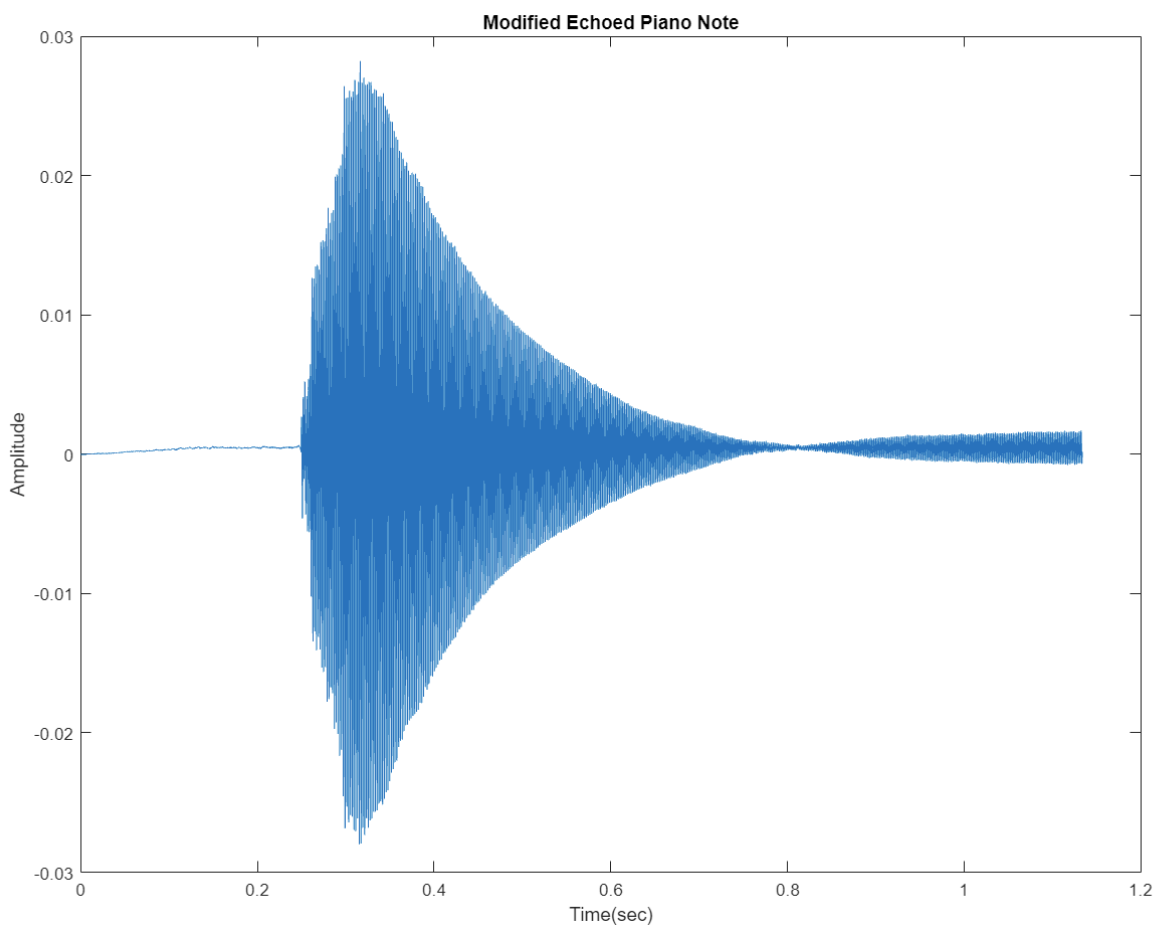
- Εντοπίζουμε το σημείο που βρίσκεται περίπου 0.15s αριστερά από το 1<sup>ο</sup> σημείο που βρήκαμε και είναι το σημείο (0.280204, 0.0164978).
- Υπολογίζουμε τη διαφορά  $0.31653 - 0.280204 = 0.036327 \text{ sec} = 36.327 \text{ ms}$ .

Ο παραπάνω χρόνος είναι ο χρόνος καθυστέρησης του echo effect ώστε να μην γίνεται αντιληπτό.

Το P που αντιστοιχεί στην καθυστέρηση αυτή είναι:  $0.036327 \times 44.1 \times 10^3 = 1602$

Άρα το echo effect σταματά να γίνεται αντιληπτό για  **$P \cong 1602$** .

Το τροποποιημένο σήμα ηχούς με παράμετρο καθυστέρησης  $P = 1602$  δίνεται παρακάτω:

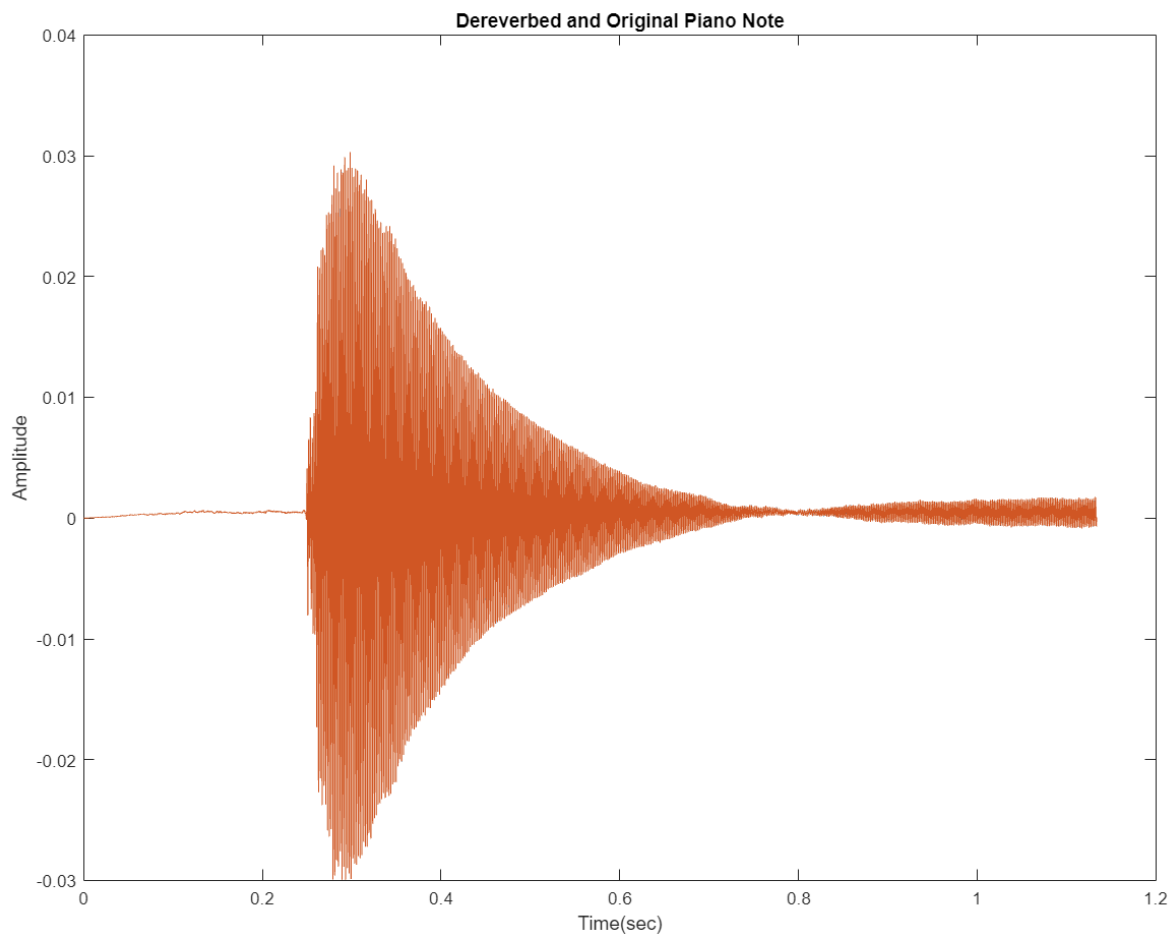


Φαίνεται και από το γράφημα ότι το echo effect δεν γίνεται αντιληπτό πλέον.

**2.2.ε)** Για την αποθήκευση των δύο φιλτραρισμένων σημάτων σε δύο αρχεία τύπου .wav χρησιμοποιήθηκε η εντολή audiowrite() καθώς δεν εντοπίστηκε η εντολή wavwrite() από την online έκδοση του Matlab. Τα δύο αρχεία ονομάστηκαν όπως καθοδηγήθηκε από την εκφώνηση και βρίσκονται στο αρχείο zip που στάλθηκε.

**2.2.στ)** Προκειμένου να δημιουργήσουμε ένα φίλτρο απαλοιφής της αντήχησης θα χρησιμοποιηθούν όσα είδαμε στο ερώτημα (1.1.στ). Αξιοποιώντας και πάλι τις σχέσεις 1.1.6 και 1.1.7 καταλήγουμε εύκολα στο συμπέρασμα ότι το διάνυσμα  $a$  που αντιστοιχεί στο φίλτρο απαλοιφής της αντήχησης ταυτίζεται με το διάνυσμα  $b$  που αντιστοιχεί στο φίλτρο αντήχησης αλλά και το αντίστροφο, δηλαδή το διάνυσμα  $b$  του dereverberation φίλτρου ταυτίζεται με το  $a$  του reverberation.

Έτσι μέσω της συνάρτησης filter() υπολογίζεται η έξοδος του φίλτρου απαλοιφής της αντήχησης με είσοδο την έξοδο του φίλτρου αντήχησης και σχεδιάστηκε στο ίδιο διάγραμμα με το αρχικό σήμα της νότας του πιάνο όπως παρουσιάζεται:



Είναι προφανές πως το αρχικό σήμα με το dereverbed ταυτίζονται πλήρως.



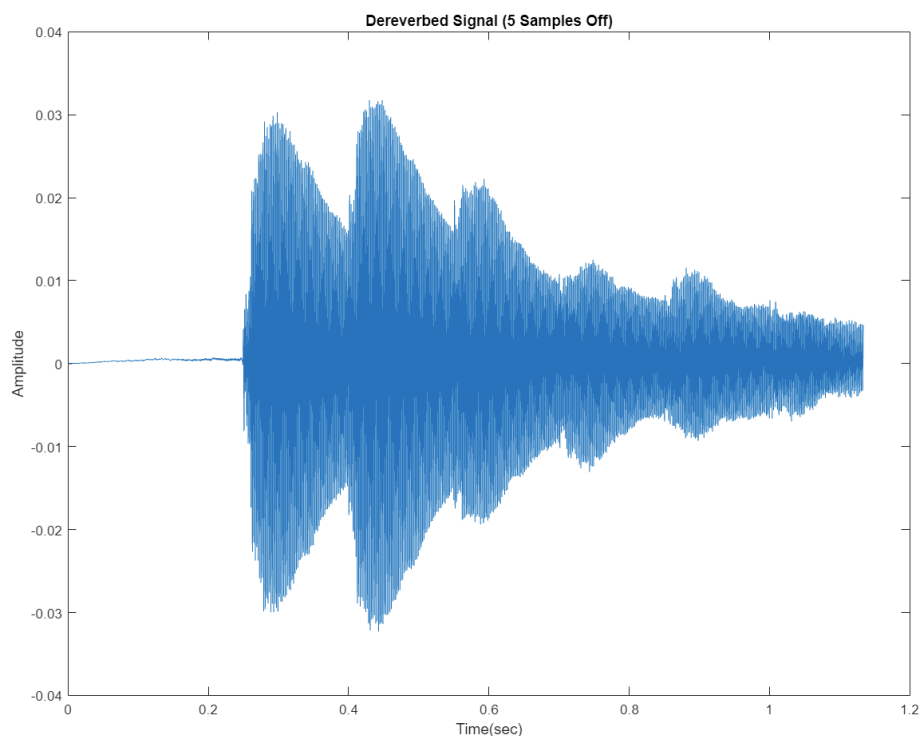
**2.2.ζ)** Για το ερώτημα αυτό θα παρουσιαστεί αναλυτικά η περίπτωση στην οποία η παράμετρος καθυστέρησης  $P$  του φίλτρου έχει τροποποιηθεί ώστε να αντιστοιχεί σε καθυστέρηση 5 επιπλέον δειγμάτων από την καθυστέρηση του ερωτήματος ( $\beta$ ) και στη συνέχεια θα παρατεθούν και τα αποτελέσματα για τις άλλες δύο περιπτώσεις χωρίς την ίδια αναλυτικότητα.

Για καθυστέρηση κατά 5 δείγματα μεγαλύτερη από αυτή του ερωτήματος ( $\beta$ ) το νέο  $P$  θα είναι ίσο με 6620. Το διάνυσμα  $a$  θα παραμείνει 1 μόνο για  $n = 0$  και 0 για κάθε άλλο  $n$  ενώ το διάνυσμα  $b$  θα είναι σε αυτήν την περίπτωση ίσο με:  $b_0 = 0.6$ ,  $b_{6620} = b_P = 0.4$  και 0 για κάθε άλλο  $n$ .

Μέσω της συνάρτησης `impz()` βρίσκουμε την κρουστική απόκριση του φίλτρου ηχούς και στη συνέχεια μετά από 2 συνελίξεις της παραπάνω κρουστικής απόκρισης με τον εαυτό της εντοπίζουμε την αντίστοιχη κρουστική του φίλτρου αντήχησης.

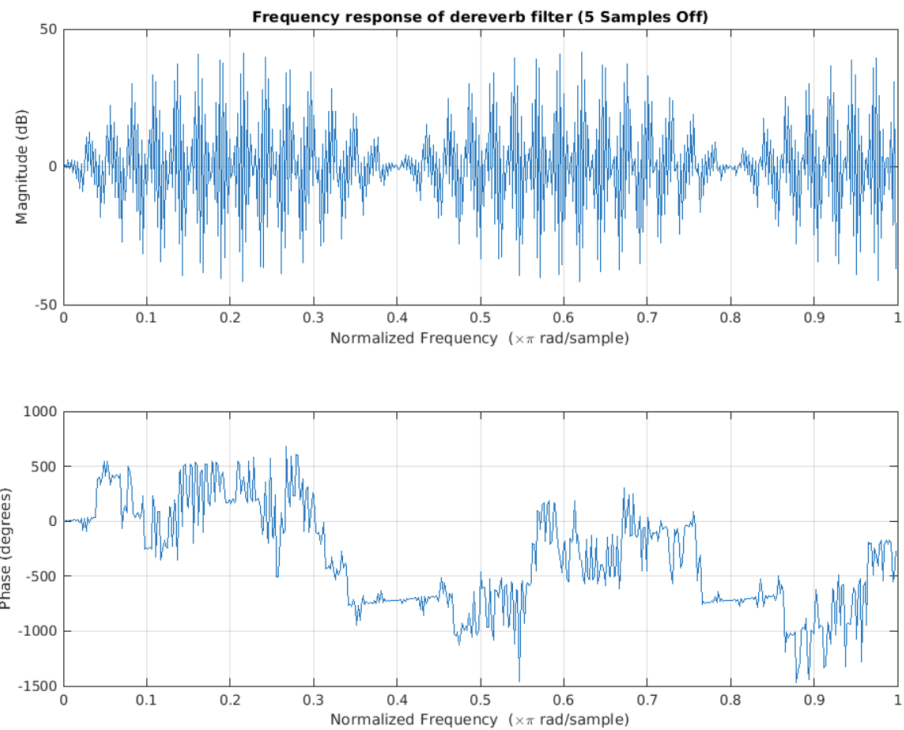
Έπειτα όπως είδαμε και στο ερώτημα 2.2.στ τα διανύσματα  $a$ ,  $b$  του `reverb` φίλτρου προκύπτουν από τις σχέσεις 1.1.3 και 1.1.4 και είναι  $a_{\text{reverb}} = 1$  και  $b_{\text{reverb}} = h_{\text{reverb}}$ , ενώ τα αντίστοιχα διανύσματα για το `dereverb` φίλτρο προκύπτουν από τις σχέσεις:  $a_{\text{dereverb}} = b_{\text{reverb}}$  και  $b_{\text{dereverb}} = a_{\text{reverb}}$ .

Στη συνέχεια, χρησιμοποιώντας την συνάρτηση `filter()` υπολογίζουμε την έξοδο του φιλτραρισμένου σήματος από το φίλτρο απαλοιφής της αντήχησης και πλοτάρουμε το αποτέλεσμα:

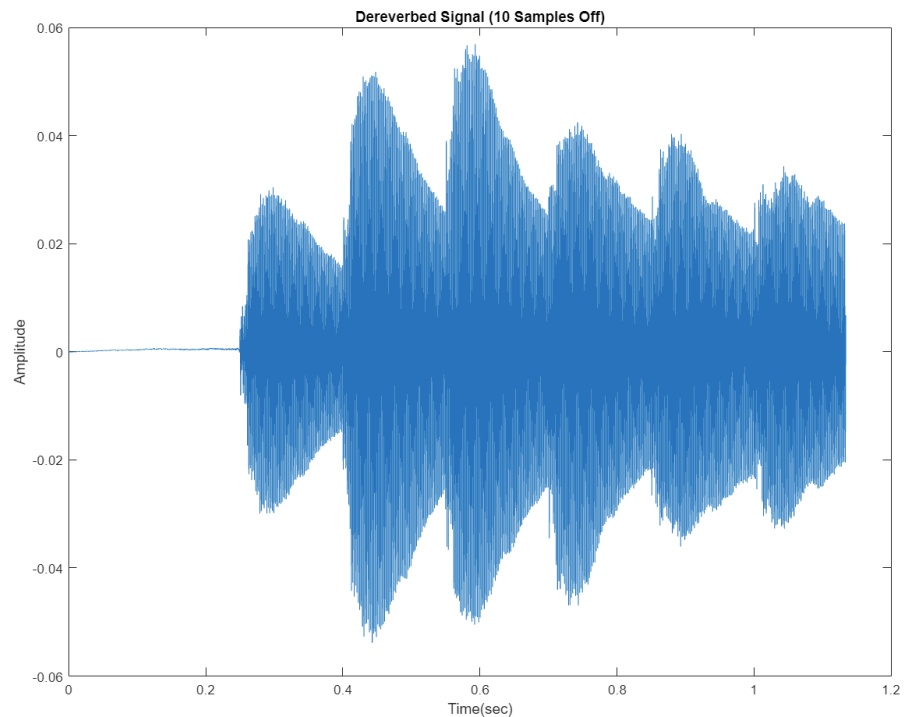


Ακολούθως, για να βρούμε την απόκριση πλάτους του συνολικού συστήματος αντήχησης και απαλοιφής υπολογίζουμε τα διανύσματα που αντιστοιχούν στο σήμα αυτό. Αυτό θα το πραγματοποιήσουμε μέσω της σχέσης 1.1.7 από όπου μπορούμε να καταλήξουμε πως το γινόμενο

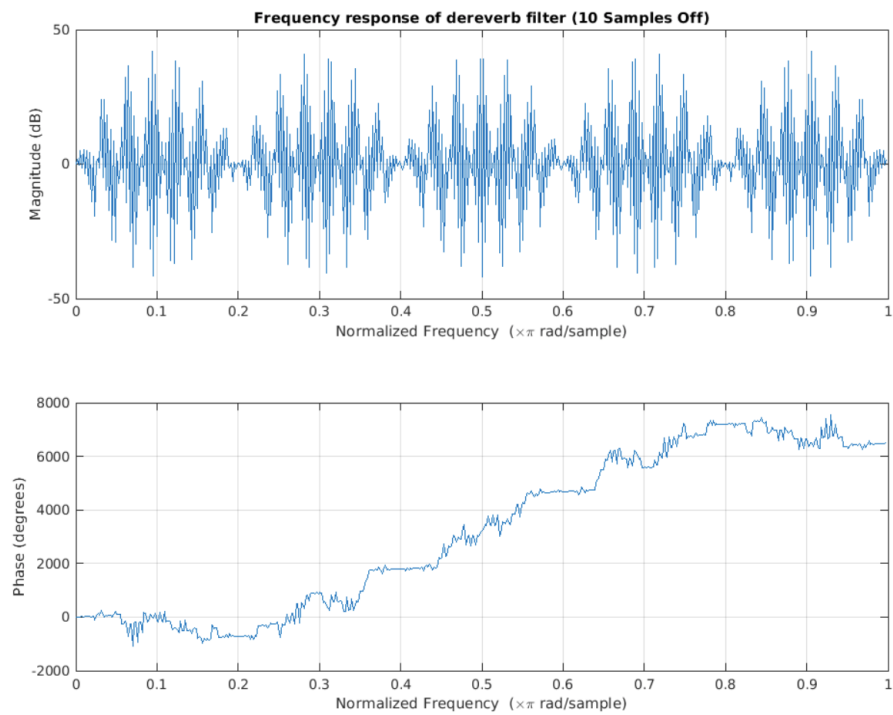
των δύο μετασχηματισμών  $Z$  αντιστοιχούν στον γινόμενο των διανυσμάτων  $a$ ,  $b$  των αντίστοιχων φίλτρων. Έτσι, η απόκριση πλάτους και φάσης που σχεδιάζεται με την εντολή `freqz()` είναι η εξής:



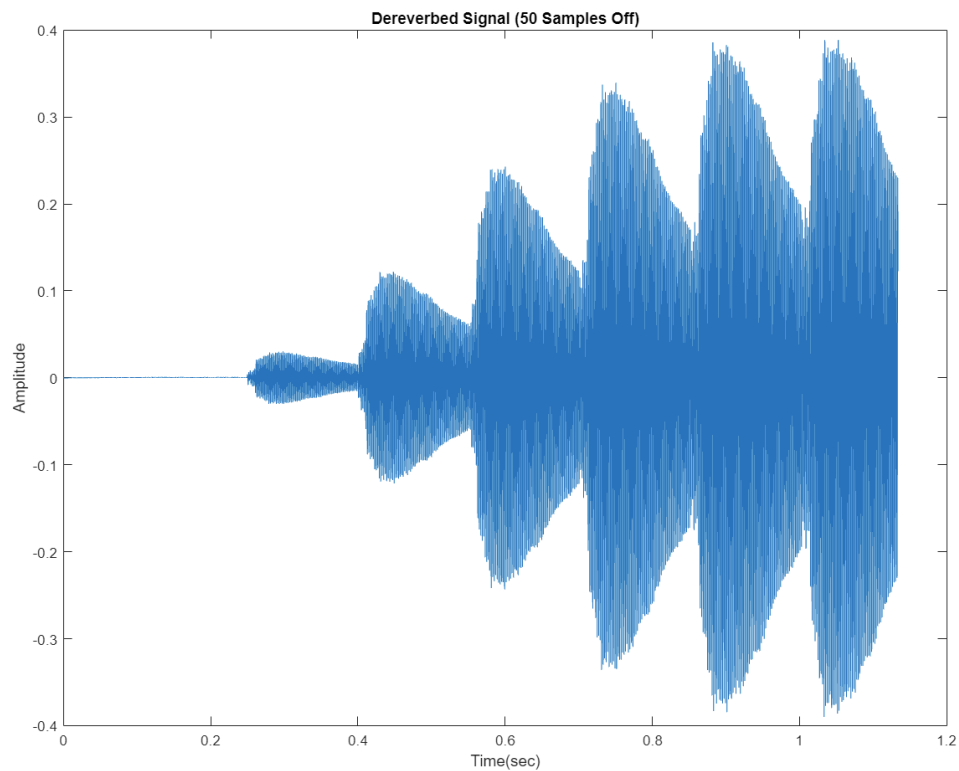
Για την καθυστέρηση που αντιστοιχεί στα 10 περισσότερα δείγματα ( $P = 6625$ ) έχουμε τα εξής διαγράμματα: Η έξοδος του dereverb φίλτρου είναι:

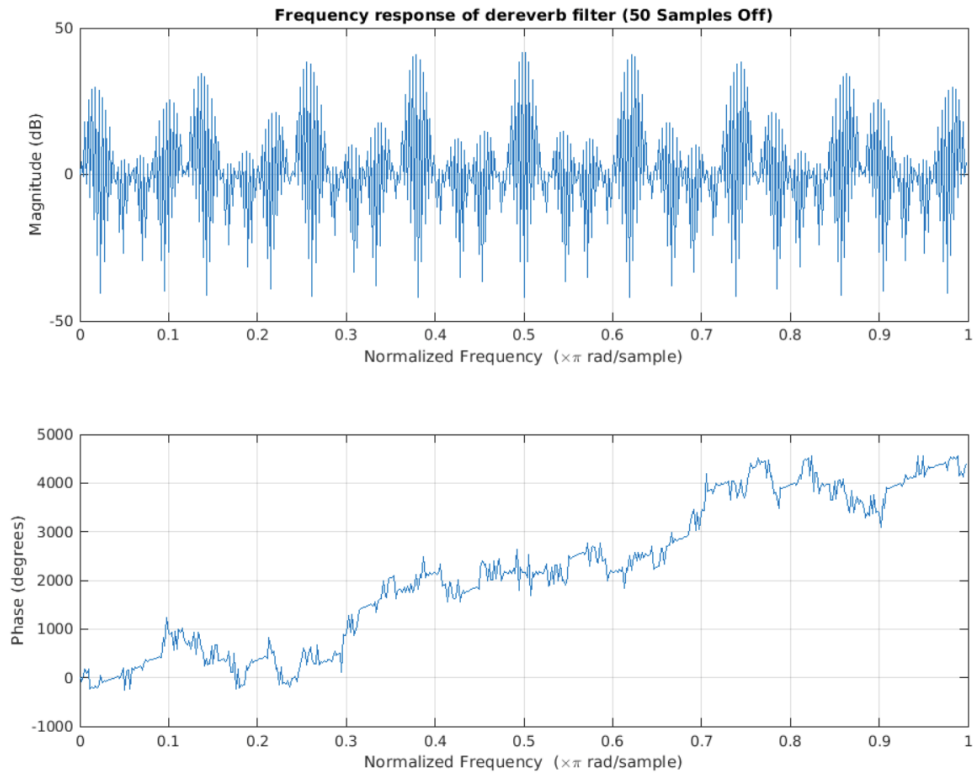


Η απόκριση πλάτους και φάσης είναι:

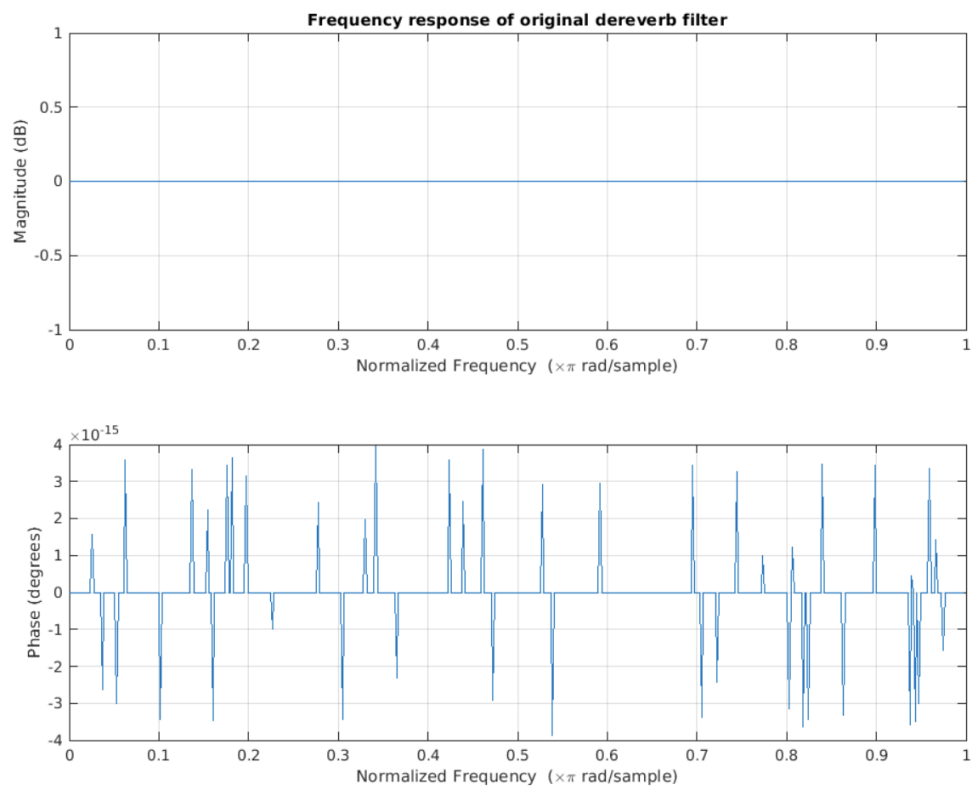


Τέλος, για τα 50 επιπλέον δείγματα το P είναι ίσο με 6665 και τα αντίστοιχα γραφήματα εξόδου του φίλτρου και της απόκρισης πλάτους και φάσης είναι:





Για να συγκρίνουμε τα παραπάνω αποτελέσματα σχεδιάστηκε και η απόκριση πλάτους-φάσης της αρχικής περίπτωσης του dereverb φίλτρου με  $P = 6615$  και είναι:



Παρατηρούμε ότι η παραπάνω απόκριση πλάτους και φάσης είναι μηδέν κάτι το οποίο αναμενόταν λόγω της ακριβούς επικάλυψης του αρχικού σήματος με το σήμα εξόδου του φίλτρου απαλοιφής όπως είδαμε και στο ερώτημα 2.2.στ. Αυτό ωστόσο δεν συμβαίνει και στις περιπτώσεις με την τροποποιημένη καθυστέρηση διότι το φιλτραρισμένο σήμα δεν ταυτίζεται με το αρχικό όπως βλέπουμε και από τα προηγούμενα γραφήματα. Τέλος, από τις αποκρίσεις στα τροποποιημένα σήματα γίνεται προφανές ότι όσο περισσότερα δείγματα έχουμε, δηλαδή όσο μεγαλύτερη καθυστέρηση, τόσο αυξάνεται και η συχνότητα της απόκρισης πλάτους.