Παράρτημα Β – Signal Processing Toolbox

Τι είναι το Signal Processing Toolbox ;

Το Signal Processing Toolbox είναι μια συλλογή από εργαλεία φτιαγμένα για την υποστήριξη του περιβάλλοντος αριθμητικών υπολογισμών Matlab. Το SPT υποστηρίζει ένα ευρύ πεδίο λειτουργιών στην επεξεργασία σήματος, όπως την γέννηση κυματομορφών, το σχεδιασμό φίλτρων, την παραμετρική μοντελοποίηση, και τέλος την φασματική ανάλυση. Το SPT παρέχει δυο κατηγορίες εργαλείων.

- Τις συναρτήσεις επεξεργασίας σήματος
- Αλληλεπιδραστικά γραφικά εργαλεία.

Η πρώτη κατηγορία είναι αυτή των συναρτήσεων, όπου μπορούν να καλεστούν είτε από την γραμμή εντολών είτε από την εφαρμογή που αναπτύσσουμε. Αυτές οι συναρτήσεις έχουν ομαδοποιηθεί ανά κατηγορία και έχουν ενσωματωθεί σε αρχεία (με κατάληξη του αρχείου *.m) και για αυτό τον λόγο λέγονται M-files. Σε αυτά τα αρχεία υπάρχουν δηλώσεις MATLAB όπου τελικά εφαρμόζονται ειδικοί αλγόριθμοι επεξεργασίας σήματος. Μπορούμε να δούμε το περιεχόμενο ενός M-file χρησιμοποιώντας την παρακάτω δήλωση.

Type function_name

Μπορούμε ακόμα να τροποποιήσουμε τα περιεχόμενα ενός τέτοιου αρχείου ή ακόμα να φτιάξουμε ένα δικό μας M-file.Η δεύτερη κατηγορία εργαλείων παρέχει ένα σύνολο από αλληλεπιδραστικά εργαλεία όπου ο έλεγχος όλων των λειτουργιών γίνεται διαμέσου ενός γραφικού περιβάλλοντος διασύνδεσης προγράμματος - χρήστη GUI (Graphical User Interface). Τα εργαλεία που βασίζονται στην παραπάνω τρόπο λειτουργίας παρέχουν τελικώς ένα ολοκληρωμένο περιβάλλον για τον σχεδιασμό φίλτρων, ανάλυση αυτών, επεξεργασία , και τέλος δυνατότητα μελέτης σημάτων.

Σε αυτό το εγχειρίδιο θα χρησιμοποιήσουμε την παρακάτω τεχνική σημειογραφία:

Nyquist frequency	Είναι το μισό της συχνότητας δειγματοληψίας. Οι περισσότερες συναρτήσεις κανονικοποιούν την τιμή αυτή στο 1.
x (1)	Το πρώτο στοιχείο μιας αλληλουχίας δεδομένων ή φίλτρου.
Ω	Αναλογική συχνότητα σε ακτίνια ανά δευτερόλεπτο
W	Ψηφιακή συχνότητα σε ακτίνια ανά δευτερόλεπτο.
f	Ψηφιακή συχνότητα

B.1 Εισαγωγή στην Επεξεργασία Σήματος

B.1.1 Ανάλυση και Υλοποίηση Φίλτρων

Σε αυτή την ενότητα θα δούμε πως μπορούμε να φιλτράρουμε ένα διακριτό σήμα χρησιμοποιώντας τις συναρτήσεις του Matlab (όπως είναι η συνάρτηση filter) . Ακόμα θα δούμε πως μπορούμε χρησιμοποιώντας συναρτήσεις του SPT για να αναλύσουμε τα χαρακτηριστικά ενός φίλτρου, όπως η κρουστική απόκριση, η απόκριση πλάτους και φάσης, η καθυστέρηση, και η θέση των πόλων και των μηδενικών.

Συνέλιξη και Φιλτράρισμα

Η μαθηματική θεμελίωση του φιλτραρίσματος είναι η συνέλιξη. Η συνάρτηση conv πραγματοποιεί μιας διάστασης συνέλιξη, μεταξύ δύο διανυσμάτων, όπως στο παρακάτω παράδειγμα:

```
conv ([1 1 1] , [1 1 1]

ans =
      1   2   3   2   1
```

Σημείωση: Χρησιμοποιώντας την συνάρτηση conv2 μπορούμε να πάρουμε το συνελικτικό γινόμενο δισδιάστατων σημάτων.

Ένα ψηφιακό φίλτρο με έξοδο y(n) σχετίζεται με την είσοδο x(n) δια μέσου της συνέλιξης της κρουστικής του απόκρισης h(n).

$$y(n) = h(n) * x(n) = \sum_{m=-\infty}^{\infty} h(n-m)x(m)$$

Εάν σε ένα ψηφιακό φίλτρο η κρουστική απόκριση h(n) και η είσοδος x(n) είναι πεπερασμένου μήκους, μπορούμε να πραγματοποιήσουμε ένα φίλτρο χρησιμοποιώντας την συνάρτηση conv. Αποθηκεύουμε την x(n) σε ένα διάνυσμα, αντίστοιχα την h(n) σε άλλο διάνυσμα και εκτελούμε συνέλιξη μεταξύ των δύο διανυσμάτων.

```
x = randn(5,1);    % Τυχαίο διάνυσμα μήκους 5
h = [1 1 1 1 ]/4;  % Φίλτρο μέσου όρου μήκους 4
y = conv(h,x);
```

Φίλτρα και Συναρτήσεις Μεταφοράς

Ο μετασχηματισμός z ,Y(z) της εξόδου y(n) ενός ψηφιακού φίλτρου, σχετίζεται άμεσα με τον μετασχηματισμό z, X(z) της εισόδου x(n) με την ακόλουθη σχέση:

$$Y(z) = H(z)X(z) = \frac{b(1) + b(2)z^{-1} + ... + b(nb+1)z^{-nb}}{a(1) + a(2)z^{-1} + ... + a(na+1)z^{-na}} X(z)$$

Όπου το H(z) είναι η κρουστική απόκριση του φίλτρου. Οι σταθερές b(i) και a(i) είναι οι συντελεστές του φίλτρου, η τάξη του φίλτρου προσδιορίζεται από τον μέγιστο αριθμό των συντελεστών na.

Σημείωση: Οι δείκτες των συντελεστών του φίλτρου ξεκινάνε από το 1. Αυτό δείχνει τον τρόπο που το Matlab καταχωρεί τους δείκτες των διανυσμάτων.

Το Matlab αποθηκεύει τους συντελεστές του φίλτρου σε δύο διανύσματα, το ένα για τον αριθμητή και το άλλο για τον παρανομαστή. Κατά σύμβαση στο Matlab χρησιμοποιούμε διανύσματα γραμμής για τους συντελεστές των φίλτρων.

Συντελεστές φίλτρων και ονόματα φίλτρων.

- Όταν nb = 0 , το φίλτρο είναι IIR (Infinity Impulse Response), μόνο πόλους , αναδρομικό , ή autoregressive (AR) φίλτρο.
- Όταν na = 0 . το φίλτρο είναι FIR (Finite Impulse Response), μόνο μηδενικά, μη αναδρομικό , ή moving average (MA) φίλτρο.
- Εάν το na και το nb είναι μεγαλύτερα από το μηδέν , το φίλτρο είναι IIR , πόλους – μηδενικά, αναδρομικό, ή autoregressive moving average (ARMA) φίλτρο.

Οι συντομεύσεις AR, MA, και ARMA χρησιμοποιούνται συνήθως σε φίλτρα που σχετίζονται με διαδικασίες στοχαστικού φιλτραρίσματος.

Φιλτράροντας Χρησιμοποιώντας την Συνάρτηση filter

Από την προηγούμενη συνάρτηση μεταφοράς μπορούμε να πάρουμε την εξίσωση διαφορών, εάν δεχθούμε ότι το a(1)= 1. Μετακινούμε τον παρανομαστή στην αριστερή πλευρά και εκτελώντας αντίστροφο μετασχηματισμό z, θα έχουμε:

y(n)+a(2)y(n-1)+…+a(na+1)y(n-na)=b(1)x(n)+b(2)x(n-1)+…+b(nb+1)x(n-nb)

Η y(n) θα είναι y(n)= b(1)x(n)+b(2)x(n-1)+…+b(nb+1)x(n-nb)-a(2)y(n-1)-…-a(na+1)y(n-na)

Η παραπάνω απεικόνιση αναπαριστά ένα ψηφιακό φίλτρο στο πεδίο του χρόνου, το οποίο υπολογίζουμε ξεκινώντας για n=1 και θεωρώντας μηδενικές τις αρχικές συνθήκες. Επομένως θα έχουμε:

y(1)= b(1)x(1)
y(2)= b(1)x(2)+b(2)x(1)-a(2)y(1)
y(3)= b(1)x(3)+b(2)x(2)+b(3)x(1)-a(2)y(2)-a(3)y(3)
.

Ένα φίλτρο αυτής της μορφής μπορεί πολύ εύκολα να πραγματοποιηθεί με την συνάρτηση filter. Στο παρακάτω παράδειγμα έχουμε ένα φίλτρο διέλευσης χαμηλών, με ένα πόλο.

a =1; *% Αριθμητής*
b = [1 –0.9]; *% Παρανομαστής*

Τα a και b είναι διανύσματα που αντιπροσωπεύουν τους συντελεστές του φίλτρου σε μια συνάρτηση μεταφοράς. Για να εφαρμόσουμε τα παραπάνω δεδομένα στην συνάρτηση filter θα έχουμε:

y = filter(b,a,x);

Ο αριθμός των δειγμάτων εξόδου του y θα είναι ίδιος με τον αριθμό των δειγμάτων εισόδου του x , δηλαδή το μήκος τους θα είναι το ίδιο. Εάν το πρώτο στοιχείο του a δεν είναι 1, τότε η συνάρτηση filter διαιρεί τους συντελεστές με 1 πριν εφαρμόσει την εξίσωση διαφορών.

Β.1.2 Υλοποίηση Συναρτήσεων Φίλτρων και Αρχικές Συνθήκες Αυτών

Για το δείγμα m , η συνάρτηση filter υπολογίζει τις παρακάτω εξισώσεις διαφορών.

y(m) = b(1)x(m)+z1(m-1)
z1(m) = b(2)x(m)+z2(m-1)-a(2)y(m)
•
•
•
zn-2(m) = b(n-1)x(m)+zn-1(m-1)-a(n-1)y(m)
zn-1(m) = b(n)x(m)-a(n)y(m)

Στην βασική σύνταξη της εντολής η filter αρχικοποιεί τις καθυστερημένες εξόδους zi(1), i= 1,…,n-1 στο μηδέν. Αυτό σημαίνει ότι οι προηγούμενες είσοδοι και έξοδοι είναι μηδέν. Μπορούμε με την συνάρτηση filter να διαφοροποιήσουμε την αρχική και τελική καθυστέρηση, για την αρχική καθυστέρηση τοποθετούμε την τιμή στην είσοδο της συνάρτησης όπου είναι και η τέταρτη παράμετρος (zi). Για την τελική καθυστέρηση η παράμετρος είναι η (zf) και είναι η δεύτερη παράμετρος εξόδου:

[y,zf] = filter(b,a,x,zi)

Αλλαγή των αρχικών και τελικών καταστάσεων είναι πολύ χρήσιμη όταν φιλτράρουμε τμήματα δεδομένων και ειδικά όταν έχουμε περιορισμό στην μνήμη. Στο παρακάτω παράδειγμα υποθέτουμε ότι συλλέγουμε δεδομένα σε δύο τμήματα των 5000 σημείων το καθένα:

```
x1 = randn(5000,1);    %  Το πρώτο τμήμα δεδομένων
x2 = randn(5000,1);    %  Το δεύτερο τμήμα δεδομένων
```

Υποθέτουμε ότι η πρώτη ακολουθία ,x1, αντιστοιχεί στην συλλογή των δεδομένων τα πρώτα δέκα λεπτά και αντίστοιχα η ακολουθία ,x2, για τα επόμενα 10 λεπτά. Η ολική ακολουθία είναι x = [x1; x2]. Εάν δεν έχουμε αρκετή μνήμη για να επεξεργαστούμε την συνολική ακολουθία (x), μπορούμε τότε να φιλτράρουμε τις ακολουθίες μια κάθε φορά. Για να είμαστε σίγουροι ότι υπάρχει συνέχεια μεταξύ των ακολουθιών του φίλτρου, χρησιμοποιούμε την τελική συνθήκη του πρώτου φίλτρου x1 σαν αρχική για το δεύτερο φίλτρο x2:

```
[y1,zf] = filter(b,a,x1);
y2 =filter(b,a,x2,zf);
```

Η συνάρτηση filtic δημιουργεί αρχικές συνθήκες για την συνάρτηση filter. Η συνάρτηση filtic υπολογίζει ένα διάνυσμα καθυστέρησης και κάνει να συμπεριφέρεται το φίλτρο λαμβάνοντας υπόψη προηγούμενες εξόδους και εισόδους όπου είναι προσδιοριζόμενες από εμάς. Μπορούμε χρησιμοποιώντας την συνάρτηση filtic να πετύχουμε την ίδια τιμή καθυστέρησης εξόδου (zf) όπως στο παραπάνω παράδειγμα:

```
zf = filtic(b,a,flipud(y1),flipud(x1));
```

Αυτό είναι χρήσιμο όταν φιλτράρουμε μικρές ακολουθίες, ή μια προσδιοριζόμενη αρχική συνθήκη μας βοηθά να αποφύγουμε τα αρχικά μεταβατικά φαινόμενα.

B.1.3 Άλλες Συναρτήσεις για Φιλτράρισμα

Εκτός από τη συνάρτηση filter που είδαμε προηγουμένως υπάρχουν και άλλες συναρτήσεις στο SPT που μπορούμε να χρησιμοποιήσουμε για να πραγματοποιήσουμε βασικές λειτουργίες φιλτραρίσματος. Αυτές οι συναρτήσεις είναι η upfirdn, όπου πραγματοποιεί φιλτράρισμα FIR με αναδειγματοληψία, η filtfilt, όπου εξαφανίζει την φασική παραμόρφωση κατά την διαδικασία φιλτραρίσματος, και η fftfilt, όπου πραγματοποιεί φιλτράρισμα FIR στο πεδίο του χρόνου.

Υλοποίηση Φίλτρων Μηδενικής Φάσεως

Στην περίπτωση φίλτρων FIR, μπορούμε να σχεδιάσουμε φίλτρα γραμμικής φάσης, χρησιμοποιώντας τις συναρτήσεις filter ή conv. Σε αυτές τις περιπτώσεις έχουμε μια σταθερή καθυστέρηση της εξόδου. Για φίλτρα IIR η παραμόρφωση φάσης είναι συνήθως μη γραμμική. Η συνάρτηση filtfilt παίρνει πληροφορία από το σήμα, πριν και μετά από το τρέχον σημείο, με σκοπό να εξαφανίσει την φασική παραμόρφωση.

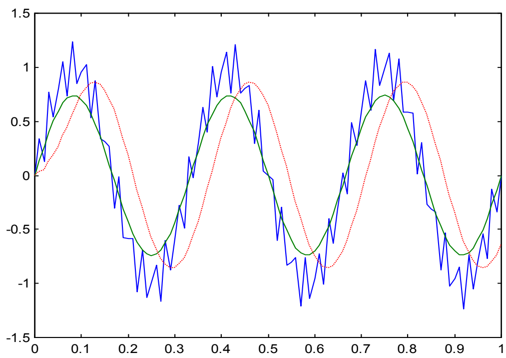
Όταν $|z| = 1$ και αυτό συμβαίνει όταν $z = e^{jw}$, η έξοδος μειώνεται στο $X(e^{jw})\left|H(e^{jw})\right|^2$. Για όλα τα δείγματα εισόδου της ακολουθίας x(n), είναι δυνατή μια διπλή φιλτραρισμένη εκδοχή του x με μηδενική φασική παραμόρφωση.

Στο παρακάτω παράδειγμα έχουμε σήμα που διαρκεί 1 δευτερόλεπτο, με συχνότητα δειγματοληψίας 100 κύκλους ,και αποτελείται από δύο ημιτονικά σήματα 3 και 40 κύκλων.

Fs = 100;
t = 0:1/Fs:1;
*x = sin(2*pi*t*3)+.25*sin(2*pi*t*40);*

Τώρα θα δημιουργήσουμε ένα φίλτρο FIR μέσου όρου 10 σημείων ,όπου θα φιλτράρουμε το διάνυσμα x χρησιμοποιώντας και τις δύο συναρτήσεις για σύγκριση (filter και filtfilt).

```
b = ones(1,10)/10;          % 10 σημείων φίλτρο μέσου όρου
y = filtfilt(b,1,x);        % μη αιτιοκρατικό φιλτράρισμα
yy = filter(b,1,x);         % κανονικό φιλτράρισμα
plot (t,x,t,y,'—',t,yy,':')
```



Αυτό που παρατηρούμε και από τις δύο συναρτήσεις είναι ότι εξαφάνισαν την ημιτονική συνιστώσα των 40 κύκλων. Ακόμα τα γράφημα δείχνει την διαφορά μεταξύ των δύο συναρτήσεων . Η γραμμή με τις παύλες (filtfilt) βρίσκεται σε φάση με το αρχικό σήμα (ημιτονικό συχνότητας 3 κύκλων) , ενώ η γραμμή με τις τελείες (filter) έχει μια καθυστέρηση περίπου 5 δειγμάτων. Ακόμα το πλάτος της γραμμής με τις παύλες έχει μικρότερο πλάτος που οφείλονται στα φαινόμενα τετραγωνισμού του πλάτους που δημιουργεί η filtfilt.

Η filtfilt μειώνει τα αρχικά μεταβατικά φαινόμενα του φίλτρου με προσεκτική επιλογή των αρχικών συνθηκών, και έχοντας εν αναμονή μια μικρή ακολουθία εισόδου ,που αντικατοπτρίζει την ακολουθία εισόδου. Για την καλύτερη δυνατή εφαρμογή της συνάρτησης πρέπει η ακολουθία προς φιλτράρισμα να είναι τουλάχιστον σε μήκος, τρεις φορές η τάξη του φίλτρου.

Υλοποίηση Φίλτρων στο Πεδίο των Συχνοτήτων

Η δυαδικότητα μεταξύ του πεδίου του χρόνου και της συχνότητας μας δίνει την δυνατότητα να πραγματοποιούμε οποιαδήποτε πράξη και στα δύο πεδία Συνήθως ένα από τα δύο πεδία μας εξυπηρετεί πιο πολύ στο να πραγματοποιούμε συγκεκριμένες λειτουργίες. Η πραγματοποίηση ενός γενικού φίλτρου IIR στο πεδίο του χρόνου, απαιτεί τον πολλαπλασιασμό του διακριτού μετασχηματισμού Fourier της ακολουθίας εισόδου με το πηλίκο του διακριτού μετασχηματισμού Fourier του φίλτρου. Ακολουθεί παράδειγμα.

```
n = length(x);
y = ifft(fft(x).*fft(b,n)./fft(a,n));
```

Τα αποτελέσματα των παραπάνω υπολογισμών είναι ίδια με τα αποτελέσματα εξόδου της συνάρτησης filter, αλλά με διαφορετική συμπεριφορά στην αρχή όπου έχουμε μεταβατικά φαινόμενα (edge effects). Για μεγάλες ακολουθίες, ο παραπάνω υπολογισμός δεν είναι αποδοτικός λόγω της μεγάλης μηδενικής επικάλυψης στον υπολογισμό FFT των συντελεστών του φίλτρου, και ακόμα ο αλγόριθμός FFT γίνεται όλο και πιο λιγότερο αποδοτικός όσο ο αριθμός των σημείων n αυξάνει.

Για φίλτρα FIR είναι δυνατό να σπάσουμε μεγάλες ακολουθίες σε μικρότερες με σκοπό να φτιάξουμε υπολογιστικά αποδοτικά μήκη FFT .

y = fftfilt(b,x)

Η παραπάνω συνάρτηση χρησιμοποιεί την μέθοδο της προσθετικής επικάλυψης για να φιλτράρει μια μεγάλη ακολουθία από πολλαπλασιαζόμενα μεσαίου μεγέθους FFTs. Η έξοδος είναι ισοδύναμη με την
filter (b,1,x).

B.1.4 Κρουστική Απόκριση

Η κρουστική απόκριση ενός ψηφιακού φίλτρου είναι η έξοδος που προκύπτει από μια ακολουθία εισόδου όπως η παρακάτω:

$$x(n) = \begin{matrix} 1. & n = 0 \\ 0, & n{=}/0 \end{matrix}$$

Στο Matlab μπορούμε να δημιουργήσουμε κρουστική ακολουθία με αρκετούς τρόπους, ένας άμεσος τρόπος είναι ο ακόλουθος:

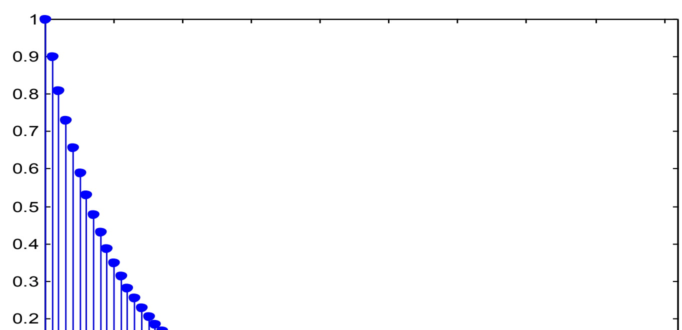
imp = [1 ; zeros(49,1)];

Η κρουστική απόκριση ενός απλού φίλτρου για b = 1 και a = [1 –0.9] είναι

h = filter(b,a,imp);

Η συνάρτηση impz απλοποιεί την όλη διαδικασία , εκλέγει των αριθμό των σημείων και μετά απεικονίζει την συνάρτηση (χρησιμοποιώντας την συνάρτηση stem).

impz (b,a)



B.1.5 Καμπύλη Απόκρισης

Το SPT έχει δυνατότητα ανάλυσης φίλτρων στο πεδίο των συχνοτήτων, τόσο για αναλογικά όσο και για ψηφιακά φίλτρα.

Ψηφιακό Πεδίο

Η συνάρτηση freqz χρησιμοποιεί ένα αλγόριθμο που βασίζεται στο γρήγορο μετασχηματισμό Fourier (FFT), για να υπολογίσει τον μετασχηματισμό z της καμπύλης απόκρισης ενός ψηφιακού φίλτρου. Το παρακάτω παράδειγμα δείχνει την σύνταξη:

[h,w] = freqz(b,a,n)

Η παραπάνω συνάρτηση επιστρέφει μία n σημείων μιγαδική καμπύλη απόκρισης.

$$H(e^{j\omega}) = \frac{b(1) + b(2)e^{-j\omega} + ... + b(nb+1)e^{-j\omega(nb)}}{a(1) + a(2)e^{-j\omega} + ... + a(na+1)e^{-j\omega(na)}}$$

Σύμφωνα με την παραπάνω απλή μορφή, η freqz δέχεται τους συντελεστές του φίλτρου διανύσματα b και α και έναν ακέραιο n όπου προσδιορίζει τον αριθμό των σημείων όπου θα γίνει ο υπολογισμός της καμπύλης απόκρισης. Η freqz επιστρέφει την μιγαδική καμπύλη απόκρισης στο διάνυσμα h, και τα πραγματικά σημεία συχνοτήτων στο διάνυσμα w σε ακτίνια ανά δευτερόλεπτο. Στην freqz μπορούμε να εισαγάγουμε και άλλες παραμέτρους όπως τη συχνότητα δειγματοληψίας, ή ένα διάνυσμα αυθαίρετων σημείων συχνότητας. Στο παρακάτω παράδειγμα έχουμε μια καμπύλη απόκρισης 256 σημείων για ένα φίλτρο Chebyshev τύπου I, δωδέκατης τάξης. Η συχνότητα δειγματοληψίας είναι 1000 κύκλοι.

[b,a] = cheby1(12,0.5,200/500);
[h,f] = freqz(b,a,256,1000);

Λόγω ότι υπάρχει η παράμετρος της συχνότητας δειγματοληψίας , η συνάρτηση θα επιστρέψει στο διάνυσμα f, 256 σημεία συχνότητας από 0 έως Fs/2 ,όπου θα χρησιμοποιηθούν στον υπολογισμό της καμπύλης απόκρισης.

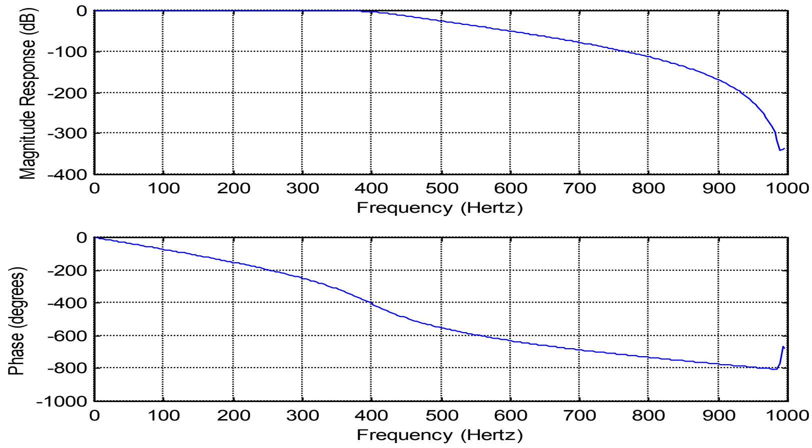
Σημείωση: Κανονικοποίηση Συχνότητας. Στο SPT χρησιμοποιούμε κατά σύμβαση ότι η μοναδιαία συχνότητα είναι η συχνότητα Nyquist, όπου είναι η μισή της συχνότητας δειγματοληψίας. Η παράμετρος της συχνότητας αποκοπής για όλα τα βασικά φίλτρα , κανονικοποιείται με την συχνότητα Nyquist. Για ένα σύστημα με συχνότητα δειγματοληψίας 1000 κύκλων θα έχουμε, η συχνότητα των 300 κύκλων θα έχει την την τιμή 300/500 = 0.6. Για να μετατρέψουμε την κανονικοποιημένη συχνότητα σε κυκλική συχνότητα δεν έχουμε παρά να πολλαπλασιάσουμε με π. Για να μετατρέψουμε την κανονικοποιημένη συχνότητα σε Hertz , πολλαπλασιάζουμε με το μισό της συχνότητας δειγματοληψίας.

Όταν καλούμε την συνάρτηση freqz χωρίς τα εξωτερικά διανύσματα η freqz αυτομάτως σχεδιάζει και το πλάτος και την φάση συναρτήσει της συχνότητας. Στο παρακάτω παράδειγμα έχουμε φίλτρο Butterworth χαμηλών συχνοτήτων ένατης τάξης , με συχνότητα αποκοπής τους 400 κύκλους ,και συχνότητα δειγματοληψίας 2000 κύκλους.

[b,a] = butter(9,400/1000);

Τώρα θα υπολογίσουμε την μιγαδική καμπύλη απόκρισης (256 σημείων), και θα απεικονίσουμε το πλάτος κ

freqz(b,a,256,2000)



Η freqz μπορεί να δεχθεί ένα διάνυσμα από αυθαίρετα σημεία συχνότητας για να υπολογίσει σύμφωνα με αυτά τα σημεία την καμπύλη απόκρισης. Ακολουθεί ένα παράδειγμα.

w = linspace(0,pi);
h = freqz(b,a,w);

Υπολογίζουμε την μιγαδική καμπύλη απόκρισης στα σημεία συχνοτήτων όπου ορίζονται από το διάνυσμα w και τα διανύσματα b και a. Τα σημεία συχνότητας μπορεί να είναι από 0 έως π. Για να δημιουργήσουμε ένα διάνυσμα συχνότητας όπου τα σημεία συχνότητας θα ξεκινούν από 0 μέχρι την συχνότητα δειγματοληψίας, θα πρέπει στην λίστα των παραμέτρων της συνάρτησης να βάλουμε την συχνότητα δειγματοληψίας και το διάνυσμα συχνότητας.

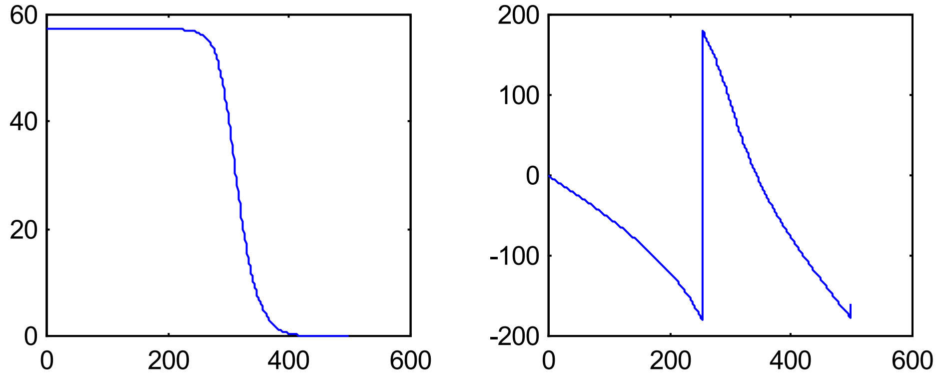
Αναλογικό Πεδίο

Η συνάρτηση freqs απεικονίζει την καμπύλη απόκρισης ενός αναλογικού φίλτρου όπου προσδιορίζεται από δύο συντελεστές διανυσμάτων το b και a. Η λειτουργία της είναι όμοια με την freqz. Μπορούμε να προσδιορίσουμε τον αριθμό των σημείων συχνότητας (σε περίπτωση μη προσδιορισμού η συνάρτηση χρησιμοποιεί 200 σημεία), να προσδιορίσουμε ένα διάνυσμα με αυθαίρετα σημεία συχνότητας, και φυσικά να απεικονίσουμε την απόκριση του πλάτους και της φάσης του φίλτρου.

Μέτρο και Φάση

Το Matlab παρέχει λειτουργίες για να εξάγουμε το πλάτος και την φάση ενός διανύσματος καμπύλης απόκρισης h. Η συνάρτηση abs επιστρέφει το μέτρο και η συνάρτηση angle επιστρέφει την φάση σε ακτίνια. Στο παρακάτω παράδειγμα εξάγουμε και απεικονίζουμε το μέτρο και την φάση ενός φίλτρου Butterworth.

```
[b,a] = butter(6,300/500);
[h,w] = freqz(b,a,512,1000);
m = abs(h);
p = angle(h);
semilogy(w,com
plot(w,p*180/pi,
```

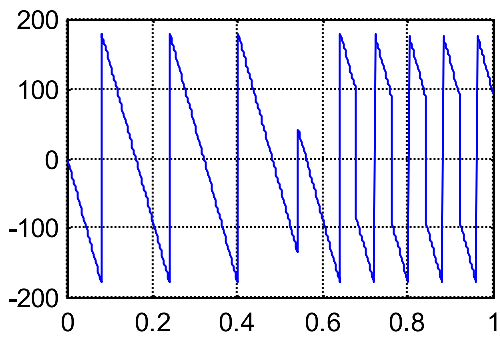


Η συνάρτηση unwrap είναι επίσης χρήσιμη στην ανάλυση συχνοτήτων. Η unwrap ξετυλίγει την φάση και την κάνει συνεχή. Στις ασυνέχειες των 360 μοιρών προσθέτει πολλαπλάσια των +/- 360 μοιρών όπου χρειάζεται. Στο παρακάτω παράδειγμα θα δούμε την χρησιμότητα της συνάρτησης, στον σχεδιασμό ενός φίλτρου FIR 25 βαθμού διέλευσης χαμηλών συχνοτήτων.

```
h = fir1(25,0.4);
```

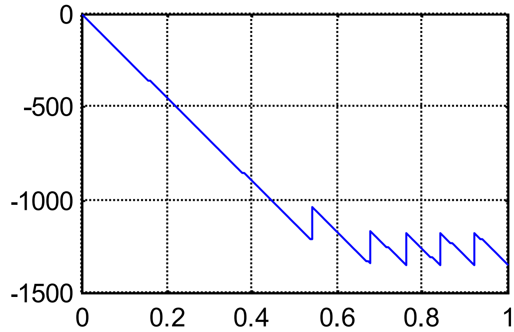
Για να πάρουμε την καμπύλη απόκρισης θα χρησιμοποιήσουμε την γνωστή συνάρτηση freqz (η φάση απεικονίζεται σε μοίρες).

```
[H,f] = freqz(h,1,512,2);
plot(f,angle(H)*180/pi); grid
```



Είναι δύσκολο να προσδιορίσουμε τα πηδήματα των 360 μοιρών από αυτά των 180 μοιρών που υποδηλώνουν τα μηδενικά στην απόκριση συχνότητας.. Θα χρησιμοποιήσουμε την συνάρτηση unwrap για να εξαφανίσουμε τα πηδήματα των 360 μοιρών.

*plot(f,unwrap(angle(H))*180/pi); grid*



Καθυστέρηση

Η συνολική καθυστέρηση είναι η μέτρηση της μέσης τιμής της καθυστέρησης ενός φίλτρου συνάρτησει της συχνότητας. Προσδιορίζεται σαν η πρώτη αρνητική παράγωγος της φασικής απόκρισης του φίλτρου. Εάν η μιγαδική καμπύλη απόκρισης ενός φίλτρου είναι $H(e^{jw})$, τότε η καθυστέρηση είναι:

$$\tau_g(\omega) = -\frac{d\theta(\omega)}{d\omega}$$

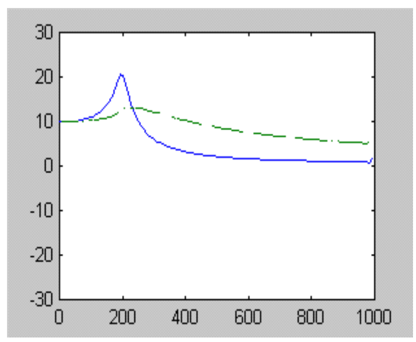
όπου θ είναι η φάση του $H(e^{jw})$. Για να την υπολογίσουμε με το Matlab έχουμε:

[gd,w] = grpdelay(b,a,n)

οπου επιστρέφεται ένα διάνυσμα n σημείων συνολικής καθυστέρησης, tg(ω), ενός ψηφιακού φίλτρου όπου προσδιορίζεται από τα b και a , και χρησιμοποιεί συχνότητες από το διάνυσμα w.

Η καθυστέρηση φάσης ενός φίλτρου είναι ίσο με την φάση (με αρνητικό πρόσημο) διά την συχνότητα. $\tau_p(\omega) = -\frac{\theta(\omega)}{\omega}$

Στο παρακάτω παράδειγμα απεικονίζεται η φασική και η ολική καθυστέρηση ενός συστήματος στην ίδια γραφική παράσταση.



[b,a] = butter(10,200/1000);
gd = grpdelay(b,a,128);
[h,f] = freqz(b,a,128,2000);
pd = -unwrap(angle(h))(2000/(2*pi))./f;*
plot(f,gd,'-',f,pd,'- -')';   axis[(0 1000 –30 30])

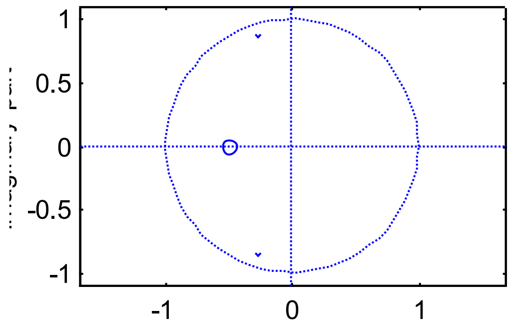
Ανάλυση Πόλων και Μηδενικών

Η συνάρτηση zplane σχεδιάζει τους πόλους και τα μηδενικά ενός γραμμικού συστήματος. Στο παρακάτω παράδειγμα έχουμε ένα μηδέν στο –0.5 και ένα ζεύγος μιγαδικού πόλου.

zer = -0.5;
*pol = .9*exp(j*2*pi*[-0.3 .3]');*

Η γραφική παράσταση του φίλτρου είναι:

zplane (zer,pol)



Για συστήματα που η συνάρτηση μεταφοράς είναι σε μορφή πόλων – μηδενικών, τα στοιχεία z και p τα εισαγάγουμε στην zplane σε διανύσματα γραμμής

zplane (z,p)

Για συστήματα που έχουμε συνάρτηση μεταφοράς , τα στοιχεία b και a τα εισαγάγουμε στην zplane σε διανύσματα στήλης.

zplane (b,a)

Στην παραπάνω συνάρτηση η zplane βρίσκει τις ρίζες των b και a χρησιμοποιώντας την συνάρτηση roots, και σχεδιάζει τα αποτελέσματα.

B.2 Σχεδιασμός Φίλτρων

B.2.1 Προδιαγραφές και Απαιτήσεις στον Σχεδιασμό Φίλτρων

Με το SPT παρέχονται αρκετές λειτουργίες και συναρτήσεις στον σχεδιασμό φίλτρων. Σε αυτό το κεφάλαιο θα αναλυθούν οι διαφορετικές μεθοδολογίες, τα προβλήματα και τα εργαλεία για τον σχεδιασμό φίλτρων IIR (Infinity Impulse Response) και FIR (Finite Impulse Response).

Ο σκοπός ενός ψηφιακού φίλτρου είναι να διαφοροποιήσει την ακολουθία των αριθμών έτσι ώστε να πετύχει την επιθυμητή διαδικασία φιλτραρίσματος. Μία πιθανή περίπτωση είναι να αφαιρέσουμε θόρυβο πάνω από 30 κύκλους σε μια ακολουθία δεδομένων όπου η συχνότητα δειγματοληψίας ήταν στους 100 κύκλους. Μια πιο αυστηρή προδιαγραφή θα μπορούσε να εμπεριέχει την κυμάτωση στην ζώνη διέλευσης, την εξασθένιση στην ζώνη αποκοπής ή το εύρος μετάβασης. Μια πιο ακριβής προδιαγραφή θα μπορούσε να θέτει τις παραπάνω προδιαγραφές με το ελάχιστο δυνατό τάξης φίλτρο, ή να ζητούσε την κατασκευή φίλτρου με αυθαίρετη απόκριση του μέτρου, ή χρησιμοποίηση φίλτρου FIR.

Οι μέθοδοι σχεδιασμού φίλτρου διαφέρουν βασικά ως προς τις προσδιοριζόμενες επιδόσεις. Για ελλιπείς προσδιοριζόμενες απαιτήσεις όπως στην παραπάνω περίπτωση ,ένα φίλτρο Butterworth IIR ,είναι αρκετό. Για τον σχεδιασμό ενός φίλτρου διέλευσης χαμηλών Butterworth με συχνότητα αποκοπής 30 κύκλων πέμπτης τάξεως μπορούμε να χρησιμοποιήσουμε την παρακάτω συνάρτηση, όπου εισάγουμε μία ακολουθία δεδομένων σε ένα διάνυσμα x.

```
[b,a]  =butter(5,30/50);
 y  = filter(b,a,x);
```

Το δεύτερο στοιχείο εισόδου στην συνάρτηση butter δείχνει την συχνότητα αποκοπής, όπου με την κανονικοποίηση έχουμε την μισή συχνότητα δειγματοληψίας.

Κανονικοποίηση συχνότητας στο SPT

Όλες οι συναρτήσεις για τον σχεδιασμό φίλτρου λειτουργούν με κανονικοποιημένες συχνότητες, επομένως δεν απαιτείται από το σύστημα ο ρυθμός δειγματοληψίας σαν ένα επιπλέον στοιχείο εισόδου. Το SPT χρησιμοποιεί αυτή την σύμβαση, έτσι ώστε η μοναδιαία συχνότητα να είναι η συχνότητα Nyquist, όπου προσδιορίζεται σαν η μισή της συχνότητας δειγματοληψίας.
Πιο αυστηρές απαιτήσεις για φίλτρα, εμπεριέχουν στοιχεία όπως η κυμάτωση στην ζώνη διέλευσης (Rp σε dB), εξασθένιση στην ζώνη αποκοπής (Rs σε dB), και το εύρος μετάβασης (Ws-Wp, σε Hertz). Μπορούμε να σχεδιάσουμε φίλτρα Butterworth, ChebyshevI, ChebyshevII, και ελλειπτικά φίλτρα που να ικανοποιούν τις παραπάνω προδιαγραφές επιδόσεων. Το SPT διαθέτει συναρτήσεις που προσδιορίζουν τον ελάχιστο βαθμό φίλτρου για δεδομένες προδιαγραφές.

B.2.2 Σχεδιασμός Φίλτρου IIR (Infinite Filter Response)

Το βασικό πλεονέκτημα των φίλτρων IIR σε σχέση με τα φίλτρα FIR είναι ότι ικανοποιούν για δεδομένες προδιαγραφές πολύ μικρότερου βαθμό φίλτρο σε σχέση με τα φίλτρα FIR. Εκτός αυτού τα IIR φίλτρα έχουν μη γραμμική φάση, και η επεξεργασία των δεδομένων στο Matlab πραγματοποιείται σε μη πραγματικό χρόνο. Αυτό γίνεται λόγω του ότι χρειάζεται όλη η ακολουθία εισόδου πριν την διαδικασία φιλτραρίσματος. Αυτό επιτρέπει μηδενικής φάσης φιλτράρισμα (με χρήση της συνάρτησης filtfilt), όπου εξαφανίζει τις φασικές μη γραμμικότητες ενός IIR.

Τα κλασικά IIR φίλτρα, Butterworth, Chebyshev τύπου I και II, ελλειπτικά και Bessel, μπορούν να κατασκευαστούν με προσέγγιση της ιδανικής χαρακτηριστικής με διαφόρους τρόπους. Το SPT παρέχει συναρτήσεις για να κατασκευαστούν όλοι οι τύποι των κλασσικών φίλτρων IIR, και στο αναλογικό και στο ψηφιακό πεδίο (εκτός της συνάρτησης για το φίλτρο Bessel που είναι μόνο για το αναλογικό πεδίο), σε διατάξεις διελεύσεις χαμηλών και υψηλών συχνοτήτων, και σε φίλτρα διέλευσης και αποκοπής ζώνης. Για τους περισσότερους τύπους φίλτρων, μπορούμε να βρούμε τον μικρότερο βαθμό του φίλτρου όπου ταιριάζει στις προδιαγραφές.

Η συνάρτηση yulewalk κατασκευάζει άμεσα ένα φίλτρο με απόκριση που να προσεγγίζει την επιθυμητή λειτουργία. Αυτός είναι ένα τρόπος για να δημιουργήσουμε ένα φίλτρο διέλευσης πολλαπλών ζωνών. Μπορούμε επίσης να χρησιμοποιήσουμε την παραμετρική μοντελοποιήση για να σχεδιάσουμε φίλτρο IIR. Ο παρακάτω πίνακας συνοψίζει τις διαφόρους μεθόδους και τις αντίστοιχες συναρτήσεις.

Μέθοδος	**Περιγραφή**	**Συναρτήσεις**
Με Χρήση Αναλογικού Πρωτοτύπου	Χρησιμοποιώντας τους πόλους και τα μηδενικά σε ένα κλασικό φίλτρο διέλευσης χαμηλών στον συνεχή χρόνο (Laplace), Παίρνομε ένα ψηφιακό φίλτρο διαμέσου μετασχηματισμού στην συχνότητα και	Συναρτήσεις Ολοκληρωμένης σχεδίασης: Besself, butter, cheby1 ,cheby2 ,ellip Συναρτήσεις προσδιορισμού του βαθμού του φίλτρου: Buttord, cheb1ord, cheb2ord, ellipord Συναρτήσεις μετασχηματισμού συχνότητας: Lp2bp, lp2bs, lp2hp, lp2lp Συναρτήσεις μετασχηματισμού συστημάτων από τον συνεχή στον διακριτό χρόνο : Bilinear, impinvar
Άμεσος Σχεδιασμός	Άμεσος σχεδιασμός ψηφιακού φίλτρου με τμηματική προσέγγιση της γραμμικής απόκρισης του μέτρου.	Yulewalk
Παραμετρική Μοντελοποίηση	Εύρεση ενός ψηφιακού φίλτρου όπου κατά προσέγγιση	Συναρτήσεις μοντελοποίησης στο πεδίο του χρόνου. Lpc, prony, stmcb Συναρτήσεις μοντελοποίησης στο Πεδίο των συχνοτήτων. Invfreqs, invfreqz
Γενικοποιημένος Σχεδιασμός Butterworth	Σχεδιασμός φίλτρων διέλευσης χαμηλών συχνοτήτων Butterworth με περισσότερα μηδενικά από πόλους.	Maxflat

Κλασική Σχεδίαση Φίλτρου IIR Χρησιμοποιώντας Αναλογικό Πρότυπο.

Η αρχή στον σχεδιασμό φίλτρων IIR στο SPT βασίζεται στην τεχνική της μετατροπής ενός κλασικού φίλτρου διέλευσης χαμηλών στο ψηφιακό ισοδύναμό του. Τα επόμενα τμήματα περιγράφουν τον τρόπο σχεδιασμού φίλτρων και συνοψίζουν τα χαρακτηριστικά φίλτρων διαφόρων τύπων.

Ολοκληρωμένη κλασσική σχεδίαση φίλτρου IIR.

Μπορούμε εύκολα να σχεδιάσουμε φίλτρα οποιαδήποτε βαθμού, σε διάταξη διέλευσης χαμηλών, διέλευσης υψηλών, διέλευσης ζώνης, και αποκοπής ζώνης, χρησιμοποιώντας τις παρακάτω συναρτήσεις

Τύπος Φίλτρου	Συνάρτηση Σχεδίασης
Butterworth	[b,a] = butter(n,Wn,options) [z,p,k] = butter(n,Wn,options) [A,B,C,D] = butter(n,Wn,options)
Chebyshev type I	[b,a] = cheby1(n,Rp,Wn,options) [z,p,k] = cheby1(n,Rp,Wn,options) [A,B,C,D] = cheby1(n,Rp,Wn,options)
Chebysev type II	[b,a] = cheby2(n,Rs,Wn,options) [z,p,k] = cheby2(n,Rs,Wn,options) [A,B,C,D] = cheby2(n,Rs,Wn,options)
Elliptic	[b,a] = ellip(n,Rp,Rs,Wn,options) [z,p,k] = ellip(n,Rp,Rs,Wn,options) [A,B,C,D] = ellip(n,Rp,Rs,Wn,options)
Bessel (Αναλογικά μόνο)	[b,a] = besself(n,Wn,options) [z,p,k] = besself(n,Wn,options) [A,B,C,D] = besself(n,Wn,options)

Η κάθε μία από τις παραπάνω συναρτήσεις επιστρέφουν ένα φίλτρο διέλευσης χαμηλών. Το μόνο που χρειάζεται να προσδιοριστεί είναι η συχνότητα αποκοπής Wn σε κανονικοποιημένη συχνότητα. Για φίλτρα διέλευσης υψηλών θέτουμε στοιχείο εισόδου 'high' στην είσοδο των παραμέτρων και συγκεκριμένα στην option. Για φίλτρο διέλευσης ή αποκοπής ζώνης, προσδιορίζουμε το Wn σαν δύο στοιχείων διάνυσμα περιέχοντας τις ακραίες συχνότητες διέλευσης, και θέτοντας το στοιχείο εισόδου 'stop' σε διάταξη ζώνης αποκοπής.

Παρακάτω δίνονται παραδείγματα ψηφιακών φίλτρων:

```
[b,a] = butter(5,.4);                 % Φίλτρο διέλευσης χαμηλών Butterworth
[b,a] = chebys1(4,1,[.4 .7]);         % Φίλτρο διέλευσης ζώνης Chebyshevτύπου I
[b,a] = cheby2(6,60, .8,'high');      % Φίλτρο διέλευσης υψηλών Chebyshev τύπου II
[b,a] = ellip(3,1,60,[.4 .7],'stop'); % Φίλτρο αποκοπής ζώνης ελλειπτικό
```

Για να σχεδιάσουμε ένα αναλογικό φίλτρο, πιθανότατα για εξομοίωση, τοποθετούμε ένα 's' στην είσοδο παραμέτρων της συνάρτησης, και προσδιορίζουμε τις συχνότητες αποκοπής σε ακτίνια ανά δευτερόλεπτο:

[b,a] = butter(5,.4,'s'); % αναλογικό φίλτρο Butterworth

Όλες οι συναρτήσεις σχεδιασμού φίλτρων επιστρέφουν τα αποτελέσματα με τρεις διαφορετικούς τρόπους

Σχεδιασμός Φίλτρου IIR Στο Πεδίο Της Συχνότητας

Το SPT παρέχει συναρτήσεις για τον υπολογισμό του ελάχιστου βαθμού του φίλτρου που να ικανοποιούν όμως τις προδιαγραφές.

Τύπος Φίλτρου	Συνάρτηση Προσδιορισμού Βαθμού Τάξης
Butterworth	[n,Wn] = buttord(Wp,Ws,Rp,Rs)
Chebyshev type I	[n,Wn] = cheb1ord(Wp,Ws,Rp,Rs)
Chebyshev type II	[n,Wn] = cheb2ord(Wp,Ws,Rp,Rs)
Elliptic	[n,Wn] = ellipord(Wp,Ws,Rp,Rs)

Αυτές οι συναρτήσεις είναι χρήσιμες σε συνδιασμό με τις συναρτήσεις σχεδιασμού φίλτρων. Στο παρακάτω παράδειγμα θα σχεδιάσουμε ένα φίλτρο διέλευσης ζώνης απο 1000 έως 2000 κύκλους, οι ζώνες αποκοπής ξεκινάνε 500 κύκλους εκατέρωθεν της κάθε πλευράς, συχνότητα δειγματοληψίας 10 Khz , 1db κυμάτωση στην ζώνη διέλευσης , και 60 dB εξασθένιση στη ζώνη αποκοπής.

[n,Wn] = buttord([1000 2000] / 5000, [500 2500] / 5000,1,60)

```
n =
    12

Wn =
     0.1951      0.4080
```

[b,a] = butter(n,Wn);

Ένα ελλειπτικό φίλτρο που ικανοποιεί τα παραπάνω χαρακτηριστικά μπορεί να κατασκευαστεί όπως παρακάτω:

[n,Wn] = ellipord([1000 2000] / 5000, [500 2500] / 5000,1,60)

```
n =
     5
```

Wn =
0.2000 0.4000

[b,a] = ellip(n,1,60,Wn);

Άμεση Σχεδίαση Φίλτρου IIR

Στο SPT χρησιμοποιούμε τον όρο 'άμεσοι μέθοδοι' για να περιγράψουμε μεθόδους για τον σχεδιασμό φίλτρων που βασίζονται σε προδιαγραφές του διακριτού πεδίου. Αντίθετα με την μέθοδο του αναλογικού προτύπου, οι μέθοδοι άμεσου σχεδιασμού δεν επικεντρώνονται στα πρότυπα των φίλτρων διέλευσης χαμηλών ή υψηλών και στα φίλτρα διέλευσης ή αποκοπής ζώνης. Αυτές οι συναρτήσεις σχεδιάζουν φίλτρα με αυθαίρετη απόκριση, πιθανότατα πολλαπλών ζωνών. Σε αυτή την παράγραφο θα δούμε την συνάρτηση yulewalk , που χρησιμοποιείται ειδικά για σχεδιασμό φίλτρων. Η yulewalk σχεδιάζει αναδρομικά ψηφιακά φίλτρα IIR προσαρμόζοντας προσδιοριζόμενες καμπύλες απόκρισης. Το όνομα της yulewalk αντικατοπτρίζει την μέθοδο εύρεσης των συντελεστών (του παρανομαστή) του φίλτρου. Βρίσκει τον αντίστροφο FFT του επιθυμητού φάσματος ισχύος και λύνει της 'τροποποιημένες εξισώσεις Yule-Walker' χρησιμοποιώντας τα αποτελέσματα των δειγμάτων από την συνάρτηση αυτοσυσχέτισης. Στο παρακάτω παράδειγμα έχουμε:

[b,a] = yulewalk(n,f,m)

Η συνάρτηση yulewalk επιστρέφει διάνυσμα γραμμής b και a όπου εμπεριέχονται οι συντελεστές n+1 αριθμητή και παρανομαστή του βαθμού του φίλτρου IIR όπου τα χαρακτηριστικά πλάτους και συχνότητας προσεγγίσουν τα δεδομένα διανύσματα f και m. Το f είναι ένα διάνυσμα των σημείων συχνότητας όπου κινείται από το 0 έως το 1, όπου το 1 αντιπροσωπεύει την συχνότητα Nyquist. Το διάνυσμα m αντιπροσωπεύει την επιθυμητή απόκριση πλάτους ,στα σημεία του διανύσματος f. Με τα διανύσματα m και f μπορούμε να κατασκευάσουμε οποιαδήποτε γραμμική μορφή απόκρισης πλάτους, εμπεριέχοντας πολυζωνική απόκριση. Η αντίστοιχη εντολή στα φίλτρα FIR είναι η fir2 , όπου επίσης σχεδιάζουμε φίλτρα βασισμένα σε αυθαίρετη γραμμική απόκριση πλάτους.

Η συνάρτηση yulwalk δεν δέχεται πληροφορίες φάσης , και το φίλτρο που προκύπτει δεν είναι το βέλτιστο. Στο παρακάτω παράδειγμα έχουμε ένα πολυζωνικό φίλτρο κατασκευασμένο με την συνάρτηση yulwalk, όπου σχεδιάζουμε την καμπύλη απόκρισής του.

```
m = [0 0 1 1 0 0 1 1 0 0];
f = [0 0.1 0.2 0.3 0.4 0.5 0.6 0.7 0.8 1];
[b,a] = yelewalk(10,f,m);
[h,w] = freqz(b,a,128);
plot(f,m,w/pi,abs(h))
```

Γενικευμένη Σχεδίαση Φίλτρου Butterworth

Στο SPT η συνάρτηση maxflat μας επιτρέπει την κατασκευή γενικευμένων φίλτρων Butterworth, φίλτρα με διαφορετικό αριθμό πόλων και μηδενικών. Αυτό το χαρακτηριστικό είναι μερικές φορές επιθυμητό σε ορισμένες εφαρμογές όπου οι πόλοι είναι πολύ πιο ακριβοί στον υπολογισμό από τα μηδενικά. Η συνάρτηση maxflat είναι ίδια με την butter αλλά με την βασική διαφορά ότι εδώ προσδιορίζουμε δύο βαθμούς (ένα για τον αριθμητή και ένα για παρανομαστή). Τα φίλτρα που προκύπτουν είναι τα βέλτιστα για κάθε βαθμό αριθμητή και παρανομαστή.

Στο παρακάτω παράδειγμα, χρησιμοποιούμε τη συνάρτηση (με ίδιο βαθμό στον αριθμητή και στον παρανομαστή) maxflat και butter. Όπως παρατηρούμε είναι ακριβώς ίδια.

[b,a] = maxflat(3,3,0.25)
b =
0.0317 0.0951 0.0951 0.0317

a =
1.0000 -1.4590 0.9104 -0.1978

[b,a] = butter(3,0.25)
b =
0.0317 0.0951 0.0951 0.0317

a =
1.0000 -1.4590 0.9104 -0.1978

Η maxflat είναι πιο βολική διότι μπορούμε να σχεδιάσουμε φίλτρο με περισσότερα μηδενικά παρά πόλους.

[b,a] = maxflat(3,1,0.25)
b =
0.0950 0.2849 0.2849 0.0950

a=
1.0 *-0.2402*

Το τρίτο στοιχείο εισόδου στην συνάρτηση είναι η συχνότητα μισής ισχύος, δηλαδή μια συχνότητα μεταξύ 0 και 1 με επιλεγόμενη απόκριση πλάτους $1/\sqrt{2}$

Μπορούμε επίσης να σχεδιάσουμε φίλτρα με γραμμική φάση που να έχουν μεγιστοποιημένη επίπεδη απόκριση χρησιμοποιώντας το προαιρετικό στοιχείο εισόδου 'sym' όπως στο παρακάτω παράδειγμα.

maxflat(4,'sys',0.3)
ans =
0.0331 0.2500 0.4337 0.2500 0.0331

B.2.3 Σχεδιασμός Φίλτρου FIR (Finite Impulse Response)

Τα ψηφιακά φίλτρα με πεπερασμένη διάρκεια κρουστικής απόκρισης (όλο μηδενικά ή φίλτρα FIR) έχουν και πλεονεκτήματα και μειονεκτήματα σε σχέση με τα φίλτρα IIR.

Τα φίλτρα FIR έχουν τα ακόλουθα πλεονεκτήματα:

- Έχουν γραμμική φασική συμπεριφορά
- Είναι πάντοται σταθερά

- Οι μέθοδοι σχεδίασης είναι συνήθως γραμμικές.
- Μπορούν να υλοποιηθούν εύκολα σε υλισμικό.
- Έχουν πεπερασμένη διάρκεια τα αρχικά μεταβατικά φαινόμενα

Το βασικό μειονέκτημα των φίλτρων FIR είναι ότι απαιτείται μεγαλύτερης τάξης φίλτρο σε σχέση με τα IIR για να πετύχουμε ίδια επίπεδα απόδοσης. Με άμεσο αποτέλεσμα η καθυστέρηση αυτών των φίλτρων να είναι κατά πολύ μεγαλύτερη για ίδιας απόδοσης φίλτρα IIR.

Μέθοδος	Συνάρτηση Matlab
Παράθυρα	fir1, fir2, kaiserord
Πολλαπλών Ζωνών με Ζώνες Μετάβασης	firls, remez, remezord
Ελαχίστων Τετραγώνων	fircls,fircls1
Αυθαίρετης Απόκριση	Cremez
Αυξανόμενου Συνημίτονου	Firrcos

Φίλτρα Γραμμικής Φάσης

Εκτός την συνάρτηση cremez, όλες οι υπόλοιπες συναρτήσεις σχεδιασμού φίλτρων FIR σχεδιάζουν φίλτρα μόνο με γραμμική φάση. Οι συντελεστές (taps) τέτοιων φίλτρων υπακούουν σε σχέσεις άρτιας ή περιττής συμμετρίας. Εξαρτώμενη από την συμμετρία, από τον βαθμό του φίλτρου n (εάν είναι άρτιος ή περιττός ο βαθμός), ένα γραμμικό φίλτρο έχει αναπόφευκτους εγγενείς περιορισμούς στην απόκριση συχνότητάς του.

Τύπος Φίλτρου Γραμμικής Φάσης	Βαθμός Φίλτρου n	Συμμετρικότητα Συντελεστών	Απόκριση H(f),f=0	Απόκριση H(f),f=1
Τύπος I	Άρτιο	Άρτια	*	*
Τύπος II	Περιττό	b(k)=b(n+2-k),k=1,…,n+1	*	H(1) = 0
Τύπος III	Άρτιο	Περιττή	H(0) = 0	H(1) = 0
Τύπος IV	Περιττό	B(k)=-b(n+2-k),k=1,…,n+1	H(0) = 0	*

Όπου * δεν υπάρχουν περιορισμοί.

Η καθυστέρηση φάσης και η συνολική καθυστέρηση (group delay) σε ένα φίλτρο γραμμικής φάσης FIR είναι ίσες και σταθερές σε όλη την ζώνη συχνοτήτων. Σε ένα φίλτρο γραμμικής φάσης FIR βαθμού n , η συνολική καθυστέρηση είναι n/2, και επομένως το φιλτραρισμένο σήμα καθυστερεί n/2 φορές. Αυτή η ιδιότητα διατηρεί αναλλοίωτο το σχήμα του σήματος στην ζώνη διέλευσης, και δεν υπάρχει φασική παραμόρφωση.

Όλες οι συναρτήσεις fir1, fir2, ,firls, remez, fircls, fircls1, και firrcos σχεδιάζουν γραμμικά φίλτρα FIR τύπου I και II. Τα φίλτρα firls και remez σχεδιάζουν γραμμικά φίλτρα FIR τύπου III και IV έχοντας τοποθετήσει στην είσοδο της συνάρτησης τις σημαίες ‘hilbert’ ή ‘differentiator’. Η συνάρτηση cremez μπορεί να σχεδιάσει κάθε τύπο φασικά γραμμικού ή μη φίλτρου.

Σχεδιασμός Φίλτρου FIR με την Μέθοδο των Παραθύρων

Αν αναλογιστούμε ένα ιδανικό ψηφιακό φίλτρο διέλευσης χαμηλών με συχνότητα αποκοπής ωο rad/sec. Αυτό το φίλτρο θα έχει πλάτος 1 για όλες τις συχνότητες που είναι μικρότερες από το ωο, και το πλάτος του θα είναι 0 για συχνότητες που η τιμή τους θα είναι μεταξύ ωο και π. Η κρουστική απόκριση του συστήματος h(n) θα είναι ίση με:

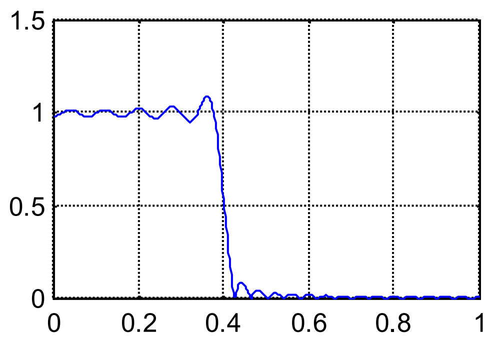
$$h(n) = \frac{1}{2\pi}\int_{-\pi}^{\pi} H(\omega)e^{j\omega n}d\omega = \frac{1}{2\pi}\int_{-\omega_o}^{\omega_o} e^{j\omega n}d\omega = \frac{\omega_o}{\pi}\sin c(\frac{\omega_o}{\pi}n)$$

Αυτό το φίλτρο δεν μπορεί να κατασκευαστεί διότι η κρουστική του απόκριση είναι απείρου μήκους και μη αιτιοκρατική. Για να δημιουργήσουμε μια κρουστική απόκριση πεπερασμένης διάρκειας, τοποθετούμε ένα παράθυρο. Διατηρώντας το κεντρικό τμήμα της κρουστικής απόκρισης μέσα στο παράθυρο, πετυχαίνουμε την κατασκευή ενός φίλτρου FIR γραμμικής φάσης. Στο παρακάτω παράδειγμα έχουμε ένα φίλτρο διέλευσης χαμηλών μήκους 51 με συχνότητα αποκοπής ωο 0.4π rad/sec:

*b = 0.4*sinc(0.4*(-25:25));*

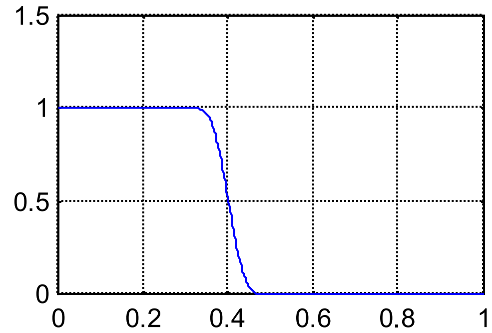
Το παράθυρο που τοποθετήσαμε είναι ένα απλό ορθογώνιο παράθυρο. Σύμφωνα με το θεώρημα του Parseval΄s, αυτό το φίλτρο έχει μήκος 51 και προσεγγίζει πολύ καλά ένα ιδανικό φίλτρο διέλευσης χαμηλών. Για να παρατηρήσουμε την απόκριση συχνότητας του παραπάνω φίλτρου θα έχουμε:

[H,w] = freqz(b,1,512,2);
plot(w,abs(H)), grid



Παρατηρούμε ότι υπάρχουν κυματώσεις στην απόκριση ιδιαίτερα στην άκρη των ζωνών. Τα φαινόμενα Gibbs δεν εξαφανίζονται όσο το μήκος του φίλτρου αυξάνει, αλλά ένα μη ορθογώνιο παράθυρο μπορεί να μειώσει το πλάτος αυτών. Πολλαπλασιασμός με ένα παράθυρο στο πεδίο του χρόνου προκαλεί συνέλιξη (εξομάλυνση) στο πεδίο των συχνοτήτων. Στο παρακάτω παράδειγμα θα εφαρμόσουμε ένα παράθυρο Hamming μήκους 51 στο φίλτρο για να παρατηρήσουμε την δράση.

```
b = b.*hamming(51)';
[H,w] = freqz(b, 1, 512, 2);
plot(w,abs(H)),grid
```



Όπως μπορούμε να δούμε οι κυματώσεις έχουν μειωθεί πάρα πολύ. Αυτή η βελτίωση όμως έχει γίνει εις βάρος του εύρους μεταγωγής (στη περίπτωση όπου χρησιμοποιούμε το παράθυρο, μας παίρνει περισσότερο για να μεταβούμε από την ζώνη διέλευση στην ζώνη αποκοπής) και της βελτιστοποίησης (στην έκδοση του φίλτρου με το παράθυρο δεν ελαχιστοποιείται το συνολικό τετραγωνικό σφάλμα).

Οι συναρτήσεις fir1 και fir2 βασίζονται στην διαδικασία χρησιμοποίησης παραθύρων. Δίνοντας τον βαθμό του φίλτρου και την περιγραφή του ιδανικά επιθυμητού φίλτρου, αυτές οι συναρτήσεις επιστρέφουν τον αντίστροφο μετασχηματισμό Fourier (με χρήση παραθύρου) αυτού του ιδανικού φίλτρου. Και οι δύο συναρτήσεις χρησιμοποιούν παράθυρα Hamming σαν προεπιλογή αλλά μπορούμε να επιλέξουμε όποιο παράθυρο χρειαστούμε.

Σχεδιασμός Φίλτρου FIR μιας Ζώνης : fir1

Η συνάρτηση fir1 ενσωματώνει την κλασσική μέθοδο κατασκευής φίλτρων FIR γραμμικής φάσης με χρήση παραθύρων. Η συνάρτηση έχει κοινά γνωρίσματα με τις συναρτήσεις σχεδιασμού φίλτρων IIR όπου μπορούμε να δημιουργήσουμε τις τυπικές διατάξεις φίλτρων: διέλευσης χαμηλών και υψηλών, αποκοπής και διέλευσης ζώνης.

Στο παρακάτω παράδειγμα έχουμε:

```
n = 50;
Wn = 0.4;   b = fir1(n,Wn);
```

Δημιουργούμε ένα διάνυσμα γραμμής b όπου εμπεριέχει όλους τους συντελεστές ενός φίλτρου βαθμού n με χρήση παραθύρου Hamming. Αυτό είναι ένα φίλτρο FIR διέλευσης χαμηλών , γραμμικής φάσης με συχνότητα αποκοπής Wn. Το Wn είναι ένας αριθμός που μπορεί να παίρνει τιμές από 0 έως 1, όπου το 1 αντιστοιχεί στην συχνότητα Nyquist, όπου είναι η μισή δειγματοληψίας. (Η τιμή του Wn αντιστοιχεί στο σημείο των 6dB). Για φίλτρο διέλευσης υψηλών, απλά τοποθετούμε την παράμετρο ‘high’ σαν δεδομένο εισόδου της συνάρτησης. Για φίλτρα διέλευσης ή αποκοπής ζώνης, προσδιορίζουμε την τιμή του Wn σαν διάνυσμα δύο στοιχείων περιέχοντας τις δύο συχνότητες στα άκρα της ζώνης. Για φίλτρο αποκοπής ζώνης τοποθετούμε την παράμετρο ‘stop’.

Στο παρακάτω παράδειγμα:

```
b = fir1(n,Wn,window)
```

χρησιμοποιούμε ένα παράθυρο όπου προσδιορίζεται από το διάνυσμα στήλης window για τον σχεδιασμό του φίλτρου. Το διάνυσμα window πρέπει να έχει n+1 στοιχεία. Εάν δεν προσδιορίσουμε παράθυρο η συνάρτηση fir1 χρησιμοποιεί παράθυρο Hamming.

Σχεδιασμός Φίλτρου FIR Πολλαπλών Ζωνών: fir2

Η συνάρτηση fir2 κατασκευάζει επίσης φίλτρα FIR αλλά με αυθαίρετη μορφή που όμως έχουν γραμμική απόκριση συχνότητας. Υπενθυμίζουμε ότι η συνάρτηση fir1 σχεδιάζει μόνο τυπικά φίλτρα (διέλευσης υψηλών ,χαμηλών και αποκοπής, διέλευσης ζώνης).

Στο παρακάτω παράδειγμα έχουμε:

```
n  =50;
f = [0  .4  .5 1];
m = [1 1 0 0];
b  = fir2(n,f,m);
```

Η συνάρτηση fir2 επιστρέφει ένα διάνυσμα γραμμής b συντελεστών n+1, ενός φίλτρου FIR βαθμού n, με χαρακτηριστικά πλάτους συχνότητας προσδιοριζόμενα από τα διανύσματα f και m. Το διάνυσμα f περιέχει τα σημεία συχνοτήτων όπου παίρνουν τιμές από 0 έως 1, όπου το 1 αντιπροσωπεύει την συχνότητα Nyquist. Το διάνυσμα m εμπεριέχει τις επιθυμητές τιμές πλάτους της απόκρισης στις προηγούμενες επιθυμητές συχνότητες του διανύσματος f .

Σχεδιασμός Φίλτρου Πολλαπλών Ζωνών με Ζώνες Μετάβασης

Η συνάρτηση firls και remez παρέχουν μία πιο γενική μέθοδο στον προσδιορισμό του ιδανικά επιθυμητού φίλτρου σε σχέση με τις συναρτήσεις fir1 και fir2. Αυτές οι συναρτήσεις σχεδιάζουν μετασχηματισμούς Hilbert, διαφοριστές, και άλλα φίλτρα με περιττή συμμετρία συντελεστών (τύπου III και τύπου IV γραμμικής φάσης). Μπορούμε να συμπεριλαμβάνουμε περιοχές μετάβασης όπου το σφάλμα δεν είναι ελαχιστοποιημένο, και να πραγματοποιούμε ανεξαρτήτου ζώνης βάρος της ελαχιστοποίησης.

Η συνάρτηση remez είναι μια προέκταση των συναρτήσεων fir1 και fir2 όπου ελαχιστοποιεί το σφάλμα μεταξύ της επιθυμητής απόκρισης συχνότητας και της πραγματικής καμπύλης απόκρισης.

Η συνάρτηση remez εφαρμόζει τον αλγόριθμο Parks-McClellan , όπου χρησιμοποιεί τον αλγόριθμο εκχώρησης Remez , και την προσεγγιστική θεωρία Chebyshev για τον σχεδιασμό βέλτιστων φίλτρων. Τα φίλτρα είναι βέλτιστα με την λογική ότι παρουσιάζουν ελαχιστοποίηση του μεγίστου σφάλματος μεταξύ της επιθυμητής και της πραγματικής καμπύλης απόκρισης (αυτά τα φίλτρα ονομάζονται φίλτρα minimax.) Τα φίλτρα που είναι σχεδιασμένα με αυτό τον τρόπο παρουσιάζουν όμοιας κυμάτωσης συμπεριφορά στην καμπύλη απόκρισης και για αυτό είναι γνωστά σαν φίλτρα equiripple.

Ο αλγόριθμος Parks-McClellan για τον σχεδιασμό φίλτρων FIR είναι ίσως ο πιο ευρέως χρησιμοποιούμενος και αποδεκτός τρόπος.

Η σύνταξη των συναρτήσεων firls και remez είναι όμοια.

Βασική Μορφή

Στην βασική λειτουργία τους οι συναρτήσεις firls και remez σχεδιάζουν φίλτρα γραμμικής φάσης τύπου I ή II , εξαρτώμενα από τον επιθυμητό βαθμό είναι άρτια ή περιττά αντίστοιχα.

Στο παρακάτω παράδειγμα έχουμε ένα φίλτρο διέλευσης χαμηλών με πλάτος κατά προσέγγιση 1 για συχνότητες από 0 έως 0.4 Hz, και πλάτος κατά προσέγγιση 0 από 0.5 έως 1 Hz.

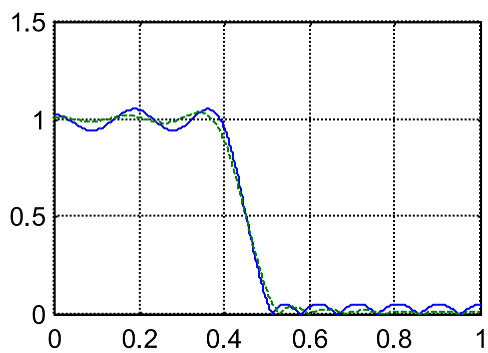
```
n =  20;              % βαθμός φίλτρου
f =[0 .4 .5 1];       % συχνότητες στα άκρα της ζώνης
a =[1 1 0 0 ];        % επιθυμητά πλάτη
b = remez(n,f,a);
```

Για την περιοχή 0.4 έως 0.5 Hz η συνάρτηση remez δεν πραγματοποιεί ελαχιστοποίηση του σφάλματος. Αυτή ονομάζεται περιοχή μετάβασης ή περιοχή μη ενδιαφέροντος . Μια περιοχή μετάβασης ελαχιστοποιεί το σφάλμα περισσότερο στις ζώνες που μας ενδιαφέρουν σε βάρος του αργότερου ρυθμού μετάβασης. Με αυτό τον τρόπο, αυτά του τύπου τα φίλτρα έχουν ένα εσωτερικό περιορισμό παρόμοιο με τον σχεδιασμό φίλτρων FIR με παράθυρα.

Ας συγκρίνουμε τον σχεδιασμό φίλτρων με την μέθοδο των ελαχίστων τετραγώνων με τον σχεδιασμό φίλτρου με όμοια κυμάτωση, χρησιμοποιώντας την συνάρτηση firls για να δημιουργήσουμε ένα παρόμοιο φίλτρο.

```
bb = firls(n,f,a);

[H,w] = freqz(b);
[HH,w] = freqz(bb);
plot(w/pi, abs(H) ,w/pi, abs(HH) ,'- -'), grid
```



Μπορούμε να παρατηρήσουμε ότι το φίλτρο σχεδιασμένο από την remez εμφανίζει συμπεριφορά όμοιας κυμάτωσης. Ακόμα η συνάρτηση firls έχει καλύτερη απόκριση σχεδόν σε όλη την διέλευση ζώνης και αποκοπής, αλλά στα άκρα της ζώνης (f=0.4 και f=0.5), η απόκριση απέχει από την ιδανική σε σχέση με την remez.

Μπορούμε να αναλογιστούμε τις ζώνες συχνοτήτων σαν γραμμές πάνω σε μικρά διαστήματα συχνοτήτων. Η συνάρτηση remez και η firls χρησιμοποιούν το παραπάνω σχήμα για να αναπαραστήσουν κάθε γραμμική περιοχή. Με τις συναρτήσεις remez και firls μπορούμε να σχεδιάσουμε φίλτρα διέλευσης χαμηλών και υψηλών καθώς και αποκοπής και διέλευσης ζώνης. Στο παρακάτω παράδειγμα έχουμε ένα φίλτρο διέλευσης ζώνης.

```
f = [0 .3 .4 .7 .8 1];   % οι άκρες της ζώνης σε ζευγάρια
a = [0  0  1 1  0  0 ]  % το πλάτος του φίλτρο διέλευσης ζώνης
```

Τα διανύσματα f και a προσδιορίζουν πέντε ζώνες.

- Δύο ζώνες αποκοπής, από 0.0 έως 0.3 και από 0.8 έως 1.0.
- Μια ζώνη αποκοπής από 0.4 έως 0.7
- Δύο ζώνες μετάβασης , από 0.3 έως 0.4 και από 0.7 έως 0.8

Ακολουθούν παραδείγματα για φίλτρα διέλευσης υψηλών και αποκοπής ζώνης:

f = [0 .7 .8 1]; % οι άκρες της ζώνης σε ζεύγη
a =[0 0 1 1] ; % το πλάτος του φίλτρου διέλευσης υψηλών

f = [0 .3 .4 .5 .8 1]; % οι άκρες της ζώνης σε ζεύγη
a =[1 1 0 0 1 1]; % το πλάτος του φίλτρου αποκοπής ζώνης

Ακολουθεί παράδειγμα φίλτρου διέλευσης ζώνης πολλαπλών ζωνών:

f = [0 .1 .15 .25 .3 .4 .45 .55 .6 .7 .75 .85 .9 1];
a =[1 1 0 0 1 1 0 0 1 1 0 0 1 1];

Άλλη δυνατότητα είναι η κατασκευή φίλτρου με γραμμική μετάβαση μεταξύ δύο ζωνών.

f = [0 .4 .42 .48 .5 1];
a=[1 1 .8 .2 0 0]; % διέλευσης ζώνης γραμμική μετάβαση αποκοπή ζώνης.

Το διάνυσμα Βάρους

Και οι δύο συναρτήσεις, η firls και η remez επιτρέπουν την ελαχιστοποίηση του σφάλματος σε κάποια ιδιαίτερα ζώνη συχνοτήτων σε σχέση με τις άλλες. Για να συμβεί αυτό, προσδιορίζουμε ένα διάνυσμα βάρους που ακολουθεί τα διανύσματα συχνοτήτων και πλάτους. Στο παρακάτω παράδειγμα έχουμε ένα φίλτρο διέλευσης χαμηλών με όμοια κυμάτωση με 10 φορές λιγότερη κυμάτωση στην ζώνη αποκοπής σε σχέση με την ζώνη διέλευσης:

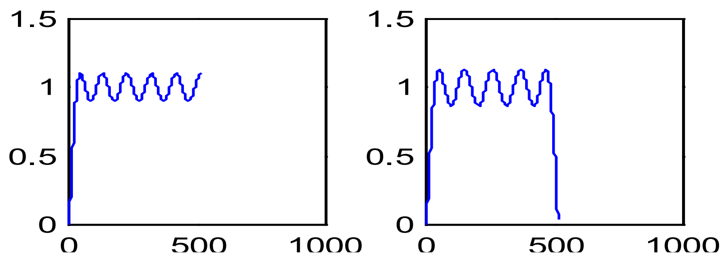
n = 20; % βαθμός του φίλτρου
f = [0 .4 .5 1]; % συχνότητες στα άκρα της ζώνης
a = [1 1 0 0]; % επιθυμητά πλάτη
w = [1 10]; % διάνυσμα βάρους
b = remez(n,f,a,w);

Ένα κατάλληλο διάνυσμα βάρους πρέπει να είναι πάντοτε το μισό του μήκους των διανυσμάτων f και a, επομένως πρέπει να έχουμε ακριβώς ένα βάρος ανά ζώνη.

Αντισυμμετρικά Φίλτρα / Μετασχηματισμοί Hilbert

Στις συναρτήσεις remez και firls μπορούμε να χρησιμοποιήσουμε μια νέα προαιρετική παράμετρο την ‘h’ ή ‘hilbert’ όπου σχεδιάζει φίλτρα FIR με περιττή συμμετρία, τύπου III (για άρτιου βαθμού) ή τύπου IV (για περιττού βαθμού) γραμμικά φίλτρα. Ένας ιδανικός μετασχηματισμός Hilbert έχει αντισυμμετρική ιδιότητα και πλάτος 1 σε όλο το εύρος συχνοτήτων. Στο παρακάτω παράδειγμα έχουμε προσεγγίσεις μετασχηματισμών Hilbert:

```
b = remez(21,[.05  1],[1  1],'h');       % διέλευσης υψηλών hilbert
bb = remez(20,[.05  .95],[1  1],'h');  % διέλευσης ζώνης hilbert
```



Μπορούμε να βρούμε τον καθυστερημένο μετασχηματισμό Hilbert ενός σήματος x όπου περνά μέσα από τέτοια φίλτρα:

```
Fs = 1000;                  % συχνότητα δειγματοληψίας
t = (0:1/Fs:2)';            % διάνυσμα χρόνου διάρκειας δύο δευτερολέπτων
x = sin(2*pi*300*t);  % ημιτονικό σήμα συχνότητας 300 κύκλων
xh = filter(bb,1,x);      % μετασχηματισμός hilbert του x
```

Το αναλυτικό σήμα που αντιστοιχεί στο x είναι ένα μιγαδικό σήμα όπου έχει το x σαν πραγματικό μέρος και τον μετασχηματισμό Hilbert του x σαν φανταστικό μέρος. Για αυτή την μέθοδο πρέπει να καθυστερήσουμε το x κατά το μισό του βαθμού του φίλτρου για να δημιουργήσουμε το αναλυτικό σήμα:

```
xd = [zeros(10,1);     x(1:length(x)-10];     % καθυστέρηση 10 δειγμάτων
xa = xd + j*xh;          % αναλυτικό σήμα
```

Η παραπάνω μέθοδος δεν λειτουργεί άμεσα για φίλτρα περιττού βαθμού όπου απαιτούν μη ακέραια καθυστέρηση. Στην προκειμένη περίπτωση μπορούμε να χρησιμοποιήσουμε την συνάρτηση resample για να καθυστερήσουμε το σήμα για μη ακέραιο αριθμό δειγμάτων.

Διαφοριστές

Το διαφορικό ενός σήματος στο πεδίο του χρόνου είναι ισοδύναμο με τον πολλαπλασιασμό του μετασχηματισμένου κατά Fourier σήμα με μια φανταστική συνάρτηση ράμπας.

Μπορούμε να προσεγγίσουμε τον ιδανικό διαφοριστή, περνώντας το σήμα μέσα από ένα φίλτρο όπου έχει απόκριση $H(\omega)=j\omega$. Κατά προσέγγιση μπορούμε να υλοποιήσουμε έναν ιδανικό διαφοριστή χρησιμοποιώντας τις συναρτήσεις remez ή firls με χρήση της προαιρετικής παραμέτρου ‘d’ ή ‘differentiator’.

Σχεδιασμός Φίλτρου FIR CLS (Constrained Least Squares)

Οι συναρτήσεις για την κατασκευή φίλτρων FIR (CLS), ενσωματώνουν τεχνικές που επιτρέπουν τον σχεδιασμό φίλτρων FIR χωρίς αυστηρή προδιαγραφή των ζωνών μετάβασης και την απόκριση πλάτους.

Η ικανότητα να παραλείπουμε προδιαγραφές των ζωνών μετάβασης είναι πολύ χρήσιμη σε αρκετές περιπτώσεις.

Αντί να προσδιορίζουμε τις ζώνες αποκοπής, διέλευσης ή μετάβασης, η μέθοδος CLS δέχεται σαν δεδομένα εισόδου την συχνότητα αποκοπής (στις περιπτώσεις φίλτρων διέλευσης υψηλών ή χαμηλών καθώς και διέλευσης ή αποκοπής ζώνης) ή τις άκρες των ζωνών διέλευσης ή αποκοπής ζώνης (για πολυζωνικό φίλτρο), για τον σχεδιασμό της επιθυμητής απόκρισης.

Ένα άλλο βασικό χαρακτηριστικό της μεθόδου CLS είναι η δυνατότητα προσδιορισμού του πάνω και κάτω ορίου της μέγιστης επιτρεπτής κυμάτωσης στην απόκριση του πλάτους. Δεδομένου αυτού του περιορισμού εφαρμόζεται η τεχνική των ελαχίστων τετραγώνων του σφάλματος ελαχιστοποίησης σε όλο το εύρος συχνοτήτων του φίλτρου και όχι μόνο στις προσδιοριζόμενες ζώνες.

Ένα επιπλέον όφελος είναι ότι αυτή η τεχνική επιτρέπει τον προσδιορισμό μικρών κορυφώσεων που είναι αποτέλεσμα του φαινομένου Gibb's.

Υπάρχουν δύο συναρτήσεις στο SPT που ενσωματώνουν την παραπάνω σχεδιαστική τεχνική.

Περιγραφή	Συνάρτηση
Σχεδιασμός φίλτρου FIR πολλαπλών ζωνών (CLS)	fircls
Σχεδιασμός φίλτρων διέλευσης χαμηλών και υψηλών (CLS) γραμμικής φάσης	fircls1

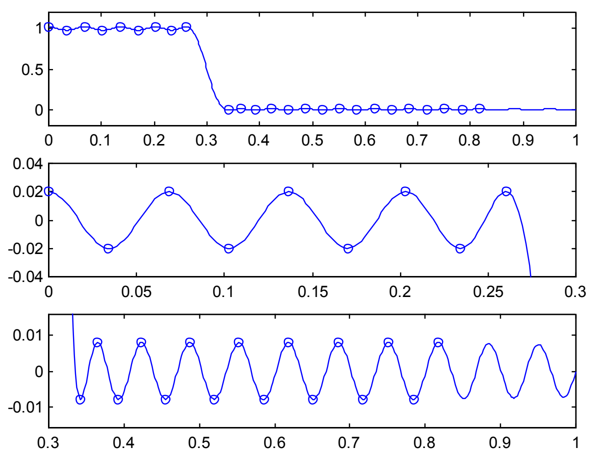
Βασική Σχεδίαση Φίλτρου CLS Διέλευσης Χαμηλών και Υψηλών Συχνοτήτων

Η πιο βασική συνάρτηση στον σχεδιασμό φίλτρων CLS είναι η fircls1, και χρησιμοποιείται για τον σχεδιασμό φίλτρων FIR διέλευσης χαμηλών και υψηλών συχνοτήτων. Στο παρακάτω παράδειγμα σχεδιάζουμε φίλτρο με βαθμό κρουστικής απόκρισης 61 και κανονικοποιημένη συχνότητα 0.3. Ακόμα διευκρινίζουμε τις κυματώσεις στις ζώνες διέλευσης και αποκοπής.

- Μέγιστη απόκλιση στην ζώνη διέλευσης από το 1 (κυμάτωση στην ζώνη διέλευσης) 0.02
- Μέγιστη απόκλιση στην ζώνη αποκοπής από το 0 (κυμάτωση στην ζώνη αποκοπής) 0.008

Χρησιμοποιώντας την συνάρτηση fircls1 θα έχουμε :

```
n = 61;
wo = 0 .3;
dp = 0 .02;
ds = 0.008;
h = fircls1(n,wo,dp,ds,'plot'),
```



Σχεδιασμός Φίλτρου CLS Πολλαπλών Ζωνών

Η συνάρτηση fircls χρησιμοποιεί την ίδια τεχνική όπως η προηγούμενων συνάρτηση (fircls1) για τον σχεδιασμό φίλτρων FIR Σε αυτή την περίπτωση προσδιορίζουμε ένα διάνυσμα για τις συχνότητες που βρίσκονται στα άκρα των ζωνών και αντίστοιχα ακόμα ένα διάνυσμα για τον προσδιορισμό των πλατών. Επιπρόσθετα μπορούμε να προσδιορίσουμε το μέγιστο ποσό κυμάτωσης για κάθε ζώνη.

Θα υποθέσουμε ότι χρειαζόμαστε ένα φίλτρο με τις παρακάτω προδιαγραφές:

- Από 0 έως 0.3 : πλάτος 0, πάνω όριο 0.005, κάτω όριο –0.005
- Από 0.3 έως 0.5 : πλάτος 0.5, πάνω όριο 0.51, κάτω όριο 0.49
- Από 0.5 έως 0.7 : πλάτος 0, πάνω όριο 0.03, κάτω όριο –0.03
- Από 0.7 έως 0.9 : πλάτος 1, πάνω όριο 1.02, κάτω όριο 0.98
- Από 0.9 έως 1 : πλάτος 0, πάνω όριο 0.05, κάτω όριο –0.05

Θα σχεδιάσουμε ένα φίλτρο CLS με βαθμό κρουστικής απόκρισης 129 όπου ικανοποιεί τις προηγούμενες προδιαγραφές:

```
n = 129;
f = [0  0.3  0.5  0.7  0.9  1];
a = [ 0  0.5 0 1 0];
up = [0.005  0.51  0.03  1.02  0.05];
lo = [-0.005  0.49 –0.03  0.98  -0.05];
h = fircls(n,f,a,up,lo,'plot');
```

Σχεδιασμός Φίλτρου CLS Weighted

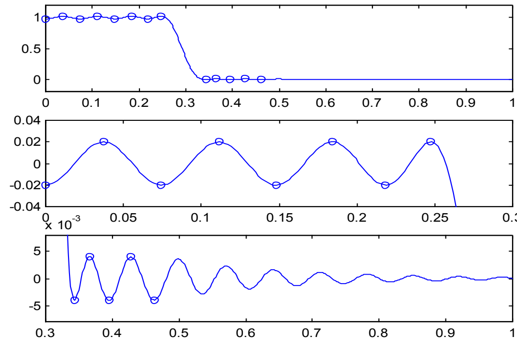
Ο σχεδιασμός φίλτρων CLS με βάρος μας επιτρέπει να σχεδιάζουμε φίλτρα FIR διέλευσης χαμηλών ή υψηλών με διαφορετικό σχετιζόμενο βάρος του σφάλματος ελαχιστοποίησης σε κάθε ζώνη. Η συνάρτηση fircls1 επιτρέπει τον προσδιορισμό των άκρων των ζωνών διέλευσης και αποκοπής συναρτήσει του βάρους των ελαχίστων τετραγώνων , αλλά επίσης και με σταθερό k όπου προσδιορίζει τον λόγο του βάρους της ζώνης διέλευσης με την αποκοπής.

Στο παρακάτω παράδειγμα έχουμε ένα φίλτρο FIR με βαθμό κρουστικής απόκρισης 55 και συχνότητα αποκοπής 0.3. Ακόμα υποθέτουμε μέγιστη επιτρεπτή κυμάτωση στην ζώνη διέλευσης 0.02 και στην ζώνη αποκοπής 0.004. Επιπρόσθετα έχουμε προσθέσει απαιτήσεις βάρους.

- Στα άκρα της ζώνης διέλευσης για την συνάρτηση βάρους έχουμε 0.28.
- Στα άκρα της ζώνης αποκοπής για την συνάρτηση βάρους έχουμε 0.32.
- Το βάρος του σφάλματος ελαχιστοποίησης θα είναι 10 φορές πιο μεγάλο στην ζώνη αποκοπής σε σχέση με την ζώνη διέλευσης.

Για να μπορέσουμε να ικανοποιήσουμε την παραπάνω προδιαγραφή θα χρησιμοποιήσουμε την συνάρτηση fircls1.

```
n = 55;
   wo = 0.3;
   dp = 0.02;
   ds = 0.004;
   wp = 0.28;
   ws = 0.32;
   k =10;
   h = fircls1(n,wo,dp,ds,wp,ws,k,'plot');
```



Σχεδιασμός Φίλτρου Αυθαίρετης Απόκρισης

Ένα βασικό εργαλείο για την κατασκευή φίλτρων είναι η συνάρτηση cremez όπου μπορεί να σχεδιάζει φίλτρα FIR με αυθαίρετη πολύπλοκη απόκριση. Διαφοροποιείται από τις άλλες συναρτήσεις σε σχέση με το πώς καθορίζει την καμπύλη απόκρισης του φίλτρου. Αυτό γίνεται περνώντας το όνομα μια συνάρτησης στην cremez, όπου επιστρέφει την υπολογισμένη απόκριση του φίλτρου σε ένα πλέγμα συχνοτήτων. Αυτή η δυνατότητα του φίλτρου κάνει την cremez πολύπλευρη και πολύ δυνατή στον σχεδιασμό φίλτρων.

Αυτή η σχεδιαστική τεχνική μπορεί να χρησιμοποιηθεί για την κατασκευή φίλτρων μη γραμμικής φάσης, με ασύμμετρη καμπύλη απόκρισης (με μιγαδικούς συντελεστές), ή συμμετρικά φίλτρα με ειδική απόκριση συχνότητας.

Ο αλγόριθμος σχεδίασης βελτιστοποιεί το σφάλμα Chebyshev χρησιμοποιώντας εκτεταμένα τον αλγόριθμο Remez για αρχικό προσδιορισμό.

Σχεδιασμός Φίλτρου Πολλαπλών Ζωνών

Θα κατασκευάσουμε ένα φίλτρο πολλαπλών ζωνών με την ακόλουθα χαρακτηριστικά στο πεδίο των συχνοτήτων.

Ζώνες	Πλάτος	Βελτιστοποιημένο Βάρος
[-1 -0.5]	[5 1]	1
[-0.4 +0.3]	[2 2]	10
[+0.4 +0.8]	[2 1]	1

Ένα φίλτρο γραμμικής φάσης πολλαπλών ζωνών μπορεί να σχεδιαστεί χρησιμοποιώντας την προκαθοριζόμενη συνάρτηση απόκρισης συχνότητας multiband, ως ακολούθως:

```
b = cremez(38,  [-1  -0.5  -0.4  0.3  0.4  0.8], ...
                {'multiband', [5 1 2 2 2 1]}, [1  10  5]);
```

Για ειδικές περιπτώσεις στα φίλτρα πολλαπλών ζωνών, μπορεί να χρησιμοποιηθεί μια συντόμευση στην παράσταση της συνάρτησης που είναι όμοια με την συνάρτηση remez:

b = cremez(38, [-1 -0.5 -0.4 0.3 0.4 0.8], ...
[5 1 2 2 2 1]}, [1 10 5]);

Όπως με την remez, έτσι και στην cremez περνάει ένα διάνυσμα στην συνάρτηση με τις άκρες της ζώνης. Αυτό το διάνυσμα προσδιορίζει τις ζώνες συχνοτήτων όπου θα πραγματοποιηθεί η βελτιστοποίηση. Υπενθυμίζουμε ότι στο προηγούμενο παράδειγμα έχουμε δύο ζώνες μετάβασης, από –0.5 έως –0.4 και από 0.3 έως 0.4.

Μπορούμε να απεικονίσουμε την καμπύλη απόκρισης του φίλτρου σε γραμμική κλίμακα.

[h,w] = freqz(b,1,512,'whole');
plot(w/pi-1,fftshift(abs(h)));

Υπενθυμίζουμε ότι η καμπύλη απόκρισης υπολογίζεται για όλο το εύρος της κανονικοποιημένης συχνότητας [-1 +1], περνώντας την παράμετρο 'whole' στην συνάρτηση freqz.

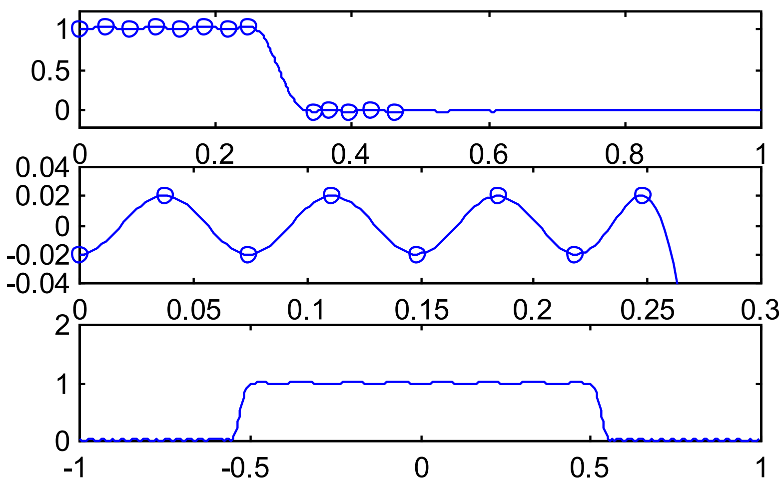
Σχεδιασμός Φίλτρου με Περιορισμένη Καθυστέρηση

Έστω ο σχεδιασμός φίλτρου διέλευσης χαμηλών μήκους 62 με συχνότητα αποκοπής την μισή συχνότητα Nyquist. Εάν τοποθετήσουμε αρνητική τιμή αντιστάθμισης στην συνάρτηση σχεδιασμού του φίλτρου διέλευσης χαμηλών, τότε το συνολικό καθυστερούμενο αντιστάθμισμα για την σχεδίαση είναι σημαντικά λιγότερο σε σχέση με αυτό που θα πετυχαίναμε σε κλασική σχεδίαση γραμμικής φάσης. Αυτό μπορεί να υπολογιστεί ως ακολούθως:

b = cremez(61, [0 0.5 0.55 1], {'lowpass', -16});

Η απόκριση του πλάτους είναι:

[h,w] = freqz(b,1,512,'whole');
plot(w/pi-1,fftshift(abs(h)));



Η συνολική καθυστέρηση του φίλτρου φανερώνει ότι το αντιστάθμισμα έχει μειωθεί από N/2=30.5 σε N/2-16=14.5. Ως εκ τούτου η συνολική καθυστέρηση δεν είναι πλέον επίπεδη στην ζώνη διέλευσης. Εάν συγκρίνουμε αυτό το μη γραμμικό φασικό φίλτρο με το γραμμικό φίλτρο όπου έχει ακριβώς 14.5 δειγμάτων συνολική καθυστέρηση, το τελικό φίλτρο είναι βαθμού 2*14.5=29.

Χρησιμοποιώντας την συνάρτηση b=cremez(29, [0 0.5 0.55 1], 'lowpass'), η κυμάτωση στην ζώνη αποκοπής και διέλευσης είναι πολύ μεγαλύτερη για φίλτρο 29 βαθμού. Αυτές οι συγκρίσεις μπορούν να βοηθήσουν στην επιλογή του φίλτρου που είναι πιο κατάλληλο για μια ειδική εφαρμογή.

B.3 Παράθυρα

B.3.1 Παράθυρα

Και στις δύο περιπτώσεις δηλαδή στον σχεδιασμό φίλτρων και στον προσδιορισμό του φάσματος ισχύος, η επιλογή του παραθύρου παίζει ένα πολύ σημαντικό ρόλο στον προσδιορισμό της ποιότητας των τελικών αποτελεσμάτων. Ο κύριος ρόλος του παραθύρου είναι να μειώσει τα φαινόμενα Gibb's όπου είναι αποτέλεσμα του περιορισμού μιας άπειρης σειράς.

Το SPT παρέχει τις παρακάτω συναρτήσεις για την κατασκευή παραθύρων.

Παράθυρο	Συνάρτηση
Παράθυρο Bartlett	bartlett
Παράθυρο Blackman	blackman
Ορθογώνιο Παράθυρο	boxcar
Παράθυρο Chebyshev	chebwin
Παράθυρο Hamming	hamming
Παράθυρο Hanning	hanning
Παράθυρο Kaiser	kaiser
Τριγωνικό Παράθυρο	triang

Βασικά Σχήματα

Το πιο βασικό παράθυρο είναι το, ορθογώνιο παράθυρο, είναι ένα διάνυσμα από 1 με κατάλληλο μήκος. Παρακάτω έχουμε ένα ορθογώνιο παράθυρο μήκους 50:

```
n=50;
w= boxcar(n);
```

Το SPT αποθηκεύει το παράθυρο σε διανύσματα στηλών κατά σύμβαση, επομένως μια ισοδύναμη έκφραση θα είναι η ακόλουθη:

```
w=ones(50,1);
```

Το παράθυρο Bartlett (ή τριγωνικό) είναι η συνέλιξη δύο ορθογώνιων παραθύρων. Οι συναρτήσεις bartlett και triang υπολογίζουν παρόμοια τριγωνικά παράθυρα, με τρεις σημαντικές διαφορές. Η συνάρτηση bartlett πάντοτε επιστρέφει ένα παράθυρο με δύο μηδενικά στις άκρες της ακολουθίας, επομένως για περιττό n , το κεντρικό τμήμα της bartlett (n+2) θα είναι ισοδύναμο με το triang(n):

```
bartlett(7)
ans=
         0
    0.3333
    0.6667
    1.0000
    0.6667
    0.3333
         0
```

```
triang(5)
ans=
    0.3333
    0.6667
    1.0000
    0.6667
    0.3333
```

Για n άρτιο, η συνάρτηση bartlett εξακολουθεί και είναι η συνέλιξη δύο ορθογώνιων ακολουθιών. Δεν υπάρχει κοινώς αποδεκτή διευκρίνηση για τριγωνικό παράθυρο με n άρτιο.

```
w= bartlett(8);
[w(2:7)   triang(6)]
ans=
    0.2857        0.1667
    0.5714        0.5000
    0.8571        0.8333
    0.8571        0.8333
    0.5714        0.5000
    0.2857        0.1667
```

Η τελική διαφορά μεταξύ των τριγωνικών και Bartlett παραθύρων είναι ο μετασχηματισμός Fourier των δύο συναρτήσεων. Ο μετασχηματισμός Fourier το παραθύρου Bartlett είναι αρνητικός για n περιττό. Ο μετασχηματισμός Fourier του τριγωνικού παραθύρου είναι πάντοτε μη αρνητικός.

Αυτή η διαφορά μπορεί να είναι σημαντική όταν διαλέγουμε παράθυρο για ορισμένες τεχνικές προσδιορισμού του φάσματος, όπως η μέθοδος Blackman – Tykey.

Γενικευμένα Παράθυρα Συνημίτονων

Τα παράθυρα, Blackman, Hamming, Hanning, και τα ορθογώνια ανήκουν σε ειδική κατηγορία του γενικευμένου συνημιτονικού παραθύρου. Αυτά τα παράθυρα είναι συνδυασμοί ημιτονοειδών ακολουθιών με συχνότητες 0, 2π/(N-1), και 4π/(N-1), όπου το N είναι το μήκος του παραθύρου. Ένας τρόπος για να παράγουμε αυτό είναι:

```
ind = (0:n-1)' 2*pi/(n-1);
w = A-B*cos(ind) + C*cos(2*ind);
```

όπου τα A, B και C είναι προσδιοριζόμενες σταθερές.

Τα παράθυρα Hamming και Hanning είναι δύο όρων γενικευμένα παράθυρα συνημιτόνου για A=0.54 , B=0.46, για Hamming και A=0.5 και B=0.5 για Hanning (το C είναι και στις δύο περιπτώσεις μηδέν). Οι συναρτήσεις hamming και hanning υπολογίζουν τα αντίστοιχα παράθυρα.

Πρέπει να διευκρινίσουμε ότι το γενικευμένο παράθυρο συνημίτονου παράγει μηδενικά στα δείγματα 1 και n για A=0.5 και B=0.5. Για να αποκλείσουμε τα μηδενικά αυτά στις άκρες του παραθύρου, η συνάρτηση hanning χρησιμοποιεί ένα συνημίτονο με συχνότητα 2π/(N+1) και όχι το 2π/(N-1).

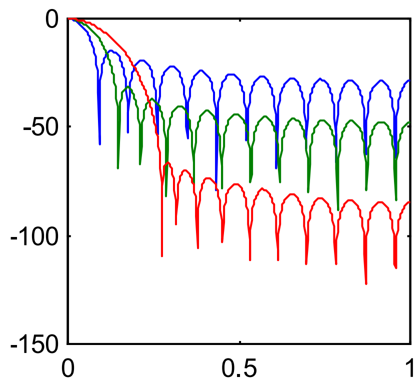
Το παράθυρο blackman είναι ένα δημοφιλές παράθυρο τριών όρων με A=0.42, B=0.5, C=0.08. Η συνάρτηση blackman υπολογίζει αυτό το παράθυρο.

Παράθυρο Kaiser

Το παράθυρο Kaiser είναι μία προσέγγιση του επιμηκυμένου σφαιρικού παραθύρου, όπου ο λόγος ενέργειας του κυρίου ως προς του πλευρικού λοβού είναι μεγιστοποιημένος. Για το παράθυρο Kaiser το μήκος του είναι συγκεκριμένο, η παράμετρος β ελέγχει το ύψος του πλευρικού λοβού. Για δεδομένο β , το ύψος του πλευρικού λοβού είναι σταθερό συναρτήσει του μήκους του παραθύρου. Η δήλωση kaiser(n,beta) υπολογίζει ένα παράθυρο kaiser μήκους n με παράμετρο β.

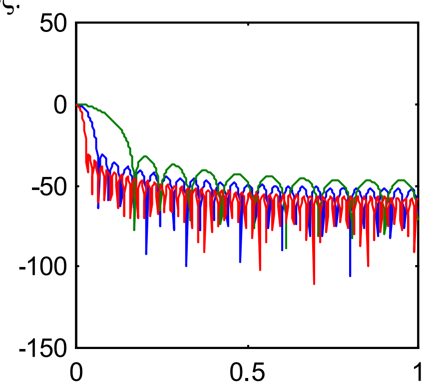
Στο παρακάτω παράδειγμα έχουμε ένα παράθυρο Kaiser μήκους 50 με διαφορετικές τιμές της παραμέτρου β.

```
N = 50;
w1 = kaiser(n,1);
w2 = kaiser(n,4);
w3 = kaiser(n,9);
[W1,f] = freqz(w1/sum(w1), 1, 512, 2);
[W2,f] = freqz(w2/sum(w2), 1, 512, 2);
[W3,f] = freqz(w3/sum(w3), 1, 512, 2);
plot(f, 20*log10(abs([W1  W2  W3])))
```



Όσο το β αυξάνει, τόσο το ύψος του πλευρικού λοβού μειώνεται, και το εύρος του κυρίου λοβού αυξάνει. Στο παρακάτω παράδειγμα βλέπουμε πως το ύψος του πλευρικού λοβού παραμένει σταθερό για σταθερό β , και μεταβάλλοντας το μήκος.

```
w1 = kaiser(50,4);
w2 = kaiser(20,4);
w3 = kaiser(101,4);
[W1,f] = freqz(w1/sum(w1), 1, 512, 2);
[W2,f] = freqz(w2/sum(w2), 1, 512, 2);
[W3,f] = freqz(w3/sum(w3), 1, 512, 2);
plot(f, 20*log10(abs([W1  W2  W3])))
```



Παράθυρα Kaiser στον Σχεδιασμό Φίλτρων FIR

Υπάρχουν δύο σχεδιαστικά υποδείγματα που μπορούν να μας βοηθήσουν στον σχεδιασμό φίλτρων FIR και ταυτόχρονα να ικανοποιούν προδιαγραφές με χρησιμοποίηση παραθύρου Kaiser. Για να πετύχουμε ύψος πλευρικού λοβού –α dB, η παράμετρος β πρέπει να είναι:

$$B = \begin{cases} 0.1102(\alpha-8.7), & \alpha>50 \\ 0.5842(a-21)^{0.4}+0.07886(a-21), & 50 \geq a \geq 21 \\ 0, & \alpha<21 \end{cases}$$

και για εύρος μετάβασης Δω rad/sec , χρησιμοποιούμε το παρακάτω μήκος.

$$n = \frac{a-8}{2.285\Delta\omega} + 1$$

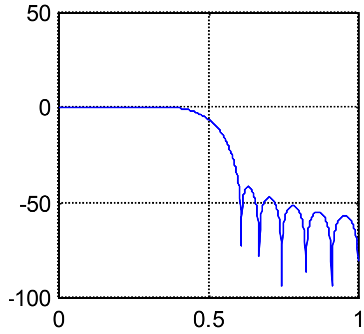
Η κατασκευή φίλτρων με αυτές τις μεθόδους αντιμετωπίζουν τις προδιαγραφές προσεγγιστικά και για αυτό πρέπει να το ελέγχουμε. Για τον σχεδιασμό ενός φίλτρου χαμηλών συχνοτήτων με συχνότητα αποκοπής 0.5 rad/sec, εύρος μετάβασης 0.2 rad/sec, και εξασθένιση 40 dB στην ζώνη αποκοπής:

```
[n,wn,beta] = kaiserord([0.4  0.6]*pi, [1 0], [0.01  0.01], 2*pi);
h = fir1(n, wn, kaiser(n+1, beta), ‘nonscale’);
```

Η συνάρτηση kaiserord προσδιορίζει τον βαθμό του φίλτρου, την συχνότητα αποκοπής, και την παράμετρο β του παραθύρου kaiser που χρειάζεται για τις δεδομένες προδιαγραφές στο πεδίο των συχνοτήτων.

Η κυμάτωση στην ζώνη διέλευσης είναι χονδρικά ίδια με την κυμάτωση στην ζώνη αποκοπής. Όπως μπορούμε να δούμε από την καμπύλη απόκρισης, το φίλτρο περίπου ικανοποιεί τις προδιαγραφές:

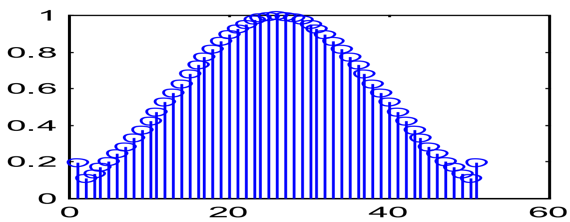
[H,f] = freqz(h, 1, 512, 2);
*plot(f, 20*log10(abs(H))),grid*



Παράθυρο Chebychev

Το παράθυρο Chebyshev ελαχιστοποιεί το εύρος του κυρίου παραθύρου, για δεδομένο συγκεκριμένο ύψος πλαγίου λοβού. Χαρακτηρίζεται από συμπεριφορά ίδιας κυμάτωσης και επομένως όλοι οι πλευρικοί λοβοί έχουν το ίδιο ύψος. Η συνάρτηση chebwin με δεδομένα το μήκος και τις παραμέτρους ύψους του πλευρικού λοβού , υπολογίζει το παράθυρο chebyshev.

n = 51;
Rs = 40;   %
w = chebwin(n,Rs);
stem(w)



Το παράθυρο Chebyshev προσδιορίζεται μόνο για περιττά μήκη. Εάν το n δεν είναι περιττό η συνάρτηση chebwin το αυξάνει κατά ένα σχεδιάζοντας το παράθυρο Chebyshev μήκους n+1.

Σχεδιάζουμε την καμπύλη απόκρισης για ίδια κυμάτωση στα –40 dB χρησιμοποιώντας τις εντολές:

[W,f] = freqz(w, 1, 512, 2);
*plot(f, 20*log10(abs(W)/sum(w))), grid*