Τμήμα Μηχανικών Η/Υ και Πληροφορικής
Εργαστήριο Επεξεργασίας Σημάτων & Τηλεπικοινωνιών
Τομέας Υλικού και Αρχιτεκτονικής Υπολογιστών
Πανεπιστήμιο Πατρών

Εργαστήριο Ψηφιακής Επεξεργασίας Σημάτων

Σετ Βοηθητικών Ασκήσεων #4

(Στοχαστικές Διαδικασίες & Προσαρμοστικά Φίλτρα)

Επιβλέπων Καθηγητής:
Δημήτριος Κοσμόπουλος

Σύνταξη – Επιμέλεια:
Αλέξανδρος-Οδυσσέας Φαρμάκης

Πάτρα, Εαρινό Εξάμηνο, 2022-23

Θέματα: Εισαγωγή στη MATLAB για ΨΕΣ

Θέμα 1ο) Αξιολόγηση Στοχαστικών Διαδικασιών

Η στοχαστική επεξεργασία σημάτων είναι απαραίτητη λόγω της εγγενούς αβεβαιότητας και μεταβλητότητας στα σήματα του πραγματικού κόσμου. Παρέχει ένα μαθηματικό πλαίσιο για τη μοντελοποίηση και την ανάλυση σημάτων που επηρεάζονται από τυχαίες διεργασίες, όπως ο θόρυβος και οι παρεμβολές, το οποίο μας δίνει τη δυνατότητα να κατανοήσουμε και να χαρακτηρίσουμε τη συμπεριφορά του σήματος σε μη ντετερμινιστικές συνθήκες. Συνεπώς, στην άσκηση αυτή σας δίνεται το ακόλουθο σήμα:

$$X(n,\vartheta)=A(\vartheta)+y(n), \qquad y(n)=\sin(2\cdot pi\cdot n)$$

όπου $A(\vartheta)$ τυχαία μεταβλητή η οποία προέρχεται από μια κατανομή Γάμμα άγνωστης μέσης τιμής και διασποράς, η οποία δίνεται στο αρχείο *gamma_distribution.mat*.

α) Είναι το σήμα που περιγράφεται παραπάνω ντετερμινιστικό ή στοχαστικό;
β) Υπολογίστε τη μέση τιμή του ντετερμινιστικού μέρους του σήματος.
γ) Χρησιμοποιώντας το Νόμο των Μεγάλων αριθμών, υπολογίστε τη στοχαστική μέση τιμή και τη διασπορά του σήματος.
δ) Να υπολογίσετε τη μέση τιμή και τη διασπορά της κατανομής Γάμμα που σας δόθηκε.
ε) Να εκτιμηθεί η ακολουθία αυτοσυσχέτισης. Τι παρατηρείτε καθώς αυξάνει ο αριθμός *K* των υλοποιήσεων της διαδικασίας που χρησιμοποιούνται στην εκτίμηση της ακολουθίας αυτοσυσχέτισης;
στ) Είναι η παραπάνω διαδικασία “λευκή”;
ζ) Να υπολογίσετε την Πυκνότητα Φάσματος (Spectral Density) της διαδικασίας. Πόσο κοντά στην ιδανική πυκνότητα είναι η εκτίμησή της από την ακολουθία αυτοσυσχέτισης του Ερωτήματος δ και πώς επηρεάζεται από το *K*;

Λύση:

α) Το παραπάνω σήμα αποτελείται από δύο (2) ξεχωριστά μέρη: το πρώτο είναι μια τυχαία μεταβλητή που προέρχεται από κατανομή Γάμμα και το δεύτερο το οποίο αποτελεί ντετερμινιστικό σήμα, το οποίο είναι το $y(n)$. Συνεπώς, επειδή εμπλέκεται στη συνάρτηση μας τυχαία μεταβλητή, το σήμα είναι στοχαστικό.

β) Αφού γνωρίζουμε ότι το σήμα μας είναι ημίτονο, ως γνωστόν ισχύει ότι η μέση τιμή του κατά την περίοδο του είναι μηδέν. Οπότε:

$$\mu(y)=\int_0^1 y(n)\,dn=\int_0^1 \sin(2\cdot pi\cdot n)\,dn=0$$

γ) Λόγω του Νόμου των Μεγάλων Αριθμών, ο μέσος όρος των αποτελεσμάτων που λαμβάνονται από μεγάλο αριθμό δοκιμών θα πρέπει να είναι κοντά στην αναμενόμενη τιμή και τείνει στην αναμενόμενη τιμή καθώς οι δοκιμές τείνουν στο άπειρο. Συνεπώς, ισχύουν οι ακόλουθες σχέσεις:

$$\mu=\frac{1}{n}\sum_{i=1}^{n}X_i=\frac{1}{n}(X_1+...+X_n) \quad \text{και} \quad Var(\mu)=Var\left(\frac{1}{n}(X_1+...+X_n)\right)=\frac{1}{n^2}Var((X_1+...+X_n))=\frac{n\cdot\sigma^2}{n^2}=\frac{\sigma^2}{n}$$

Οπότε, όσες παραπάνω υλοποιήσεις πάρουμε του στοχαστικού σήματός μας, τόσο παραπάνω θα τείνει η αριθμητική μέση τιμή στη στοχαστική μέση τιμή, και αντίστοιχα για την αριθμητική διασπορά και στοχαστική διασπορά. Μέσω των υλοποιήσεων, βγάζουμε ότι η στοχαστική μέση τιμή είναι 1 και η διασπορά είναι 0,25.

δ) Αφού βρήκαμε ότι η μέση τιμή του ντετερμινιστικού μέρους του σήματός μας είναι 0, και εκ φύσεως δεν έχει διασπορά, τότε οι τιμές που θα βγάλουμε είναι ίδιες και για τη κατανομή. Άρα, η κατανομή Γάμμα που δόθηκε έχει μέση τιμή 1 και διασπορά 0,25, τα οποία αντιστοιχούν σε συντελεστές α = 4 και β = 4 που καθορίζουν το σχήμα της κατανομής Γάμμα.

ε) Για να υπολογίσουμε την αυτοσυσχέτιση της συνάρτησης $X(n, \vartheta)$, πρέπει να υπολογίσουμε τη συσχέτιση μεταξύ της συνάρτησης σε διαφορετικές χρονικές καθυστερήσεις, ας πούμε k. Η αυτοσυσχέτιση μετρά την ομοιότητα μεταξύ ενός σήματος και μιας καθυστερημένης εκδοχής του εαυτού του. Συνεπώς:

$$R_X(k)=E\{(X(n,\vartheta)-\mu)(X(n\pm k,\vartheta)-\mu)\}\Leftrightarrow R_x(k)=E\{(X(n,\vartheta)-1)(X(n\pm k,\vartheta)-1)\}$$

Πρέπει να υπολογίσουμε την αναμενόμενη τιμή του γινομένου εντός της συνάρτησης προσδοκίας E για όλες τις τιμές του n. Στην περίπτωσή μας, $X(n, \vartheta) = A(\vartheta) + y(n)$, όπου $y(n)$ το ημίτονο και $A(\vartheta)$ η τυχαία μεταβλητή από τη κατανομή Γάμμα που είχαμε υπολογίσει τη μέση τιμή και διασπορά της προηγουμένως, και δεδομένου ότι η συνάρτηση $y(n)$ είναι μια ντετερμινιστική συνάρτηση του n και η $A(\vartheta)$ είναι ανεξάρτητη από την $y(n)$, η αναμενόμενη τιμή του αθροίσματός τους είναι το άθροισμα των αναμενόμενων τιμών τους, οπότε έχουμε:

$$E\{X(n,\vartheta)\}=E\{A(\vartheta)+y(n)\}=E\{A(\vartheta)\}+E\{y(n)\}$$

Αφού η $A(\vartheta)$ είναι τυχαία μεταβλητή από κατανομή με μέσο όρο 1, $E\{A(\vartheta)\} = 1$. Επιπλέον, αφού η συνάρτηση $y(n)$ είναι περιοδική με μέση τιμή 0 όπως υπολογίσαμε νωρίτερα, η αναμενόμενη τιμή της σε μία περίοδο είναι μηδέν. Επομένως, $E\{y(n)\} = 0$, άρα έχομε τελικά $E\{X(n, \vartheta)\} = 1 + 0 = 1$. Απλοποιώντας την εξίσωση της αυτοσυσχέτισης, προκύπτει:

$$R_X(k)=E\{X(n,\vartheta)\cdot X(n\pm k,\vartheta)\}-1$$

Τώρα, πρέπει να υπολογίσουμε τη τιμή του νέου γινομένου που σχηματίστηκε εντός της συνάρτησης προσδοκίας E για όλες τις τιμές του n. Επεκτείνοντας την έκφραση, έχουμε:

$$E\{X(n,\vartheta)\cdot X(n\pm k,\vartheta)\}=E\{A(\vartheta)A(\vartheta)\}+E\{A(\vartheta)y(n\pm k)\}+E\{y(n)A(\vartheta)\}+E\{y(n)y(n\pm k)\}$$

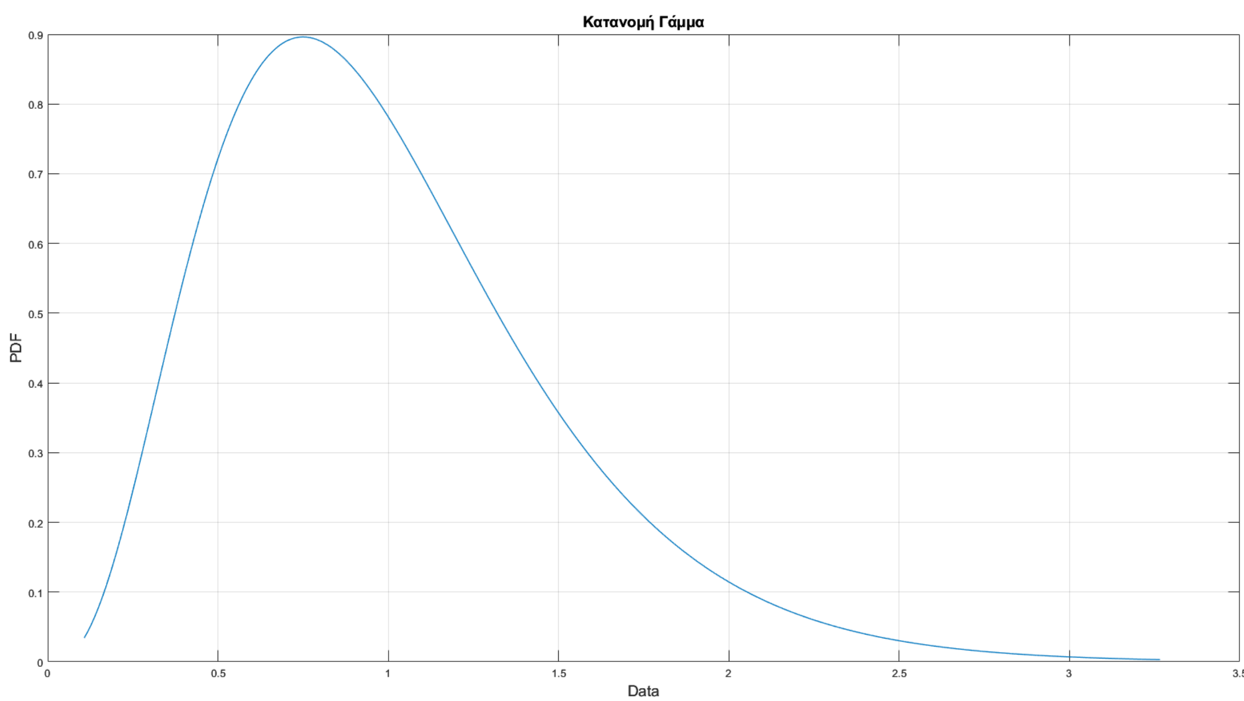
Η αναμενόμενη τιμή του $A(\vartheta)A(\vartheta)$ είναι η διασπορά του $A(\vartheta)$ συν το τετράγωνο του μέσου όρου του, δηλαδή $E\{A(\vartheta)A(\vartheta)\}$ = Var[$A(\vartheta)$] + $E\{A(\vartheta)\}^2$ = 0,25 + 1^2 = 1,25. Μπορούμε επίσης να υπολογίσουμε τη τιμή του $E\{y(n)y(n\pm k)\}$ ως:

$$R_y(k)=E\{y(n)y(n\pm k)\}=E\{\sin(2\cdot\pi\cdot n)\cdot\sin(2\cdot\pi\cdot(n\pm k))\}=\frac{1}{2}\cdot\cos(2\cdot\pi\cdot k)$$

Τέλος, μπορούμε να υπολογίσουμε τα $E\{y(n\pm k)A(\vartheta)\}$ και $E\{A(\vartheta)y(n)\}$ ως γινόμενα όπου οι όροι είναι ανεξάρτητοι μεταξύ τους, οπότε τελικά προκύπτει ότι:

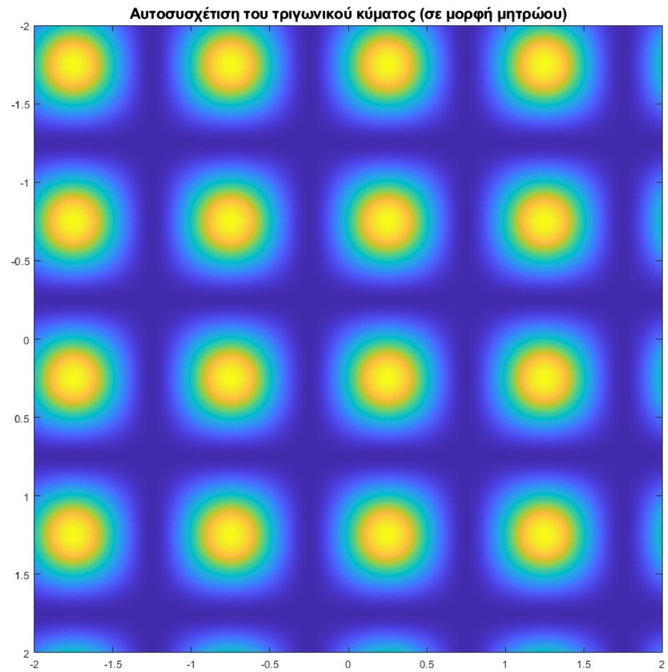
$$R_X(k)=E\{X(n,\vartheta)\cdot X(n\pm k,\vartheta)\}-1\Leftrightarrow R_X(k)=1{,}25+0+0+\frac{1}{2}\cdot\cos(2\cdot\pi\cdot k)-1\Leftrightarrow R_X(k)=\frac{1}{2}\cdot\cos(2\cdot\pi\cdot k)+0{,}25$$

Η εκτίμηση της αυτοσυσχέτισης που περιμένουμε να δούμε είναι το τετράγωνο του αθροίσματος της μέσης τιμής της κατανομής μας και της μέγιστης τιμής του ντετερμινιστικού μέρους, δηλαδή $(1 + 1)^2 = 4$. Όσο αυξάνεται το πλήθος των υλοποιήσεων, βλέπουμε ότι η τιμή που παρατηρούμε τείνει σε αυτό το νούμερο, βέβαια πρέπει να έχουμε αρκετές υλοποιήσεις, μιας και η κατανομή Γάμμα δεν είναι συμμετρική, οπότε το πλήθος των "μεγάλων" αριθμών είναι μεγαλύτερο από το πλήθος των "μικρών" αριθμών. Παρακάτω φαίνεται ενδεικτικά το γράφημα της κατανομής Γάμμα:



Στον κώδικα στο τέλος της άσκησης υπάρχουν και οι αντίστοιχοι υπολογισμοί.

στ) Το μητρώο που προκύπτει στο τέλος, αφού το εξετάσουμε, δεν είναι Toeplitz, οπότε η διαδικασία δεν είναι “λευκή”. Αντίστοιχη δικαιολόγηση είναι ότι η αυτοσυσχέτιση που υπολογίσαμε δεν είναι της μορφής $R_X(k)\cdot\delta(n-(n\pm k))$. Τυπώνοντας το μητρώο αυτοσυσχέτισης, βλέπουμε για K = 1000:



ζ) Για να βγάλουμε τη Πυκνότητα Φάσματος του σήματος, αρκεί να πάρουμε το μετασχηματισμό Fourier της αυτοσυσχέτισής του. Αξιοποιώντας τις απαραίτητες ιδιότητες, καταλήγουμε σε μια εξίσωση για τη Πυκνότητα Φάσματος που περιγράφεται ως:

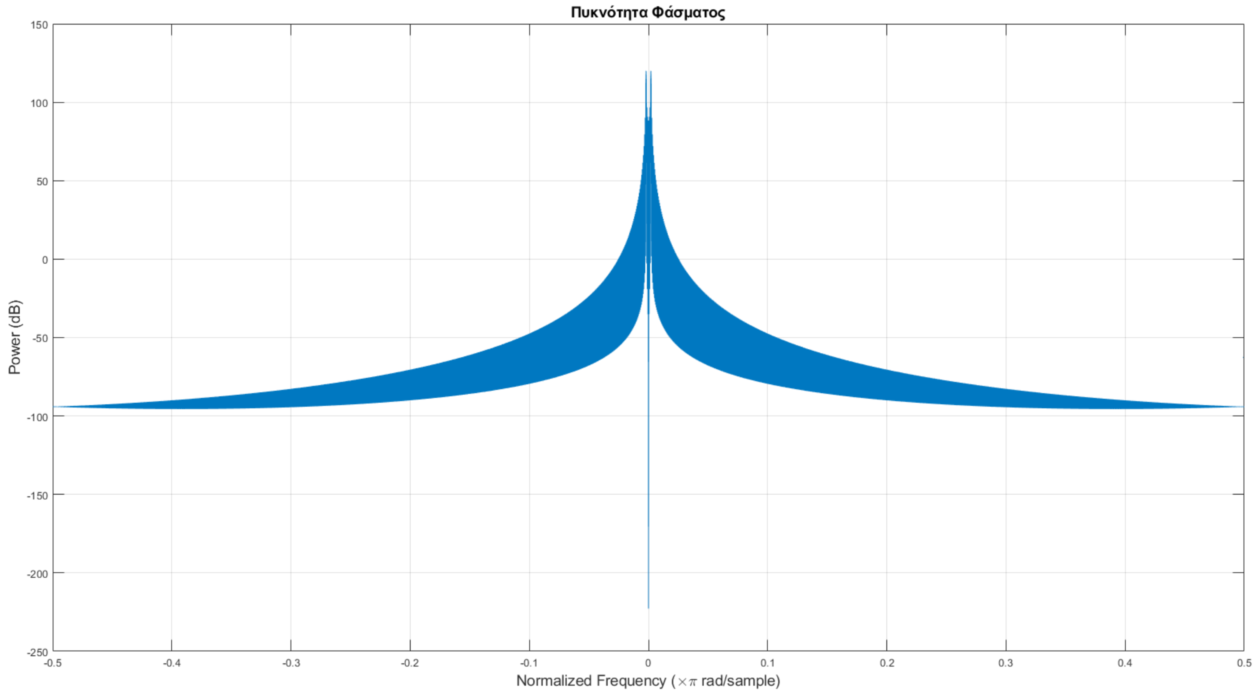
$$S_x(\omega)=\hat{f}(R_X(\omega))=\hat{f}\left(\frac{1}{2}\cdot\cos(2\cdot\pi\cdot\omega)\right)+\hat{f}(0.25)=\frac{\pi}{2}\cdot[\delta(\omega-2\pi)+\delta(\omega+2\pi)]+0.25\cdot\delta(\omega)$$

Το παραπάνω αποτέλεσμα προκύπτει έτσι, διότι γνωρίζουμε ότι ο μετασχηματισμός Fourier ενός αθροίσματος είναι οι επιμέρους μετασχηματισμοί, ο μετασχηματισμός Fourier μιας σταθεράς είναι η συνάρτηση του Dirac επί αυτή τη σταθερά και, τέλος, ο μετασχηματισμός Fourier του συνημιτόνου είναι:

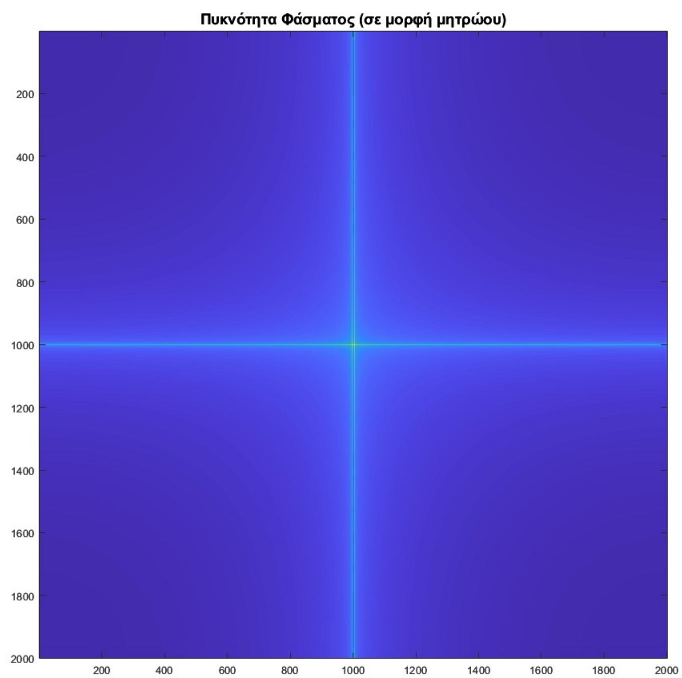
$$\hat{f}(\cos(2\cdot\pi\cdot\omega))=\pi\cdot[\delta(\omega-2\pi)+\delta(\omega+2\pi)]$$

Το αποτέλεσμα αυτού στο περιβάλλον της MATLAB ως μονοδιάστατη συνάρτηση φαίνεται στην επόμενη σελίδα.

Η πυκνότητα φάσματος του σήματος:



Τυπώνοντας το μητρώο της πυκνότητας φάσματος, βλέπουμε για *K* = 1000:



Ο κώδικας της άσκησης δίνεται στις επόμενες σελίδες.

Ο κώδικας της άσκησης είναι ο παρακάτω:

```
% Δημιουργία ημιτονοειδούς σήματος συχνότητας f = 1Hz, πλάτους A = [-1, 1],
% με fs = 500Hz και δημιουργία του αντίστοιχου γραφήματος
samples = 500;
freq = 1;
x = -2*samples : 1 : 2*samples;
n = x / samples;
y = sin(2*pi*freq*n);
figure(1);
plot(n, y);
hold on;
plot([min(n) max(n)], [0 0], 'black');
plot([0 0], [(1.1*min(y)) (1.1*max(y))], 'black');
hold off;
xlabel('Time');
ylabel('Amplitude');
title('Ημιτονοειδές Σήμα');
ylim([(1.1*min(y)) (1.1*max(y))]);

% Δημιουργία K υλοποιήσεων του στοχαστικού σήματος φορτώνοντας τη
% κατανομή Γάμμα από το αντίστοιχο .mat αρχείο
K = 1000;
load('gamma_distribution.mat', 'pd');
A = random(pd, K, 1);
X = A + y;      % Υλοποίηση της στοχαστικής διαδικασίας
X1 = 1 + y;     % Ξεχωριστή υλοποίηση με τη μέση τιμή της κατανομής

% Υπολογισμός τιμών μέσης τιμής, διασποράς, αυτοσυσχέτισης και
% πυκνότητας φάσματος
arithmetic_mean = mean(mean(X));
variance = mean(var(X));
autocorrelation_p = xcorr(y);
autocorrelation_m = X' * X / K;
spectral_density_p = 20*log10(fftshift(abs(fft(autocorrelation_p))));
spectral_density_m = 20*log10(fftshift(abs(fft2(autocorrelation_m))));
expected_acorr = (X1)' * (X1);
expected_sd = 20*log10(fftshift(abs(fft2(expected_acorr))));

max_exp_acorr = max(expected_acorr(:));
max_acorr = max(autocorrelation_m(:));
max_exp_sd = max(expected_sd(:));
max_sd = max(spectral_density_m(:));

% Εκτύπωση τιμών μέσης τιμής, διασποράς και μέγιστης παρατήρησης
fprintf('Arithmetic Mean = %f\n', arithmetic_mean);
fprintf('Variance = %f\n\n', variance);
fprintf('Largest number expected in autocorrelation = %f\n', max_exp_acorr);
fprintf('Largest number observed in autocorrelation = %f\n\n', max_acorr);
fprintf('Largest number expected in SD = %f\n', max_exp_sd);
fprintf('Largest number observed in SD = %f\n', max_sd);
```

```
% Δημιουργία γραφημάτων
figure(2);
plot(pd);
grid on;
title('Κατανομή Γάμμα');

figure(3);
plot(n, X);
xlabel('Time');
ylabel('Amplitude');
title('Υλοποιήσεις της στοχαστικής διαδικασίας');

figure(4);
f = linspace(-0.5, 0.5, length(autocorrelation_p));
plot(f, autocorrelation_p);
grid on;
title('Αυτοσυσχέτιση του τριγωνικού κύματος');
xlabel('Normalized Frequency (\times\pi rad/sample)');
ylabel('Autocorrelation');

figure(5);
imagesc(n, n, autocorrelation_m);
axis image;
title('Αυτοσυσχέτιση του τριγωνικού κύματος (σε μορφή μητρώου)');

figure(6);
f = linspace(-0.5, 0.5, length(spectral_density_p));
plot(f, spectral_density_p);
grid on;
title('Πυκνότητα Φάσματος');
xlabel('Normalized Frequency (\times\pi rad/sample)');
ylabel('Power (dB)');

figure(7);
imagesc(spectral_density_m);
axis image;
title('Πυκνότητα Φάσματος (σε μορφή μητρώου)');
```

Θέματα: Εισαγωγή στη MATLAB για ΨΕΣ

Θέμα 2ο) Εφαρμογή FIR Φίλτρου Wiener σε Εικόνα

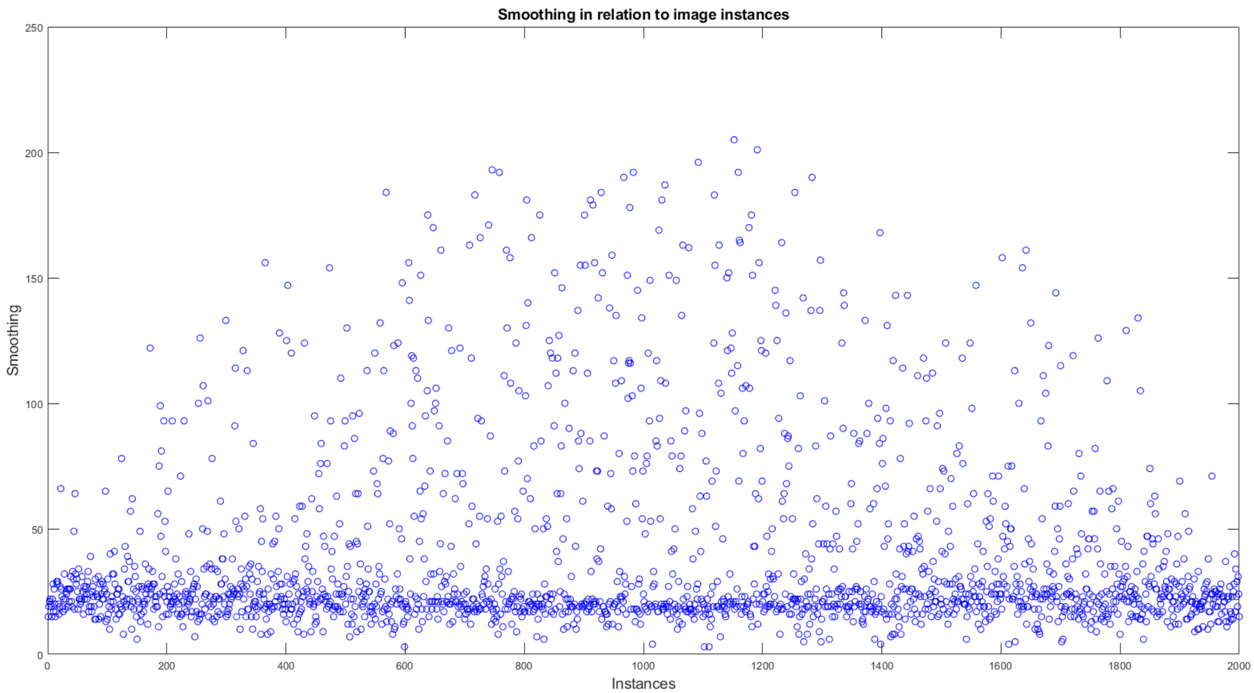
Το φίλτρο Wiener αποτελεί προσαρμοστικό φίλτρο το οποίο χρησιμοποιείται για την αποθορυβοποίηση σημάτων, υποθέτοντας γνώση πάνω στο σήμα που θέλουμε να αποθορυβοποιήσουμε και του θορύβου που το μολύνει, το οποίο γίνεται μέσω της ελαχιστοποίησης του μέσου τετραγωνικού σφάλματος μεταξύ της εκτιμώμενης τυχαίας διαδικασίας (σήμα με προσθετικό θόρυβο) και του επιθυμητού αποτελέσματος (καθαρό σήμα). Η προσέγγιση αυτή διαφέρει από τα φίλτρα που έχουμε δει ως τώρα στην λεγόμενη "κλασσική επεξεργασία σημάτων", τα οποία στοχεύουν μια συγκεκριμένη απόκριση συχνότητας διότι υποθέτουμε ότι το σήμα και ο θόρυβος δεν έχουν επικάλυψη. Συνεπώς, στην άσκηση αυτή:
α) Χρησιμοποιώντας τη συνάρτηση ***wiener2()*** της MATLAB, δείτε τα αποτελέσματα του φίλτρου σε εικόνα μολυσμένη με λευκό θόρυβο κλιμακούμενης διασποράς και μέσης τιμής ίση με το μηδέν.
β) Να κατασκευάσετε το γράφημα εξομάλυνσης που έκανε το φίλτρο ανά στιγμιότυπο της εικόνας.
γ) Να συγκρίνετε το μέσο τετραγωνικό σφάλμα μεταξύ των στιγμιοτύπων της αρχικής εικόνας με τη θορυβοποιημένη εικόνα και των στιγμιοτύπων της αρχικής εικόνας με τη φιλτραρισμένη εικόνα.
δ) Να κατασκευάσετε το γράφημα της εκτιμώμενης υπολειπόμενης ενέργειας του θορύβου σε σχέση με τη διασπορά του.

Λύση:

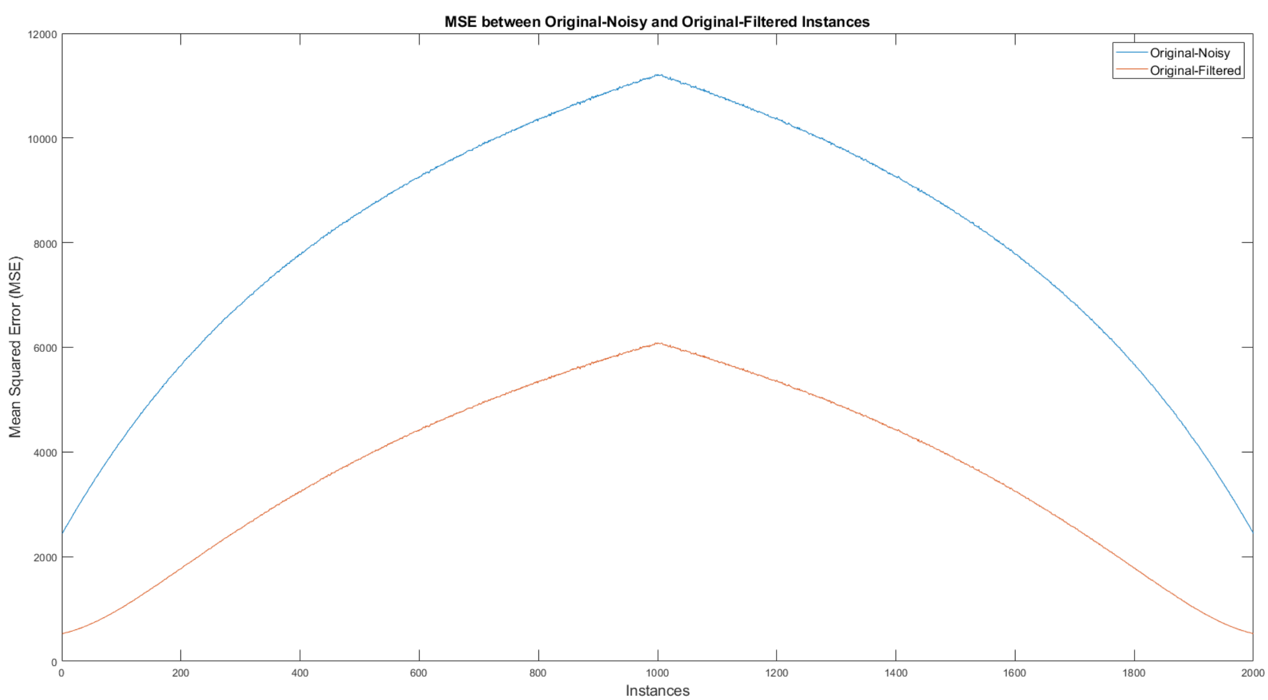
α) Παρατηρούμε ότι αυτή η προσέγγιση παράγει καλύτερα αποτελέσματα σε σχέση με τεχνικές γραμμικού φιλτραρίσματος σαν αυτές που είδαμε προηγουμένως. Το προσαρμοστικό φίλτρο είναι πιο επιλεκτικό από ένα συγκρίσιμο γραμμικό φίλτρο, οπότε διατηρεί τις ακμές, τις γωνίες και άλλα μέρη υψηλής συχνότητας μιας εικόνας. Βέβαια, απαιτεί παραπάνω υπολογιστικούς πόρους για να το πετύχει αυτό το αποτέλεσμα.

β) Παρατηρούμε ότι τα περισσότερα δείγματα φιλτράρονται περίπου το ίδιο κατά μέσο όρο με βάση στοιχεία που λαμβάνουμε από το φίλτρο Wiener, και σε κάθε περίπτωση οι ακμές και οι γωνίες της εικόνας διατηρούνται, κι ας μην αφαιρείται όλος ο θόρυβος. Το αντίστοιχο γράφημα φαίνεται στην επόμενη σελίδα.

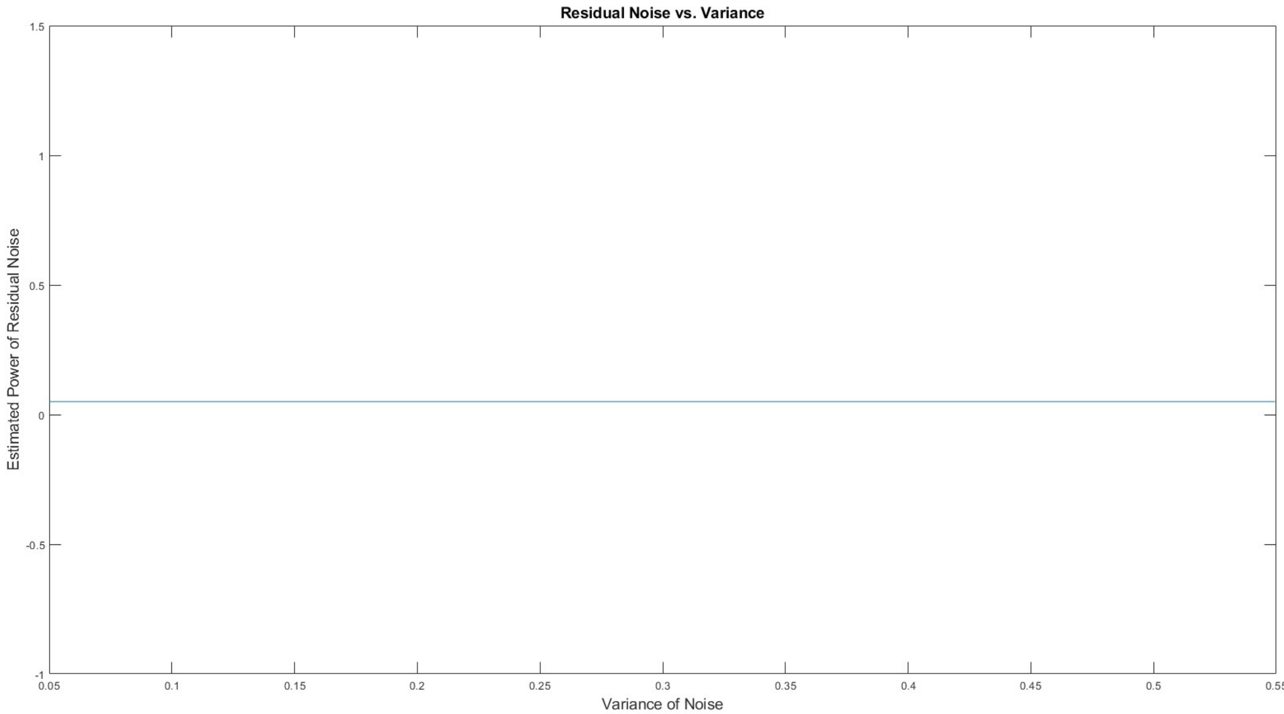
Το γράφημα φιλτραρίσματος για 2000 δείγματα είναι το παρακάτω:



γ) Παρατηρούμε ότι, κατά το φιλτράρισμα της εικόνας, μειώθηκε σημαντικά το MSE σε σχέση με τις τιμές που είχαν τα θορυβοποιημένα στιγμιότυπα. Επίσης, αν και η κλίση της καμπύλης του MSE των φιλτραρισμένων στιγμιοτύπων δεν είναι ίδια με αυτή των θορυβοποιημένων, η γενική μορφή των καμπυλών είναι πολύ παρόμοια, ειδικά στις “απότομες” αλλαγές που παρουσιάζονται ανά δείγμα. Το γράφημα σύγκρισης των καμπύλων MSE για 2000 δείγματα είναι το παρακάτω:



δ) Παρατηρούμε ότι το φίλτρο Wiener, ασχέτως της διασποράς του θορύβου που προστίθεται στην εικόνα, αφαιρεί όσο από αυτό μπορεί, και σε κάθε περίπτωση η εκτιμώμενη υπολειπόμενη ενέργεια από τον θόρυβο είναι ίδια (συγκεκριμένα 0,05). Το γράφημα φαίνεται παρακάτω:



Ο κώδικας συνολικά της άσκησης δίνεται στις επόμενες σελίδες.

Ο κώδικας της άσκησης είναι ο παρακάτω:

```
% Φόρτωση της εικόνας, μετατροπή της σε grayscale και λήψη του μεγέθους της
image = imread('acropolis.png');
I = rgb2gray(image);
imageSize = size(I);

% Αρχικοποίηση των μεταβλητών
N = 5;
numInstances = 200;
varStart = 0.05;
varEnd = 0.55;
varDropStart = numInstances / 2;
varStep = (varEnd - varStart) / varDropStart;

% Προκατανομή μνήμης για τη μεταβλητή noiseInstances
noiseInstances = cell(numInstances, 1);

% Αρχικοποίηση μητρώων για την αποθήκευση της διασποράς και
% των τιμών φιλτραρίσματος
varValues = zeros(numInstances, 1);
filteredValues = zeros(numInstances, 1);
noise_out_values = zeros(numInstances, 1);
mse_values_n = zeros(numInstances, 1);
mse_values_f = zeros(numInstances, 1);

% Βρόχος για το φιλτράρισμα σε κάθε εμφάνιση της εικόνας, αποθήκευση των
% στιγμιοτύπων θορύβου και αναπαραγωγή των εικόνων σειριακά σαν βίντεο
for instance = 1:numInstances
    % Υπολογισμός της διασποράς για το τρέχον στιγμιότυπο
    if instance <= varDropStart
        variance = varStart + (instance - 1) * varStep;
    else
        variance = varEnd - (instance - varDropStart) * varStep;
    end

    % Δημιουργία λευκού θορύβου μέσω της imnoise()
    noiseInstance = imnoise(I, 'gaussian', 0, variance);

    % Αποθήκευση της διασποράς και του στιγμιότυπου θορύβου
    varValues(instance) = variance;
    noiseInstances{instance} = noiseInstance;

    % Αναπαραγωγή των στιγμιοτύπων της θορυβοποιημένης εικόνας
    figure(1);
    imshow(noiseInstance);
    title(sprintf('Noisy Image (Variance = %.3f)', variance));
    axis off;
    drawnow;
    pause(0.001);
```

```
    % Υπολογισμός MSE μεταξύ αρχικής με τις θορυβοποιημένες εικόνες
    mse_n = sum(sum((double(I) - double(noiseInstance)).^2)) / ...
    (imageSize(1) * imageSize(2));
    mse_values_n(instance) = mse_n;
end

pause(1);

% Βρόχος για την εφαρμογή του φιλτραρίσματος σε κάθε στιγμιότυπο της
% θορυβοποιημένης εικόνας και και αναπαραγωγή των εικόνων σειριακά σαν βίντεο
for instance = 1:numInstances
    % Λήψη του θορυβοποιημένου στιγμιότυπου της εικόνας
    noiseInstance = noiseInstances{instance};

    % Εφαρμογή του wiener2()
    [filteredImage, noise_out] = wiener2(noiseInstance, [N N], variance);

    % Εμφάνιση των φιλτραρισμένων στιγμιοτύπων της εικόνας
    figure(2);
    imshow(filteredImage);
    title(sprintf('Filtered Image (Variance = %.3f)', varValues(instance)));
    axis off;
    drawnow;
    pause(0.001);

    % Αποθήκευση της τιμής φιλτραρίσματος και των τιμών θορύβου
    filteredValues(instance) = filteredImage(1, 1);
    noise_out_values(instance) = noise_out;

    % Υπολογισμός MSE μεταξύ θορυβοποιημένων και φιλτραρισμένων εικόνων
    mse_f = sum(sum((double(I) - double(filteredImage)).^2)) / ...
    (imageSize(1) * imageSize(2));
    mse_values_f(instance) = mse_f;
end

% Σχεδιασμός γραφήματος της σχέσης μεταξύ της εξομάλυνσης και
% των στιγμιοτύπων της εικόνας
figure(3);
plot(filteredValues, 'bo');
xlabel('Instances');
ylabel('Smoothing');
title('Smoothing in relation to image instances');
xlim([0 numInstances]);

% Σχεδιασμός γραφήματος του μέσου τετραγωνικού σφάλματος (MSE) μεταξύ
% των στιγμιοτύπων της αρχικής εικόνας και της θορυβώδους εικόνας σε σχέση
% με τα στιγμιότυπα της αρχικής εικόνας και της φιλτραρισμένης εικόνας
figure(4);
plot(mse_values_n);
hold on;
plot(mse_values_f);
```

```matlab
hold off;
xlabel('Instances');
ylabel('Mean Squared Error (MSE)');
title('MSE between Original-Noisy and Original-Filtered Instances');
legend('Original-Noisy', 'Original-Filtered');
xlim([0 numInstances]);

% Σχεδιασμός γραφήματος της σχέσης μεταξύ της πυκνότητας (διασπορά)
% του θορύβου και της εκτιμώμενης ισχύος του υπολειπόμενου θορύβου
figure(5);
plot(varValues, noise_out_values);
xlabel('Variance of Noise');
ylabel('Estimated Power of Residual Noise');
title('Residual Noise vs. Variance');
xlim([varStart varEnd]);
```