



ΠΟΛΥΤΕΧΝΕΙΟ ΚΡΗΤΗΣ  
Τμήμα Ηλεκτρονικών Μηχ. & Μηχ. Υπολογιστών  
Τομέας Τηλεπικοινωνιών  
Ψηφιακά Συστήματα Επικοινωνιών

# ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

με θέμα:

«ΜΕΛΕΤΗ & ΥΛΟΠΟΙΗΣΗ ΨΗΦΙΑΚΟΥ ΔΕΚΤΗ»

ΒΛΑΧΟΔΗΜΗΤΡΟΠΟΥΛΟΣ ΣΩΤΗΡΙΟΣ  
ΛΑΖΑΡΙΔΗΣ ΙΩΑΝΝΗΣ

Εξεταστική Επιτροπή:

Σιδηρόπουλος Νικόλαος  
Καρυστινός Γεώργιος

Επιβλέπων Καθηγητής:

Λιάβας Αθανάσιος

Χανιά, Οκτώβριος 2007

ΠΟΛΥΤΕΧΝΕΙΟ ΚΡΗΤΗΣ  
ΤΜΗΜΑ ΗΛΕΚΤΡΟΝΙΚΩΝ ΜΗΧ. & ΜΗΧ. ΥΠΟΛΟΓΙΣΤΩΝ  
ΤΟΜΕΑΣ ΤΗΛΕΠΙΚΟΙΝΩΝΙΩΝ

*ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ*

## **ΜΕΛΕΤΗ & ΥΛΟΠΟΙΗΣΗ ΨΗΦΙΑΚΟΥ ΔΕΚΤΗ**

ΒΛΑΧΟΔΗΜΗΤΡΟΠΟΥΛΟΣ ΣΩΤΗΡΙΟΣ  
ΛΑΖΑΡΙΔΗΣ ΙΩΑΝΝΗΣ

**Εξεταστική Επιτροπή**

**Καθ. Σιδηρόπουλος Νικόλαος**  
**Επ. Καθ. Καρυστινός Γεώργιος**

**Επιβλέπων Καθηγητής**

**Χανιά, 22 Οκτωβρίου 2007**

---

**Αν. Καθ. Λιάβας Αθανάσιος**

Φωτογραφία Εξώφυλλου: 70-meter DSN Antenna. NASA, Jet Propulsion Laboratory,  
California Institute of Technology  
(site: <http://deepspace.jpl.nasa.gov/dsn/antennas/>)

# Περιεχόμενα

<b>Πρόλογος</b>	4
<b>Κεφάλαιο 1 - Εισαγωγή</b>	
1.1 Γενικά	5
1.2 Προδιαγραφές Συστήματος Υλοποίησης	7
<b>Κεφάλαιο 2 - Δειγματοληψία</b>	
2.1 Γενικά	10
2.2 Ολίσθηση Σήματος στη Βασική Ζώνη	
2.2.i Μέθοδοι	11
2.2.ii Ολίσθηση Συχνότητας Σήματος μέσω Δειγματοληψίας	12
2.3 Προσαρμοσμένο Φίλτρο	15
<b>Κεφάλαιο 3 - Ανάκτηση Φορέα &amp; Αποδιαμόρφωση</b>	
3.1 Γενικά	18
3.2 Μέθοδοι Ανάκτησης Φορέα	
3.2.i Βρόχος Κλειδώματος Φάσης	19
3.2.ii Βρόχος Costas	22
<b>Κεφάλαιο 4 - Ανάκτηση Συγχρονισμού</b>	
4.1 Γενικά	25
4.2 Παλμός Nyquist	26
4.3 Μέθοδοι Ανάκτησης Συγχρονισμού	28
4.3.i Ελαχιστοποίηση του Cluster Variance	29
4.3.ii Μεγιστοποίηση της Ισχύος Εξόδου	31
4.4 Παρεμβολή	33



<b>Κεφάλαιο 5 - Εκτίμηση &amp; Ισοστάθμιση Καναλιού</b>	
5.1 Γενικά	35
5.2 Γραμμική Ισοστάθμιση	
5.2.i Γενικά	38
5.2.ii Γραμμική Ισοστάθμιση Ελαχίστων Τετραγώνων	40
5.2.iii Προσαρμοστική Ισοστάθμιση - Αλγόριθμος <i>LMS</i>	42
5.3 Μη Γραμμική Ισοστάθμιση	
5.3.i Γενικά	44
5.3.ii <i>Trellis</i> Αναπαράσταση	44
5.3.iii Αναδρομικός Αλγόριθμος <i>Viterbi</i>	47
5.4 Εκτίμηση Καναλιού	
5.4.i Μέθοδος Ετεροσυσχέτισης	48
5.4.ii Μέθοδος Ελαχίστων Τετραγώνων	50
5.5 Συγχρονισμός Πλαισίου	52
<b>Κεφάλαιο 6 - Προσομοίωση Συστήματος</b>	54
<b>Παράρτημα Α</b>	
Αποκωδικοποίηση	89
<b>Παράρτημα Β</b>	
Πλήρης Υλοποίηση του Συστήματος Προσομοίωσης σε περιβάλλον <i>Matlab</i>	92

“Ακούω τον παφλασμό των κυμάτων,  
η βουτιά του ανθρώπου προς το θείο...”

- *Zorz Pilali*

*Αφιερώνεται στις όμορφες, αλλά και άσχημες στιγμές  
που ζήσαμε στα Χανιά όλα αυτά τα χρόνια...*

# Πρόλογος

Ο στόχος αυτής της διπλωματικής εργασίας είναι η μελέτη, η υλοποίηση και η προσομοίωση ενός ψηφιακού δέκτη. Το σύστημα αναλύθηκε στις πέντε κύριες βαθμίδες που αναφέρονται παρακάτω:

- Ολίσθηση Συχνότητας Σήματος μέσω Δειγματοληψίας
- Ανάκτηση Φορέα & Αποδιαμόρφωση
- Ανάκτηση Συγχρονισμού
- Εκτίμηση & Ισοστάθμιση Καναλιού
- Συγχρονισμός Πλαισίου & Αποκωδικοποίηση

Με γνώμονα αυτά τα πέντε επίπεδα έγινε η συγγραφή των αντίστοιχων κεφαλαίων της παρούσας εργασίας. Ένα ακόμη κεφάλαιο περιλαμβάνει την προσομοίωση ενός ψηφιακού δέκτη (αποτελέσματα και γραφήματα), ο οποίος πληρεί τις προδιαγραφές που μας δόθηκαν, ενώ στο τέλος του συγγράμματος (Παράρτημα Β), παρατίθεται η πλήρης υλοποίηση του συστήματος προσομοίωσης σε περιβάλλον Matlab.

# 1. Εισαγωγή

## 1.1 Γενικά

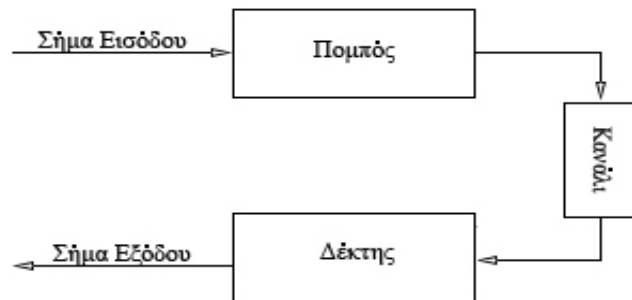
Τα συστήματα επικοινωνιών είναι σχεδιασμένα για να στέλνουν πληροφορία από μια πηγή σε έναν ή περισσότερους αποδέκτες. Η πληροφορία αυτή μπορεί να έχει την μορφή φωνής, εικόνας ή κειμένου. Γενικά, η καρδιά ενός τηλεπικοινωνιακού συστήματος αποτελείται από τρία βασικά μέρη, συγκεκριμένα τον πομπό, το κανάλι και το δέκτη. Αυτό το μοντέλο μπορεί να παρασταθεί από ένα διάγραμμα λειτουργικών βαθμίδων που φαίνεται στο σχήμα 1.1. Παρακάτω περιγράφουμε σύντομα τις λειτουργίες που επιτελούν τα τμήματα αυτά. [2]

**Πομπός.** Ο πομπός μετατρέπει την πληροφορία σε μια μορφή κατάλληλη για μετάδοση μέσα από το φυσικό κανάλι ή το μέσο διάδοσης. Εγγένει, ο πομπός επιτυγχάνει την προσαρμοσμένη αυτή σύζευξη του σήματος πληροφορίας με το κανάλι με μια διαδικασία που λέγεται διαμόρφωση. Εκτός από τη διαμόρφωση, άλλες λειτουργίες που μπορούμε συνήθως να εκτελέσουμε στον πομπό είναι η κωδικοποίηση, το φιλτράρισμα και η ενίσχυση του διαμορφωμένου σήματος πληροφορίας.

**Κανάλι.** Το κανάλι επικοινωνίας είναι το φυσικό μέσο που χρησιμεύει για την μετάδοση του σήματος από τον πομπό στο δέκτη. Το κύριο χαρακτηριστικό του καναλιού είναι ότι το μεταδιδόμενο σήμα αλλοιώνεται κατά τυχαίο τρόπο. Η πιο συνήθης μορφή υποβάθμισης του σήματος προέρχεται από έναν προσθετικό θόρυβο που γεννιέται στην είσοδο του δέκτη. Ακόμα, οι παρεμβολές από άλλους χρήστες αποτελούν έναν άλλο τύπο θορύβου, όπως επίσης και οι αλληλοπαρεμβολές που δημιουργούνται μεταξύ των συμβόλων της πληροφορίας που μεταδόθηκε από τον ίδιο χρήστη. Ο τελευταίος τύπος θορύβου, ονομάζεται *Διασυμβολική Παρεμβολή (ISI)*. Τόσο οι προσθετικές όσο και οι μη προσθετικές παραμορφώσεις του σήματος χαρακτηρίζονται συνήθως ως τυχαία φαινόμενα και περιγράφονται με στατιστικό τρόπο.



**Δέκτης.** Η διάταξη του δέκτη περιλαμβάνει αρκετές λειτουργίες όπως δειγματοληψία, αποδιαμόρφωση, εκτίμηση και ισοστάθμιση καναλιού και αποκωδικοποίηση που αποσκοπούν στην ανάκτηση του σήματος πληροφορίας που περιέχεται στο λαμβανόμενο σήμα. Η ποιότητα του δέκτη είναι συνάρτηση του τύπου διαμόρφωσης, της έντασης του προσθετικού θορύβου, του τύπου και της έντασης οποιασδήποτε προσθετικής ή μη παρεμβολής.



**Σχήμα 1.1** Διάγραμμα λειτουργικών βαθμίδων ενός τηλεπικοινωνιακού συστήματος

## 1.2 Προδιαγραφές Συστήματος Υλοποίησης

Σε αυτή την ενότητα θα αναφέρουμε τα βασικά χαρακτηριστικά του συστήματος που υλοποιήθηκε στην παρούσα εργασία. Καταρχήν, το σήμα εισόδου στον πομπό είναι ένα κείμενο αγγλικών χαρακτήρων. Κάθε χαρακτήρας έχει μετατραπεί σε μια δυαδική ακολουθία μήκους 7-bit σύμφωνα με την ASCII κωδικοποίηση. Η ακολουθία αυτή έχει κωδικοποιηθεί χρησιμοποιώντας (5,2) γραμμικό μπλοκ κώδικα<sup>1</sup> ο οποίος αντιστοιχίζει μια κωδική λέξη μήκους 5-bit σε κάθε ζεύγος bits. Στη συνέχεια, η έξοδος του μπλοκ κώδικα χωρίζεται σε ζευγάρια που συσχετίζονται με τέσσερις ακεραίους ενός 4-PAM αλφάβητου  $\pm 1$  και  $\pm 3$  μέσω της παρακάτω αντιστοίχισης του σχήματος 1.2:

$$\begin{aligned} 11 &\rightarrow +3 \\ 10 &\rightarrow +1 \\ 01 &\rightarrow -1 \\ 00 &\rightarrow -3 \end{aligned}$$

**Σχήμα 1.2** Αντιστοίχιση ζεύγων bits με 4-PAM σύμβολα

Έτσι, αν υπάρχουν  $n$  γράμματα, θα έχουμε  $7n$  μη κωδικοποιημένα bits,  $7n(\frac{5}{2})$  κωδικοποιημένα bits και  $7n(\frac{5}{2})(\frac{1}{2})$  4-PAM σύμβολα.

Στη συνέχεια, αυτά τα σύμβολα που έχουν δημιουργηθεί ομαδοποιούνται σε μπλοκ, καθένα από τα οποία έχει μήκος 875 σύμβολα. Στην αρχή κάθε μπλοκ προστίθεται μια επικεφαλίδα 245 4-PAM συμβόλων που δίνεται από τη φράση “A00h well whatever Nevermind”, η οποία θεωρείται γνωστή στο δέκτη και μπορεί να χρησιμοποιηθεί ως ακολουθία εκπαίδευσης στον ισοσταθμιστή. Η επικεφαλίδα σε συνδιασμό με το μπλοκ δεδομένων αποτελούν το πλαίσιο, έτσι κάθε πλαίσιο έχει μήκος 1120 σύμβολα. Στον πομπό, η περίοδος συμβόλου πρέπει να έχει μέγιστη απόκλιση  $\pm 0.01\%$  της επιτρεπτής περιόδου συμβόλου  $T = 6.4$  microseconds.

Το σήμα εκπομπής θα είναι της μορφής [1]

$$x(t) = \sum_k a_k p(t - kT)$$

όπου  $a_k$  είναι τα σύμβολα εισόδου και  $p(t)$  ο παλμός μορφοποίησης στον πομπό. Στη συνέχεια, το σήμα διαμορφώνεται με ένα συνημίτονο συχνότητας  $f_c$  και φάσης  $\phi$  και το σήμα που προκύπτει είναι

$$d(t) = x(t) \cos(2\pi f_c t + \phi)$$

<sup>1</sup> Λεπτομέρειες υπάρχουν στο Παράρτημα Α

Κατόπιν, μεταδίδεται διαμέσω καναλιού με κρουστική απόκριση  $c(t)$  και προστίθεται θόρυβος  $n(t)$ . Το σήμα αποδιαμορφώνεται στο δέκτη και τελικά έχει τη μορφή

$$r(t) = (d(t) * c(t) + n(t)) \cos(2\pi\hat{f}_c t + \hat{\phi})$$

όπου  $\hat{f}_c$  και  $\hat{\phi}$  είναι εκτιμήσεις της συχνότητας και της φάσης αντίστοιχα. Υποθέτοντας ότι  $\hat{f}_c = f_c$  και  $\hat{\phi} = \phi$ , τότε στην έξοδο του φίλτρου λήψης  $g_R(t)$  λαμβάνουμε

$$y(t) = d(t) * c'(t) * g_R(t) + n'(t)$$

με  $c'(t)$  το ισοδύναμο κανάλι βασικής ζώνης και

$$n'(t) = (n(t) * g_R(t)) \cos(2\pi\hat{f}_c t + \hat{\phi})$$

Ορίζοντας,  $p'(t) = p(t) * c'(t) * g_R(t)$ , έχουμε [1]

$$y(t) = \sum_k a_k p'(t - kT) + n'(t)$$

Τέλος, δειγματοληπτείται κατάλληλα η αναλογική κυματομορφή εισόδου στο δέκτη τις χρονικές στιγμές  $kT_s$  και τελικά λαμβάνουμε την ακολουθία

$$y_k = \sum_l h_l a_{k-l} + n_k$$

όπου  $h_l = h[lT_s]$  και  $h(t) = p(t) * c(t) * g_R(t)$

Η κυματομορφή σηματοδοσίας  $P(f)$  είναι μια τετραγωνική ρίζα ανυψωμένου συνημιτόνου, συνολικού μήκους 8 περιόδων συμβόλου. Ο συντελεστής επέκτασης  $\beta$  της κυματομορφής σηματοδοσίας έχει συγκεκριμένη τιμή, γνωστή στο δέκτη και παίρνει μία από τις τιμές στο διάστημα  $[0.1, 0.3]$ . Το εύρος ζώνης της κυματομορφής σηματοδοσίας μπορεί να είναι ως  $\approx 102\text{kHz}$  για το επιτρεπτό  $T$ . Λόγω διαμόρφωσης διπλής πλευρικής ζώνης, το εύρος ζώνης της κυματομορφής σηματοδοσίας διπλασιάζεται, έτσι κάθε ζωνοπερατό FDM σήμα χρειάζεται εύρος ζώνης τουλάχιστον  $204\text{kHz}$ .

Το κανάλι μπορεί να είναι ιδανικό (απλώς εισάγει μια καθυστέρηση) ή μπορεί να έχει σημαντική διασυμβολική παρεμβολή (ISI). Σε κάθε περίπτωση, η κρουστική απόκριση του καναλιού είναι άγνωστη στο δέκτη, ωστόσο η μέγιστη καθυστέρηση που μπορεί να εισαχθεί, είναι γνωστή. Κατά τη διάρκεια της μετάδοσης μπορεί να έχουμε θόρυβο μεγάλου εύρους ζώνης με επίπεδη φασματική πυκνότητα ισχύος ή στενού εύρους ζώνης παρεμβολείς ή και τα δύο. Αυτά είναι άγνωστα στον δέκτη επίσης. Στο σχήμα 1.3 παρατίθενται συγκεντρωμένες οι προδιαγραφές του ψηφιακού μας δέκτη.

---

Αλφάβητο Συμβόλων	$\pm 1, \pm 3$
Ενδιάμεση Συχνότητα	2 MHz
Επιτρεπτή Περίοδος Συμβόλου	6.4 microseconds
Συντελεστής Επέκτασης SRRC Φίλτρου	$\beta \in [0.1, 0.3]$
Εύρος Ζώνης Σήματος	204 kHz
Μήκος SRRC Φίλτρου	8 Περίοδοι Συμβόλου
Επικεφαλίδα Πλαισίου/Ακολουθία Εκπαίδευσης	A0Oh well whatever Nevermind
Μήκος Πλαισίου	1120 Σύμβολα
Απόκλιση Ενδιάμεσης Συχνότητας	$< 0.01\%$ της Δοθείσας Τιμής
Απόκλιση Συγχρονισμού	Σταθερή
Απόκλιση Περιόδου Συμβόλου	$< 0.01\%$ της Δοθείσας Τιμής
Διασυμβολική Παρεμβολή (ISI)	Μέγιστη καθυστέρηση 8 Συμβόλων
Συχνότητα Δειγματοληψίας	850 kHz

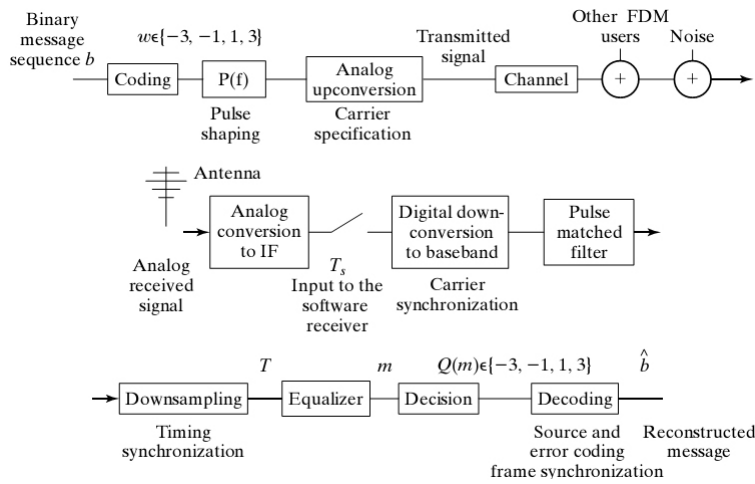
---

Σχήμα 1.3 Προδιαγραφές Συστήματος



## 2. Δειγματοληψία

### 2.1 Γενικά



Σχήμα 2.1 4-PAM Τηλεπικοινωνιακό Σύστημα

Μια αναπαράσταση του τηλεπικοινωνιακού συστήματος που μας απασχόλησε είναι αυτή του σχήματος 2.1 [1]. Εμείς επικεντρώσαμε την προσοχή μας στον δέκτη. Στην πρώτη βαθμίδα επεξεργασίας του λαμβανόμενου σήματος βρίσκεται η διαδικασία της δειγματοληψίας. Λόγω της φύσης των καναλιών δεν μπορεί να αποφευχθεί η αναλογική μετάδοση πληροφορίας. Η δειγματοληψία, η οποία μπορεί να μοντελοποιηθεί σαν ένας πολλαπλασιασμός σημείο προς σημείο στο πεδίο του χρόνου με μια παλμοσειρά  $\delta$  του *Dirac*, μεταφράζει το λαμβανόμενο σήμα από αναλογικό σε ψηφιακό και ουσιαστικά γεφυρώνει τον αναλογικό κόσμο με τις ψηφιακές μεθόδους [2]. Τα αποτελέσματα αυτής γίνονται εμφανή όταν δούμε το σήμα στο πεδίο της συχνότητας. Εξίσου σημαντική είναι η μετατροπή του σήματος από ψηφιακό σε αναλογικό. Το *Θεώρημα Δειγματοληψίας του Nyquist* εγγυάται την πλήρη ανακατασκευή του σήματος από τα δείγματά του όταν η συχνότητα δειγματοληψίας είναι τουλάχιστον διπλάσια από τη μέγιστη συχνότητα του σήματος.

## 2.2 Ολίσθηση Σήματος στη Βασική Ζώνη

### 2.2.i Μέθοδοι

Για να έχουμε επιτυχημένη μετάδοση πληροφορίας, πρέπει το σήμα από τον πομπό να σταλθεί σε υψηλές συχνότητες με μια διαδικασία που ονομάζεται *Διαμόρφωση*, που ουσιαστικά είναι ο πολλαπλασιασμός του σήματος πληροφορίας με ένα ημιτονοειδές φέρον στο πεδίο του χρόνου. Στην πλευρά του δέκτη, ακολουθείται μια διαδικασία αντίστροφη αυτής της διαμόρφωσης, που ονομάζεται *Αποδιαμόρφωση* και ουσιαστικά ολισθαίνει τη συχνότητα του σήματος στη βασική ζώνη. Υπάρχουν 2 τρόποι για να το επιτύχουμε αυτό: [1]

- Πολλαπλασιάζουμε απευθείας το λαμβανόμενο σήμα με ένα ημιτονοειδές φέρον συχνότητας ίδιας με αυτή του σήματος
- Ολισθαίνουμε το σήμα σε μια επιθυμητή ενδιάμεση συχνότητα  $f_I$  (IF) και κατόπιν μέσω δειγματοληψίας μπορούμε να ολισθήσουμε περαιτέρω το σήμα σε χαμηλότερες συχνότητες, πλησιέστερες στις συχνότητες της βασικής ζώνης. Στη συνέχεια έχουμε τους παρακάτω τρόπους με τους οποίους μπορεί να γίνει η δειγματοληψία:
  - με ρυθμό μεγαλύτερο από το ρυθμό Nyquist της ενδιάμεσης συχνότητας
  - με ρυθμό  $f_s$  μικρότερο του ρυθμού Nyquist σχετιζόμενου με την  $f_I$ , όπου δημιουργούνται ρέπλικες του φάσματος του σήματος κάθε  $f_s$ , στις συχνότητες  $f = |f_I - nf_s|$ , όπου  $n \in \mathbb{Z}$ . Η επιλογή της συχνότητας δειγματοληψίας γίνεται με τρόπο ώστε να είναι τουλάχιστον διπλάσια του εύρους ζώνης του σήματος. Τέλος, η πλησιέστερη ρέπλικα στη βασική ζώνη πολλαπλασιάζεται στο πεδίο του χρόνου με έναν τοπικό φορέα συντονισμένο σε συχνότητα ίση με αυτή στην οποία είναι κεντραρισμένο το φάσμα του σήματος
  - με τέτοιο τρόπο ώστε μία από τις ρέπλικες να εμφανιστεί απευθείας στη βασική ζώνη

Στην πρώτη μέθοδο δεν θα συμβεί αλλοίωση. Όμως, αυτή η μέθοδος μπορεί να είναι ακριβή λόγω των υψηλών ρυθμών δειγματοληψίας που απαιτούνται για να πετύχουμε τον ρυθμό Nyquist. Η τρίτη μέθοδος επιτρέπει ολίσθηση της συχνότητας του σήματος στη βασική ζώνη χωρίς επιπρόσθετο ταλαντωτή. Η δεύτερη μέθοδος ολισθαίνει ένα μέρος της συχνότητας του σήματος με δειγματοληψία και ένα μέρος με μίξη με

ένα ημιτονοειδές φέρον. Αυτή η μέθοδος χρησιμοποιήθηκε στην παρούσα διπλωματική εργασία

Τέλος, χρησιμοποιούμε χαμηλοπερατό φίλτρο με συχνότητα αποκοπής ίση με το εύρος ζώνης  $B$  του σήματος  $f_{cutoff} = B$  για να απαλλαγούμε από τις άχρηστες ρέπλικες που έχουν δημιουργηθεί. Υπάρχουν 2 βασικά πλεονεκτήματα χρησιμοποιώντας τη δεύτερη διαδικασία, η οποία ακολουθήθηκε στην παρούσα εργασία: [1]

1. Η χρήση της ίδιας ενδιάμεσης συχνότητας, μας επιτρέπει να έχουμε σταθερά τους ίδιους ενισχυτές, διαμορφωτές και φίλτρα
2. Η επεξεργασία σήματος σε υψηλές συχνότητες απαιτεί ηλεκτρονικές διατάξεις υψηλής ακρίβειας και κόστους. Με τη βοήθεια της ενδιάμεσης συχνότητας αποφεύγουμε το ρυθμό δειγματοληψίας στη συχνότητα μετάδοσης

## 2.2.ii Ολίσθηση Συχνότητας Σήματος μέσω Δειγματοληψίας

Έστω  $s(t)$  το λαμβανόμενο σήμα. Θεωρούμε ότι το σήμα πληροφορίας είναι αναλογικό, περιορισμένου εύρους ζώνης  $\pm B$  και ότι μεταδόθηκε με έναν τοπικό φορέα στον πομπό με συχνότητα  $f_c$ . Έστω, η ιδανική συνάρτηση δειγματοληψίας (Παλμοσειρά Dirac): [9]

$$\delta_{T_s}(t) = \sum_{n=-\infty}^{+\infty} \delta(t - nT_s),$$

όπου  $T_s$  η περίοδος δειγματοληψίας και  $n \in \mathbb{Z}$ .

Το δειγματοληπτημένο σήμα  $y(t)$  δίνεται από την παρακάτω εξίσωση:

$$y(t) = s(t) \sum_{n=-\infty}^{+\infty} \delta(t - nT_s),$$

το οποίο μπορεί να μετατραπεί στο πεδίο της συχνότητας ως:

$$Y(f) = \frac{1}{T_s} \sum_{n=-\infty}^{+\infty} S(f - nf_s),$$

όπου  $f_s = \frac{1}{T_s}$ , όπου  $f_s$  η συχνότητα δειγματοληψίας.

Απόδειξη [9]

$$\begin{aligned}
 y(t) &= s(t) \sum_{n=-\infty}^{+\infty} \delta(t - nT_s) \Rightarrow \\
 \mathcal{F}\{y(t)\} &= \mathcal{F}\left\{s(t) \sum_{n=-\infty}^{+\infty} \delta(t - nT_s)\right\} \Rightarrow \\
 Y(f) &= S(f) * \mathcal{F}\left\{\sum_{n=-\infty}^{+\infty} \delta(t - nT_s)\right\} \quad (1)
 \end{aligned}$$

Το σήμα είναι πραγματικό και η παλμοσειρά Dirac είναι περιοδική με περίοδο  $T_s$ , άρα μπορούμε να χρησιμοποιήσουμε απλή τριγωνομετρική σειρά Fourier, η οποία είναι:

$$\sum_{n=-\infty}^{+\infty} \delta(t - nT_s) = \frac{1}{T_s} \left[ 1 + 2 \sum_{n=1}^{+\infty} \cos(2\pi n f_s t) \right] \quad (2)$$

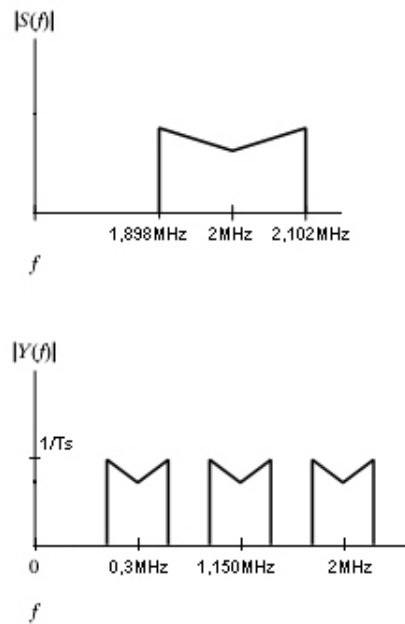
Από (1),(2) έχουμε:

$$\begin{aligned}
 Y(f) &= S(f) * \frac{1}{T_s} \left[ \mathcal{F}\{1\} + 2 \sum_{n=1}^{+\infty} \mathcal{F}\{\cos(2\pi n f_s t)\} \right] \\
 &= \frac{1}{T_s} S(f) * \left[ \delta(f) + 2 \sum_{n=1}^{+\infty} \frac{1}{2} [\delta(f - n f_s) + \delta(f + n f_s)] \right] \\
 &= \frac{1}{T_s} \left[ S(f) * \delta(f) + \sum_{n=1}^{+\infty} [S(f) * \delta(f - n f_s) + S(f) * \delta(f + n f_s)] \right] \\
 &= \frac{1}{T_s} \left[ S(f) + \sum_{n=1}^{+\infty} [S(f - n f_s) + S(f + n f_s)] \right] \\
 &= \frac{1}{T_s} \sum_{n=-\infty}^{+\infty} S(f - n f_s) \Rightarrow Y(f) = \frac{1}{T_s} \sum_{n=-\infty}^{+\infty} S(f - n f_s)
 \end{aligned}$$

Από την παραπάνω εξίσωση παρατηρούμε ότι δειγματοληπτώντας το σήμα εμφανίζονται στο πεδίο της συχνότητας ρέπλικες του φάσματος του αναλογικού σήματος, μια εκ των οποίων πιθανώς να βρίσκεται στη βασική ζώνη.



Στην περίπτωση μας, έχουμε ενδιάμεση συχνότητα  $f_I = 2 \text{ MHz}$  και συχνότητα δειγματοληψίας  $f_s = 850 \text{ kHz}$ , άρα εμφανίζονται ρέπλικες κεντραρισμένες γύρω από τις συχνότητες  $f = |f_I - n f_s|$ . Έτσι, η πλησιέστερη στη βασική ζώνη ρέπλικα είναι κεντραρισμένη γύρω από τη συχνότητα των  $300 \text{ kHz}$ , όπως φαίνεται στο σχήμα 2.2.



**Σχήμα 2.2** Στο πρώτο σχήμα φαίνεται το διαμορφωμένο σήμα μετάδοσης, ολισθημένο στην ενδιάμεση συχνότητα  $f_I$  και στο δεύτερο σχήμα τα φάσματα που δημιουργούνται μετά τη δειγματοληψία

## 2.3 Προσαρμοσμένο Φίλτρο

Ένα τέτοιο χαμηλοπερατό φίλτρο που μας βοηθάει να απαλλαγούμε από τις άχρηστες ρέπλικες που δημιουργούνται μετά τη δειγματοληψία και την αποδιαμόρφωση ονομάζεται *Προσαρμοσμένο Φίλτρο*. Το κύριο χαρακτηριστικό του φίλτρου αυτού είναι η μεγιστοποίηση του σηματοθορυβικού λόγου (SNR). Έστω, όπως φαίνεται και στο μπλοκ διάγραμμα του σχήματος 2.3 [1]

$$y(t) = u(t) + w(t)$$

η έξοδος του γραμμικού φίλτρου στο δέκτη που έχει κρουστική απόκριση  $h_R(t)$ . Έτσι, οι συνιστώσες σήματος και θορύβου αντίστοιχα μπορούν να περιγραφούν ως [1]

$$u(t) = h_R(t) * g(t), \quad w(t) = h_R(t) * n(t)$$

όπου,  $g(t)$  είναι το σήμα εκπομπής και  $n(t)$  ο προσθετικός θόρυβος.

Υποθέτουμε ότι ο θόρυβος είναι λευκός και έχει φασματική πυκνότητα ισχύος  $P_n(f)$  ίση με μια σταθερά  $N_0$  για όλες τις συχνότητες. Στόχος μας είναι να βρούμε το φίλτρο στο δέκτη, το οποίο μεγιστοποιεί το σηματοθορυβικό λόγο στις χρονικές στιγμές δειγματοληψίας. Έστω,  $h_R(t)$  τέτοιο ώστε να μεγιστοποιεί την ισχύ του σήματος  $u(t)$  συγκριτικά με την ισχύ του θορύβου, την χρονική στιγμή  $t = \tau$ . Χρησιμοποιώντας μετασχηματισμούς Fourier, μπορούμε να εκφράσουμε το  $u(\tau)$  ως εξής [1]

$$u(\tau) = \int_{-\infty}^{+\infty} H_R(f)G(f)e^{j2\pi f\tau}df$$

με  $H_R(f)$  και  $G(f)$  να είναι οι μετασχηματισμοί Fourier των  $h_R(t)$  και  $g(t)$ , αντίστοιχα. Στη συνέχεια, έχουμε

$$u^2(\tau) = \left| \int_{-\infty}^{+\infty} H_R(f)G(f)e^{j2\pi f\tau}df \right|^2$$

Ο θόρυβος στην έξοδο του προσαρμοσμένου φίλτρου έχει φασματική πυκνότητα ισχύος [1]

$$\mathcal{P}_w(f) = |H_R(f)|^2 \mathcal{P}_n(f) = N_0 |H_R(f)|^2$$

Η ποσότητα που θέλουμε να μεγιστοποιήσουμε είναι

$$\frac{u^2(\tau)}{P_w} = \frac{\left| \int_{-\infty}^{+\infty} H_R(f)G(f)e^{j2\pi f\tau}df \right|^2}{\int_{-\infty}^{+\infty} N_0 |H_R(f)|^2 df}$$

Η μεγιστοποίηση του αριθμητή επιτυγχάνεται εύκολα χρησιμοποιώντας την ανισότητα *Cauchy-Schwarz* η οποία δηλώνει γενικά ότι, αν τα σήματα  $a(x)$  και  $b(x)$  είναι σήματα πεπερασμένης ενέργειας, τότε

$$\left| \int_{-\infty}^{+\infty} a(x)b(x)dx \right|^2 \leq \int_{-\infty}^{+\infty} |a(x)|^2 dx \int_{-\infty}^{+\infty} |b(x)|^2 dx$$

και η ισότητα ισχύει μόνο όταν  $a(x) = kb(x)$ . Τότε, έχουμε

$$\frac{u^2(\tau)}{P_w} \leq \frac{\int_{-\infty}^{+\infty} |H_R(f)|^2 df \int_{-\infty}^{+\infty} |G(f)e^{j2\pi f\tau}|^2 df}{N_0 \int_{-\infty}^{+\infty} |H_R(f)|^2 df}$$

το οποίο μεγιστοποιείται με ισότητα όταν

$$H_R(f) = kG(f)e^{j2\pi f\tau}$$

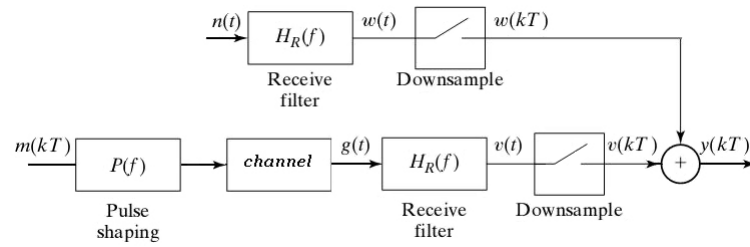
Παίρνοντας τον αντίστροφο μετασχηματισμό Fourier της τελευταίας σχέσης για να βρούμε την κρουστική απόκριση του  $h_R(t)$ , έχουμε

$$h_R(t) = \mathcal{F}^{-1} \{kG(f)e^{j2\pi f\tau}\} = kg(\tau - t)$$

όπου για την διεξαγωγή του παραπάνω αποτελέσματος χρησιμοποιήθηκε η ιδιότητα χρονικής ολίσθησης του μετασχηματισμού Fourier. Τα χαρακτηριστικά του φίλτρου αυτού είναι: [1]

- Το φίλτρο μεγιστοποιεί το σηματοθρομβικό λόγο του  $u^2(t)/P_w$  τη χρονική στιγμή  $t = \tau$
- Επειδή, η κρουστική απόκριση του φίλτρου είναι ο “καθρέπτης” του παλμού μορφοποίησης  $p(t)$  στον πομπό, λέγεται ότι το φίλτρο είναι “προσαρμοσμένο” στον παλμό μορφοποίησης και γι’αυτό το λόγο καλείται προσαρμοσμένο φίλτρο
- Το μέτρο του φάσματος του προσαρμοσμένου φίλτρου  $H_R(f)$  είναι το ίδιο με αυτό του  $G(f)$
- Το μέτρο του φάσματος του  $G(f)$  είναι το ίδιο με αυτό της απόκρισης συχνότητας του παλμού μορφοποίησης  $P(f)$
- Το προσαρμοσμένο φίλτρο για κάθε φίλτρο με άρτιας συμμετρίας, πεπερασμένη στο χρόνο, κρουστική απόκριση είναι μια καθυστερημένη ρέπλικα του ίδιου του φίλτρου. Η ελάχιστη καθυστέρηση είναι το άνω όριο του πεπερασμένου χρονικού διαστήματος της κρουστικής απόκρισης

- Ο σηματοθρομβικός λόγος εξόδου εξαρτάται από την ενέργεια της κυματομορφής εισόδου και όχι από τα ιδιαίτερα χαρακτηριστικά της



Σχήμα 2.3 Φιλτράρισμα στον Δέκτη



## 3. Ανάκτηση Φορέα & Αποδιαμόρφωση

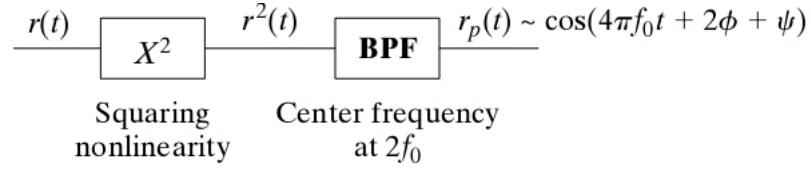
### 3.1 Γενικά

Το κεφάλαιο αυτό δίνει έμφαση σε μεθόδους *Αποδιαμόρφωσης* σήματος. Το σήμα ταξιδεύει διαμέσω καναλιού σε μια αναλογική κυματομορφή και κατόπιν φτάνοντας στο δέκτη αποδιαμορφώνεται και δειγματοληπτείται προκειμένου να ανακτήσουμε το αρχικό σήμα πληροφορίας. Για να είναι επιτυχημένη η αποδιαμόρφωση, ο δέκτης πρέπει να γνωρίζει τόσο τη συχνότητα όσο και τη φάση του ημιτονοειδούς διαμόρφωσης που χρησιμοποιήθηκε στον πομπό. Θα αναλύσουμε μεθόδους με τις οποίες μπορούμε να εκτιμήσουμε τη φάση και τη συχνότητα του τοπικού φορέα που θα χρησιμοποιηθεί για την αποδιαμόρφωση. Αυτή η διαδικασία εκτίμησης ονομάζεται *Ανάκτηση Φορέα*. Οι μέθοδοι που θα αναλυθούν είναι:

- Βρόχος Κλειδώματος Φάσης (Phase Locked Loop)
- Βρόχος Costas (Costas Loop)

## 3.2 Μέθοδοι Ανάκτησης Φορέα

### 3.2.i Βρόχος Κλειδώματος Φάσης



Σχήμα 3.1 Προεπεξεργασία σήματος

Για να οδηγήσει το λαμβανόμενο σήμα  $r(t)$  ένα βρόχο κλειδώματος φάσης, θα πρέπει να προεπεξεργαστεί, όπως φαίνεται στο σχήμα 3.1.

Το λαμβανόμενο σήμα είναι: [10]

$$r(t) = s(t) \cos(2\pi f_0 t + \phi)$$

όπου  $f_0, \phi$ , η συχνότητα και η φάση του τοπικού φορέα διαμόρφωσης, αντίστοιχα. Αν υψώσουμε το  $r(t)$  στο τετράγωνο, το σήμα που προκύπτει περιέχει μια φασματική συνιστώσα στο διπλάσιο της συχνότητας του φέροντος, δηλαδή:

$$\begin{aligned} r^2(t) &= s^2(t) \cos^2(2\pi f_0 t + \phi) \\ &= \frac{1}{2} s^2(t) (1 + \cos(4\pi f_0 t + 2\phi)) \end{aligned}$$

Θέτουμε: [10]

$$s^2(t) = s_{avg}^2 + u(t)$$

όπου  $s_{avg}^2$  είναι η μέση τιμή του  $s^2(t)$  και  $u(t)$  δηλώνει τη διακύμανση του  $s^2(t)$  γύρω από τη μέση τιμή του. Τότε, το  $r^2(t)$  μπορεί να εκφραστεί ως:

$$r^2(t) = \frac{1}{2} [s_{avg}^2 + u(t) + s_{avg}^2 \cos(4\pi f_0 t + 2\phi) + u(t) \cos(4\pi f_0 t + 2\phi)]$$

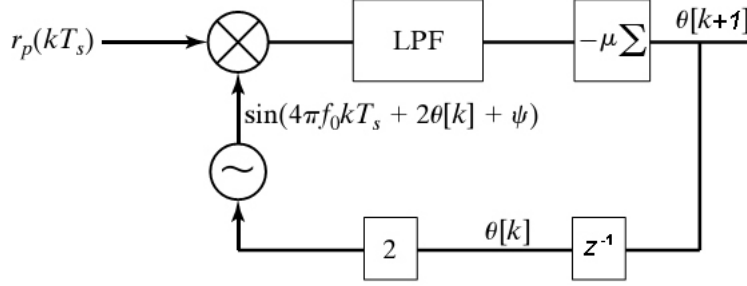
Για να απομονώσουμε την επιθυμητή συνιστώσα διπλάσιας συχνότητας από τις υπόλοιπες συχνοτικές συνιστώσες, το υψωμένο στο τετράγωνο σήμα εισόδου διέρχεται από ένα φίλτρο στενής ζώνης συντονισμένο στη συχνότητα  $2f_0$ .

Η έξοδος του ζωνοπερατού φίλτρου είναι: [10]

$$r_p(t) = BPF \{r^2(t)\} \approx \frac{1}{2} s_{avg}^2 \cos(4\pi f_0 t + 2\phi + \psi)$$

όπου  $\psi$  είναι η ολίσθηση φάσης που υπεισέρχεται από το ζωνοπερατό φίλτρο στη συχνότητα  $2f_0$ .

Η πιο διαδεδομένη μέθοδος εκτίμησης φάσης είναι ο *Βρόχος Κλειδώματος Φάσης (PLL)*, όπως φαίνεται στο σχήμα 3.2. Αυτή η μέθοδος μπορεί να εξαχθεί ως ένας προσαρμοστικός αλγόριθμος μεγιστοποίησης μιας απλής συνάρτησης απόδοσης.



Σχήμα 3.2 Μπλοκ Διάγραμμα PLL

Υποθέτοντας ότι το σήμα που θα οδηγήσει το PLL είναι: [10]

$$r_p(t) = \cos(4\pi f_0 t + 2\phi + \psi),$$

ορίζουμε τη συνάρτηση απόδοσης: [10]

$$\begin{aligned}
 \mathcal{J}_{PLL}(\theta) &= \frac{1}{2} LPF \{r_p(t) \cos(4\pi f_0 t + 2\theta + \psi)\} \\
 &= \frac{1}{2} LPF \{\cos(4\pi f_0 t + 2\phi + \psi) \cos(4\pi f_0 t + 2\theta + \psi)\} \\
 &= \frac{1}{4} LPF \{\cos(2\phi - 2\theta) + \cos(8\pi f_0 t + 2\theta + 2\phi + 2\psi)\} \\
 &= \frac{1}{4} LPF \{\cos(2\phi - 2\theta)\} + \frac{1}{4} LPF \{\cos(8\pi f_0 t + 2\theta + 2\phi + 2\psi)\} \\
 &\approx \frac{1}{4} \cos(2\phi - 2\theta),
 \end{aligned}$$

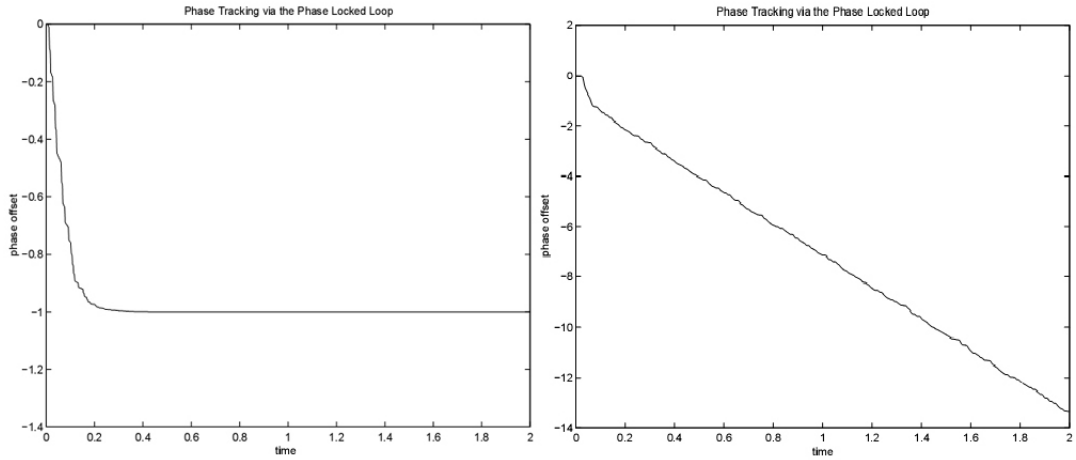
υποθέτοντας ότι η συχνότητα αποκοπής του χαμηλοπερατού φίλτρου είναι κατά πολύ μικρότερη της συχνότητας  $4f_0$ . Η παραπάνω συνάρτηση απόδοσης μεγιστοποιείται για  $\theta = \phi + k\pi$ , με  $k$  ακέραιο.

Η παράγωγος της  $\mathcal{J}_{PLL}(\theta)$  τη χρονική στιγμή  $k$  μπορεί να προσεγγιστεί ως εξής, με  $r_p(kT_s)$  τα δείγματα του λαμβανόμενου σήματος: [10]

$$\begin{aligned}
 \frac{d\mathcal{J}_{PLL}(\theta)}{d\theta} &= \frac{1}{2} \frac{dLPF \{r_p(kT_s) \cos(4\pi f_0 kT_s + 2\theta + \psi)\}}{d\theta} \bigg|_{\theta=\theta[k]} \\
 &\approx \frac{1}{2} LPF \left\{ \frac{dr_p(kT_s) \cos(4\pi f_0 kT_s + 2\theta + \psi)}{d\theta} \bigg|_{\theta=\theta[k]} \right\} \\
 &= LPF \{-r_p(kT_s) \sin(4\pi f_0 kT_s + 2\theta(k) + \psi)\}
 \end{aligned}$$

Ο αντίστοιχος προσαρμοστικός αλγόριθμος, που είναι της μορφής πιο Απότομης Ανάβασης (*Steepest Ascent*) είναι: [10]

$$\begin{aligned}\theta[k+1] &= \theta[k] + \mu \left. \frac{dJ_{PLL}(\theta)}{d\theta} \right|_{\theta=\theta[k]} \\ &= \theta[k] - \mu LPF \{r_p(kT_s) \sin(4\pi f_0 kT_s + 2\theta[k] + \psi)\}\end{aligned}$$



Σχήμα 3.3 Εκτίμηση φάσης

Αν δεν υπάρχει διαφορά συχνότητας, τότε η εκτίμηση της φάσης του φέροντος συγκλίνει σε μια τιμή, όπως φαίνεται στην αριστερή εικόνα του σχήματος 3.3. Αν έχουμε απόκλιση στη συχνότητα, τότε θα πρέπει να χρησιμοποιηθεί διάταξη διπλού PLL για την εκτίμηση τόσο της φάσης όσο και της συχνότητας του φέροντος. Ο πρώτος βρόχος υπολογίζει μια ευθεία με κλίση ανάλογη της διαφοράς ανάμεσα στην πραγματική συχνότητα του φέροντος στον πομπό και της συχνότητας στο δέκτη, όπως φαίνεται στη δεξιά εικόνα του σχήματος 3.3. Ο δεύτερος βρόχος υπολογίζει μια σταθερά κατάλληλη για την εκτίμηση της φάσης. [1]

Υποθέτουμε ότι το λαμβανόμενο προεπεξεργασμένο σήμα είναι: [1]

$$r_p(t) = \cos(4\pi f_0 t + 2\phi)$$

Αυτό εφαρμόζεται ως είσοδος στο διπλό PLL. Ο άνω βρόχος λειτουργεί ακριβώς όπως και στην περίπτωση του απλού PLL: αν η συχνότητα σε κάθε ταλαντωτή είναι  $2f_c$ , τότε η εκτίμηση φάσης  $2\theta_1$  συγκλίνει σε μια ευθεία με κλίση  $2\pi(f_0 - f_c)$ . [1]

Δηλαδή,

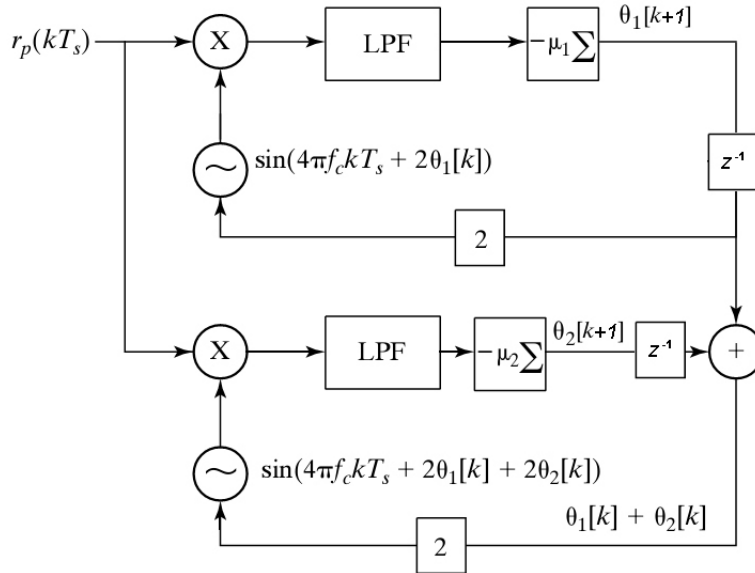
$$\theta_1(t) \rightarrow 2\pi(f_0 - f_c)t + b,$$

όπου  $b$  είναι το σημείο που τέμνει η ευθεία τον  $y$ -άξονα.

Στη συνέχεια, οι τιμές  $\theta_1$  προστίθενται στις  $\theta_2$ , που είναι η εκτίμηση φάσης στον κάτω βρόχο. Η έξοδος του κάτω ταλαντωτή είναι: [1]

$$\begin{aligned} \sin(4\pi f t + 2\theta_1(t) + 2\theta_2(t)) &= \sin(4\pi f_c t + 4\pi(f_0 - f_c)t + 2b + 2\theta_2(t)) \\ &\rightarrow \sin(4\pi f_0 t + 2b + 2\theta_2(t)) \end{aligned}$$

Επιτυχώς, ο άνω βρόχος συνθέτει ένα σήμα το οποίο έχει τη σωστή συχνότητα για τον κάτω βρόχο, όπου  $\theta_2(t) \rightarrow \phi - b$ . Έτσι, δημιουργείται ένα ημιτονοειδές συγχρονισμένο τόσο σε φάση όσο και σε συχνότητα με το  $r_p(t)$ , το οποίο είναι κατάλληλο για να αποδιαμορφώσουμε το λαμβανόμενο σήμα.



Σχήμα 3.4 Διάταξη Διπλού Βρόχου Κλειδώματος Φάσης

### 3.2.ii Βρόχος Costas

Όπως είδαμε η μέθοδος PLL απαιτεί προεπεξεργασία του λαμβανόμενου σήματος, και πιο συγκεκριμένα τετραγωνισμό και στη συνέχεια ζωνοπερατό φιλτράρισμα, για να παράγει μια “καθαρή” εκδοχή του φορέα με διπλάσια συχνότητα και φάση. Μια εναλλακτική προσέγγιση λειτουργεί απευθείας πάνω στο λαμβανόμενο σήμα  $r(kT_s) = s(kT_s) \cos(2\pi f_0 kT_s + \phi)$  αντιστρέφοντας τη σειρά επεξεργασίας: Πρώτα αποδιαμορφώνοντας στη βασική ζώνη, κατόπιν εφαρμόζοντας χαμηλοπερατό φιλτράρισμα και τέλος τετραγωνισμό. Όλα αυτά

οδηγούν στην παρακάτω συνάρτηση απόδοσης: [10]

$$J_C(\theta) = \text{avg} \{ (LPF \{ r(kT_s) \cos(2\pi f_0 kT_s + \theta) \})^2 \}$$

Εξαιτίας του τρόπου με τον οποίο εισάγεται η μη-γραμμικότητα του τετραγωνισμού στη συνάρτησης απόδοσης, επιτρέπεται στον αλγόριθμο να λειτουργήσει χωρίς να απαιτείται προεπεξεργασία του λαμβανόμενου σήματος. Αντικαθιστώντας το ισοδύναμο του  $r(kT_s)$  στη συνάρτηση κόστους έχουμε την παρακάτω σχέση: [10]

$$J_C(\theta) = \text{avg} \{ (LPF \{ s(kT_s) \cos(2\pi f_0 kT_s + \phi) \cos(2\pi f_0 kT_s + \theta) \})^2 \}$$

Υποθέτοντας ότι η συχνότητα αποκοπής του χαμηλοπερατού φίλτρου είναι μεγαλύτερη από το εύρος ζώνης του  $s(kT_s)$  και ακολουθώντας την ίδια λογική όπως στην ανάλυση του PLL αλλά με  $\phi$  αντί για  $2\phi$ ,  $\theta$  στη θέση του  $2\theta$  και  $2\pi f_0 kT_s$  αντί για  $4\pi f_0 kT_s$  καταλήγουμε στην έκφραση: [10]

$$LPF \{ s(kT_s) \cos(2\pi f_0 kT_s + \phi) \cos(2\pi f_0 kT_s + \theta) \} = \frac{1}{2} s(kT_s) \cos(\phi - \theta)$$

Αντικαθιστώντας την παραπάνω σχέση, στη συνάρτηση απόδοσης προκύπτει: [10]

$$\begin{aligned} J_C(\theta) &= \text{avg} \left\{ \left( \frac{1}{2} s(kT_s) \cos(\phi - \theta) \right)^2 \right\} \\ &= \frac{1}{4} \text{avg} \{ s^2(kT_s) \cos^2(\phi - \theta) \} \\ &\approx \frac{1}{4} s_{avg}^2 \cos^2(\phi - \theta) \end{aligned}$$

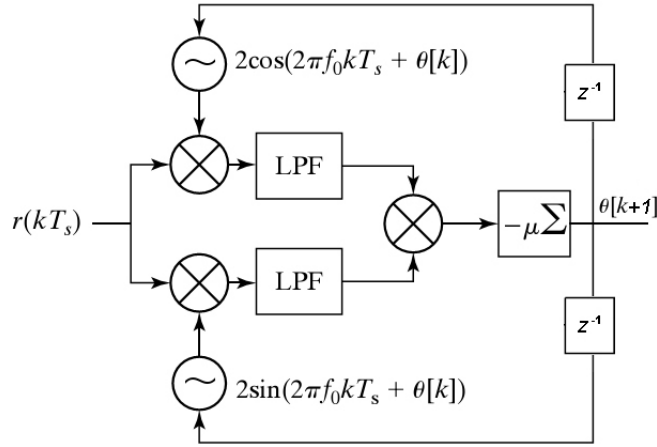
όπου  $s_{avg}^2$  είναι η μέση τιμή του τετραγώνου της ακολουθίας των δεδομένων  $s(kT_s)$ . Είναι προφανές ότι η συνάρτηση απόδοσης είναι ανάλογη του  $\cos^2(\phi - \theta)$  και μεγιστοποιείται για  $\theta = \phi + k\pi$ , για  $k$  ακέραιο. Ο βρόχος Costas μπορεί να υλοποιηθεί ως προσαρμοστικός αλγόριθμος. Η παράγωγος της συνάρτησης απόδοσης προσεγγίζεται ως εξής: [10]

$$\begin{aligned} \frac{dJ_C(\theta)}{d\theta} &\approx \text{avg} \left\{ \frac{d(LP F \{ r(kT_s) \cos(2\pi f_0 kT_s + \theta) \})^2}{d\theta} \right\} \\ &= 2 \text{avg} \left\{ LP F \{ r(kT_s) \cos(2\pi f_0 kT_s + \theta) \} \frac{dLP F \{ r(kT_s) \cos(2\pi f_0 kT_s + \theta) \}}{d\theta} \right\} \\ &\approx 2 \text{avg} \left\{ LP F \{ r(kT_s) \cos(2\pi f_0 kT_s + \theta) \} LP F \left\{ \frac{d(r(kT_s) \cos(2\pi f_0 kT_s + \theta))}{d\theta} \right\} \right\} \\ &= -2 \text{avg} \{ LP F \{ r(kT_s) \cos(2\pi f_0 kT_s + \theta) \} LP F \{ r(kT_s) \sin(2\pi f_0 kT_s + \theta) \} \} \end{aligned}$$

Επομένως, ο αντίστοιχος προσαρμοστικός αλγόριθμος, που είναι της μορφής πιο Απότομης Ανάβασης (*Steepest Ascent*) είναι: [10]

$$\begin{aligned}\theta[k+1] &= \theta[k] + \mu \left. \frac{dJ_C(\theta)}{d\theta} \right|_{\theta=\theta[k]} \\ &= \theta[k] - \mu \text{avg} \{ \text{LPF} \{ r(kT_s) \cos(2\pi f_0 k T_s + \theta[k]) \} \\ &\quad \text{LPF} \{ r(kT_s) \sin(2\pi f_0 k T_s + \theta[k]) \} \}\end{aligned}$$

Συνήθως, αγνοούμε την πράξη του *avg*. Στο σχήμα 3.5 φαίνεται μια διάταξη απλού βρόχου Costas.



Σχήμα 3.5 Μπλοκ Διάγραμμα Βρόχου Costas

Υπάρχουν δύο βασικά μονοπάτια. Το πάνω μονοπάτι διαμορφώνεται από ένα συνημίτονο και με χαμηλοπερατό φιλτράρισμα λαμβάνουμε το  $s(kT_s) \cos(\phi - \theta)$ , ενώ το κάτω διαμορφώνεται από ένα ημίτονο και με χαμηλοπερατό φιλτράρισμα λαμβάνουμε το  $-s(kT_s) \sin(\phi - \theta)$ . Ο συνδιασμός των δύο μας δίνει την εκτίμηση της φάσης. Αναλόγως με το PLL, μπορεί να χρησιμοποιηθεί διάταξη διπλού βρόχου Costas για την εκτίμηση τόσο της φάσης όσο και της συχνότητας του φέροντος. [1]

## 4. Ανάκτηση Συγχρονισμού

### 4.1 Γενικά

Το πρόβλημα της *Ανάκτησης Συγχρονισμού* είναι να επιλέξουμε τις χρονικές στιγμές στις οποίες θα δειγματοληπτήσουμε το λαμβανόμενο στο δέκτη αναλογικό σήμα. Αυτό μπορεί να μεταφραστεί ως ένα μαθηματικό πρόβλημα εύρεσης μιας απλής παραμέτρου, την απόκλιση συγχρονισμού  $\tau$ , η οποία ελαχιστοποιεί ή μεγιστοποιεί κάποια συνάρτηση κόστους ή απόδοσης αντίστοιχα [1]. Πριν αναλύσουμε λεπτομερώς την ανάκτηση συγχρονισμού, ακολουθεί μια ενότητα που αναφέρεται στους παλμούς Nyquist και άλλη μια που αναφέρεται στις μεθόδους ανάκτησης συγχρονισμού. Τέλος, ακολουθεί μια ενότητα που αναφέρεται στην έννοια της παρεμβολής.

Οι μέθοδοι ανάκτησης συγχρονισμού που αναλύονται είναι οι εξής:

- Ελαχιστοποίησης του Cluster Variance
- Μεγιστοποίησης της Ισχύος Εξόδου



## 4.2 Παλμός Nyquist

Όταν η πληροφορία είναι ψηφιακή, πρέπει να μετατραπεί σε ένα αναλογικό σήμα για να μεταδοθεί. Αυτή η μετατροπή γίνεται από το φίλτρο μετάδοσης, το οποίο μετατρέπει κάθε σύμβολο σε έναν αναλογικό παλμό. Μετά τη μετάδοση, το φίλτρο λήψης σκοπό έχει την ανάκτηση των ψηφιακών τιμών από τους λαμβανόμενους αναλογικούς παλμούς. Αν ο αναλογικός παλμός έχει διάρκεια μεγαλύτερη από το χρονικό διάστημα μεταξύ διαδοχικών συμβόλων, τότε στην έξοδο του φίλτρου λήψης διαδοχικά σύμβολα πιθανώς να επικαλύπτονται. Το πρόβλημα αυτό ονομάζεται διασυμβολική παρεμβολή (ISI). [2]

Μια κατηγορία παλμών που ελαχιστοποιούν τη διασυμβολική παρεμβολή είναι οι *παλμοί Nyquist*. Αυτοί έχουν τέτοιο σχήμα ώστε να δίνουν μονάδα τις χρονικές στιγμές  $kT$  και μηδέν στις  $mT$  για  $m \neq k$ , όπου  $T$  η χρονική διάρκεια συμβόλου και  $m, k$  ακέραιοι. [2]

$$h_{NYQ}(kT) = \begin{cases} 1, & k = 0 \\ 0, & k \neq 0 \end{cases}$$

Ένας παλμός Nyquist που είναι πολύ δημοφιλής είναι το ανυψωμένο συνημίτονο. Στην εργασία αυτή χρησιμοποιήθηκε τετραγωνική ρίζα ανυψωμένου συνημιτόνου (SRRC), το οποίο δεν είναι παλμός Nyquist, αλλά η συνέλιξη δύο τέτοιων παλμών στο πεδίο του χρόνου παράγει έναν παλμό Nyquist. Στο πεδίο του χρόνου, το SRRC περιγράφεται από την παρακάτω σχέση: [2]

$$h_{SRRC}(t) = \begin{cases} \frac{1}{\sqrt{T}} \frac{\sin(\pi(1-\beta)t/T) + (4\beta t/T) \cos(\pi(1+\beta)t/T)}{(\pi t/T)(1-(4\beta t/T)^2)} & , \quad t \neq 0, t \neq \pm \frac{T}{4\beta} \\ \frac{1}{\sqrt{T}} (1 - \beta + (4\beta/\pi)) & , \quad t = 0 \\ \frac{\beta}{\sqrt{2T}} \left[ \left(1 + \frac{2}{\pi}\right) \sin\left(\frac{\pi}{4\beta}\right) + \left(1 - \frac{2}{\pi}\right) \cos\left(\frac{\pi}{4\beta}\right) \right] & , \quad t = \pm \frac{T}{4\beta} \end{cases}$$

Στο πεδίο της συχνότητας, το SRRC περιγράφεται από την παρακάτω σχέση [2]

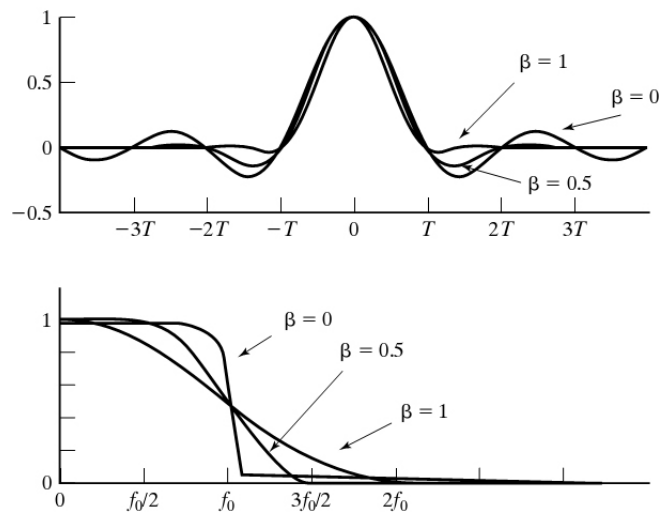
$$H_{SRRC}(f) = \begin{cases} 1 & , \quad |f| < f_1 \\ \left[ \frac{1}{2} \left( 1 + \cos \left[ \frac{\pi(|f| - f_1)}{2f_\Delta} \right] \right) \right]^{\frac{1}{2}} & , \quad f_1 < |f| < B \\ 0 & , \quad |f| > B \end{cases}$$

όπου

$B$  είναι το απόλυτο εύρος ζώνης,  
 $f_0$  είναι 6 dB εύρος ζώνης ίσο με το μισό του ρυθμού συμβόλων,  
 $f_\Delta = B - f_0$ ,  
 $f_1 = f_0 - f_\Delta$ .

Ορίζουμε τον συντελεστή επέκτασης  $\beta = \frac{f_{\Delta}}{f_0}$ . Στο σχήμα 4.1 δείχνουμε έναν χαρακτηριστικό παλμό Nyquist, που είναι το ανυψωμένο συνημίτονο (Raised Cosine), τόσο στο πεδίο του χρόνου όσο και της συχνότητας για διαφορετικούς συντελεστές επέκτασης. Τα κύρια χαρακτηριστικά του παλμού είναι τα εξής: [2]

- παρουσιάζει μηδενικά σε επιθυμητές χρονικές στιγμές
- στο πεδίο της συχνότητας οι ανεπιθύμητες ουρές εμφανίζονται με πολύ μικρότερα πλάτη σε σχέση με τον κεντρικό λωβό
- η περιβάλλουσα αποσβένει με ρυθμό  $\frac{1}{|t|^3}$  για μεγάλο  $t$ . Επίσης, ο συντελεστής επέκτασης  $\beta$  ανήκει στο διάστημα  $[0,1]$  και όσο αυτός αυξάνεται, στενεύει το εύρος του κεντρικού λωβού. Καθώς το  $\beta \rightarrow 0$ , η χρουστική απόκριση του παλμού γίνεται sinc.



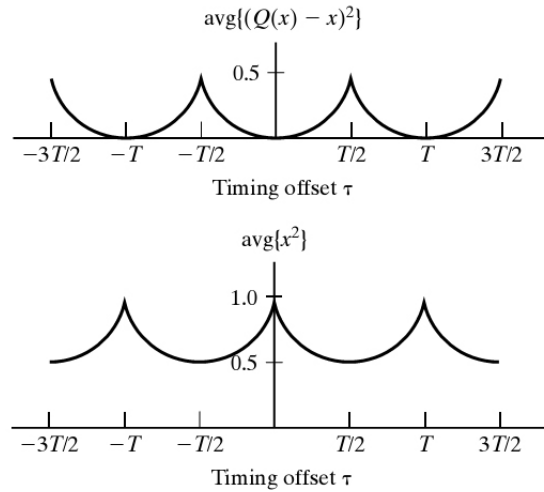
Σχήμα 4.1 RC Παλμός στο Πεδίο του Χρόνου και της Συχνότητας

### 4.3 Μέθοδοι Ανάκτησης Συγχρονισμού

Έχουμε 2 μετρικές της ποιότητας της παραμέτρου  $\tau$ , οι οποίες παρέχουν σημαντική πληροφορία σχετικά με τα σημεία δειγματοληψίας της λαμβανόμενης κυματομορφής, διότι τα ελάχιστα ή τα μέγιστα τους συμπίπτουν με τις βέλτιστες στιγμές δειγματοληψίας της κυματομορφής.

- Cluster Variance που ορίζεται η ποσότητα  $\text{avg}\{(Q(x[k]) - x[k])^2\}$ , όπου η  $Q(\cdot)$  κβαντίζει το όρισμά της στο πλησιέστερο σύμβολο του αστερισμού που χρησιμοποιείται
- Output Power που ορίζεται η ποσότητα  $\text{avg}\{x[k]^2\}$

Παραδείγματα της μορφής των συναρτήσεων αυτών φαίνονται στο σχήμα 4.2.



Σχήμα 4.2 Καμπύλες μετρικών συναρτήσεων της παραμέτρου  $\tau$

Το σήμα εισόδου σε αυτή τη βαθμίδα είναι [1]

$$x(t) = \sum_{i=-\infty}^{\infty} a_i \delta(t - iT) * g_T(t) * c(t) * g_R(t) + w(t) * g_R(t)$$

όπου  $g_T(t)$  είναι ο παλμός μορφοποίησης στον πομπό,  $g_R(t)$  ο παλμός μορφοποίησης στον δέκτη,  $c(t)$  η κρουστική απόκριση του καναλιού,  $a_i$  τα 4-PAM σύμβολα και  $w(t)$  ο θόρυβος.

Συνδιάζοντας τα 3 γραμμικά φίλτρα παίρνουμε

$$h(t) = g_T(t) * c(t) * g_R(t)$$

Έτσι, έχουμε [1]

$$x(t) = \sum_k a_k h(t - kT) + w'(t)$$

με  $w'(t) = w(t) * g_R(t)$ . Υπόθετουμε ότι ο θόρυβος είναι λευκός, άρα μια στάσιμη τυχαία διαδικασία, επομένως η ισχύς του θορύβου είναι ανεξάρτητη της χρονικής στιγμής. Σκοπός μας είναι η εύρεση των χρονικών στιγμών  $kT + \tau T$ , για τις οποίες το [1]

$$x[k] = x(kT + \tau T) = \sum_{i=-\infty}^{+\infty} a[i] h(kT + \tau T - iT) + w'[k]$$

να είναι “βέλτιστο” υπό κάποια έννοια.

### 4.3.i Ελαχιστοποίηση του Cluster Variance

Όπως αναφέρθηκε και νωρίτερα, στόχος μας είναι να βρούμε την τιμή της παραμέτρου  $\tau$ , η οποία ελαχιστοποιεί τη συνάρτηση κόστους [1]

$$J_{CV}(\tau) = \text{avg} \{ (Q(x[k]) - x[k])^2 \}$$

η οποία μπορεί να χρησιμοποιηθεί ως βάση για τη δημιουργία του παρακάτω προσαρμοστικού αλγορίθμου, που είναι της μορφής *πιο Απότομης Κατάβασης* (*Steepest Descent*). Είναι: [1]

$$\tau[k+1] = \tau[k] - \mu \left. \frac{dJ_{CV}(\tau)}{d\tau} \right|_{\tau=\tau[k]}$$

Για μικρό  $\mu$  ισχύει η προσέγγιση [1]

$$\frac{dJ_{CV}(\tau)}{d\tau} \approx \text{avg} \left\{ \frac{d(Q(x[k]) - x[k])^2}{d\tau} \right\} = -2 \text{avg} \left\{ (Q(x[k]) - x[k]) \frac{dx[k]}{d\tau} \right\}$$

Ένας τρόπος να προσεγγιστεί η παράγωγος είναι [1]

$$\frac{dx[k]}{d\tau} = \frac{dx(kT + \tau T)}{d\tau} \approx \frac{x(kT + \tau T + \delta T) - x(kT + \tau T - \delta T)}{2\delta T}$$

το οποίο είναι μια καλή προσέγγιση για μικρό  $\delta$ .

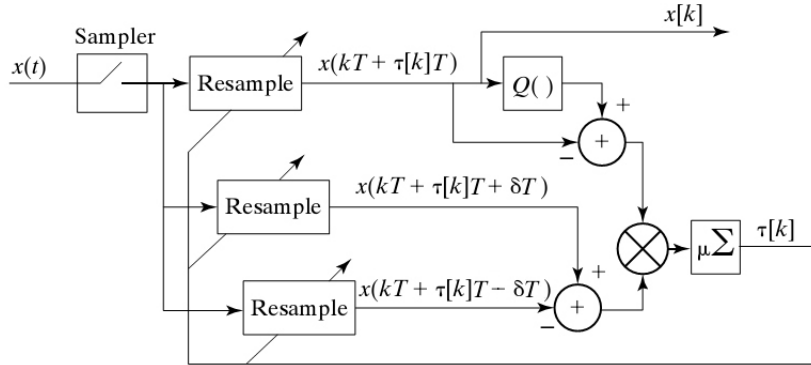
Αντικαθιστώντας την παραπάνω προσέγγιση στον προσαρμοστικό αλγόριθμο, έχουμε

$$\begin{aligned} \tau[k+1] &= \tau[k] \\ &+ \bar{\mu} \text{avg} \{ (Q(x[k]) - x[k]) [x(kT + \tau[k]T + \delta T) - x(kT + \tau[k]T - \delta T)] \} \end{aligned}$$

όπου  $\bar{\mu} = \frac{\mu}{\delta}$ . Συνήθως παραλείπεται ο τελεστής *avg* και ο αλγόριθμος είναι ο εξής [1]

$$\tau[k+1] = \tau[k] + \bar{\mu} (Q(x[k]) - x[k]) [x(kT + \tau[k]T + \delta T) - x(kT + \tau[k]T - \delta T)]$$

Οι τιμές  $x(kT + \tau[k]T + \delta T)$  και  $x(kT + \tau[k]T - \delta T)$  μπορούν να υπολογιστούν από τα δείγματα που έχουμε στη διάθεσή μας μέσω παρεμβολής, όπως θα αναλυθεί στην επόμενη ενότητα. Αν το  $\tau[k]$  είναι αρκετά θορυβώδες, τότε το  $\mu$  μπορεί να μειωθεί, ωστόσο αυτό μπορεί να οδηγήσει σε αργή σύγκλιση του αλγορίθμου. Τέλος, ακολουθεί μια σχηματική αναπαράσταση του αλγορίθμου, όπως φαίνεται στο σχήμα 4.3.



**Σχήμα 4.3** Σχηματική αναπαράσταση του αλγορίθμου βασισμένου στην ελαχιστοποίηση του Cluster Variance

### 4.3.ii Μεγιστοποίηση της Ισχύος Εξόδου

Στόχος μας είναι να βρούμε την τιμή της παραμέτρου  $\tau$ , η οποία μεγιστοποιεί τη συνάρτηση [1]

$$J_{OP}(\tau) = \text{avg} \{x^2[k]\} = \text{avg} \{x^2(kT + \tau T)\}$$

και μπορεί να χρησιμοποιηθεί ως βάση για τη δημιουργία του παρακάτω προσαρμοστικού αλγορίθμου, που είναι της μορφής *πιο Απότομης Ανάβασης (Steepest Ascent)*. Είναι: [1]

$$\tau[k+1] = \tau[k] + \mu \left. \frac{dJ_{OP}(\tau)}{d\tau} \right|_{\tau=\tau[k]}$$

Για μικρό  $\mu$  ισχύει η προσέγγιση [1]

$$\frac{dJ_{OP}(\tau)}{d\tau} \approx \text{avg} \left\{ \frac{dx^2[k]}{d\tau} \right\} = 2 \text{avg} \left\{ x[k] \frac{dx[k]}{d\tau} \right\}$$

Ένας τρόπος για να προσεγγιστεί η παράγωγος είναι [1]

$$\frac{dx[k]}{d\tau} = \frac{dx(kT + \tau T)}{d\tau} \approx \frac{x(kT + \tau T + \delta T) - x(kT + \tau T - \delta T)}{2\delta T}$$

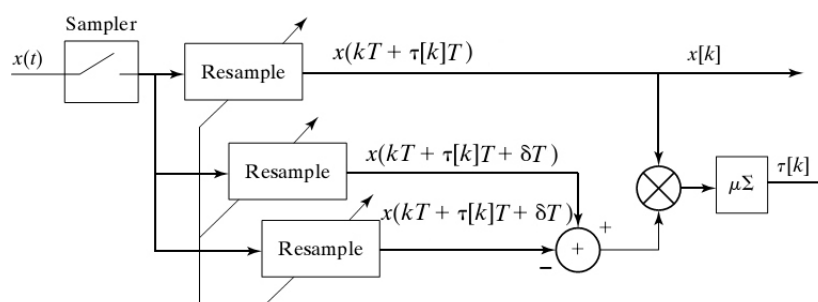
το οποίο είναι μια καλή προσέγγιση για μικρό  $\delta$ . Αντικαθιστώντας την παραπάνω προσέγγιση στον προσαρμοστικό αλγόριθμο, έχουμε

$$\tau[k+1] = \tau[k] + \bar{\mu} \text{avg} \{x[k] [x(kT + \tau[k] T + \delta T) - x(kT + \tau[k] T - \delta T)]\}$$

όπου  $\bar{\mu} = \frac{\mu}{\delta}$ . Συνήθως παραλείπεται ο τελεστής  $\text{avg}$  και ο αλγόριθμος είναι ο εξής [1]

$$\tau[k+1] = \tau[k] + \bar{\mu} x[k] [x(kT + \tau[k] T + \delta T) - x(kT + \tau[k] T - \delta T)]$$

Οι τιμές  $x(kT + \tau[k] T + \delta T)$  και  $x(kT + \tau[k] T - \delta T)$  μπορούν να υπολογιστούν από τα δείγματα που έχουμε στη διάθεσή μας μέσω παρεμβολής, όπως θα αναλυθεί στην επόμενη ενότητα. Αν το  $\tau[k]$  είναι αρκετά θορυβώδες, τότε το  $\mu$  μπορεί να μειωθεί, ωστόσο αυτό μπορεί να οδηγήσει σε αργή σύγκλιση του αλγορίθμου. Τέλος, ακολουθεί μια σχηματική αναπαράσταση του αλγορίθμου, όπως φαίνεται στο σχήμα 4.4.



**Σχήμα 4.4** Σχηματική αναπαράσταση του αλγορίθμου βασισμένου στην μεγιστοποίηση της Ισχύος Εξόδου

## 4.4 Παρεμβολή<sup>2</sup>

Ολόκληρη η αναλογική κυματομορφή μπορεί να ανακτηθεί από τις διακριτές τιμές της όταν η δειγματοληψία γίνεται με ρυθμό μεγαλύτερο από το ρυθμό Nyquist, όπως έχει αναφερθεί ήδη. Όταν σκοπός μας είναι να βρούμε ολόκληρη την κυματομορφή, η διαδικασία ονομάζεται ανακατασκευή, ενώ όταν σκοπός μας είναι να βρούμε τιμές της κυματομορφής σε σημεία ανάμεσα στις στιγμές δειγματοληψίας, η διαδικασία ονομάζεται *Παρεμβολή*. [1]

Έστω το λαμβανόμενο σήμα

$$x(t) = \sum_i a_i g_T(t - iT - \tau T)$$

όπου  $g_T(t)$  το φίλτρο μετάδοσης που είναι παλμός Nyquist και  $a_i$  η ακολουθία δεδομένων εισόδου. Θέλουμε να υπολογίσουμε το

$$x(kT + \tau T) = \sum_i a_i g_T((k - i)T) = a_k$$

από την ακολουθία  $\{x[n]\} = \{x(nT_s)\}$ . Από το θεώρημα της δειγματοληψίας, έχουμε

$$x(t) = \sum_n x(nT_s) h_I(t - nT_s)$$

με

$$h_I(t) = \text{sinc}\left(\frac{\pi t}{T_s}\right)$$

Για  $t_n = nT + \hat{\tau}^{(n)}T$ ,

$$\begin{aligned} x(t_n) &= \sum_m x(mT_s) h_I(t_n - mT_s) \\ &= \sum_m x(mT_s) h_I(nT + \hat{\tau}^{(n)}T - mT_s) \end{aligned} \quad (1)$$

Στη συνέχεια ορίζοντας

$$l_n = \text{int}\left(\frac{t_n}{T_s}\right), \mu_n = \text{frc}\left(\frac{t_n}{T_s}\right)$$

έχουμε ότι

$$t_n = (l_n + \mu_n)T_s$$

---

<sup>2</sup>Λιάβας, Αθανάσιος. Πανεπιστημιακές Σημειώσεις Μαθήματος “Ψηφιακά Τηλεπικοινωνιακά Συστήματα ΙΙ”



και η (1) γίνεται

$$\begin{aligned}
 x(t_n) &= \sum_m x(mT_s) h_I((l_n - m)T_s + \mu_n T_s) \\
 &\stackrel{k=l_n-m}{=} \sum_k h_I(kT_s + \mu_n T_s) x((l_n - k)T_s) \\
 &= \sum_m h_I(mT_s + \mu_n T_s) x((l_n - m)T_s) \quad (2)
 \end{aligned}$$

Αν χρησιμοποιήσουμε πεπερασμένο πλήθος δειγμάτων για την παρεμβολή, έστω  $2M + 1$ , τότε

$$x(t_n) \approx \sum_{m=-M}^M h_I(mT_s + \mu_n T_s) x((l_n - m)T_s)$$

Αν ορίσουμε

$$x[m] = x(mT_s), h_I^n[m] = h_I(mT_s + \mu_n T_s) = \frac{\sin\left(\frac{\pi m T_s + \pi \mu_n T_s}{T_s}\right)}{\left(\frac{\pi m T_s + \pi \mu_n T_s}{T_s}\right)} = \frac{\sin(\pi(m + \mu_n))}{\pi(m + \mu_n)}$$

τότε

$$x(t_n) \approx \sum_{m=-M}^M h_I^n[m] x[l_n - m]$$

Έτσι, από τα δείγματα που έχουμε στη διάθεσή μας, μπορούν να υπολογιστούν οι τιμές  $x(kT + \tau[k]T + \delta T)$  και  $x(kT + \tau[k]T - \delta T)$  για τους δύο αλγορίθμους που αναλύθηκαν στις προηγούμενες ενότητες.

## 5. Εκτίμηση και Ισοστάθμιση Καναλιού

### 5.1 Γενικά

Σε προηγούμενη ενότητα, είδαμε πως για τον μηδενισμό της διασυμβολικής παρεμβολής στις χρονικές στιγμές δειγματοληψίας, όπου το κανάλι είναι γνωστό στο δέκτη και τα χαρακτηριστικά του δεν μεταβάλλονται με το χρόνο, στόχος μας ήταν ο κατάλληλος σχεδιασμός του φίλτρου λήψης. Ωστόσο, στην πράξη συναντούμε κανάλια που είναι άγνωστα ή μεταβάλλονται με το χρόνο. Για παράδειγμα, στην περίπτωση ενός τηλεφωνικού δικτύου, το κανάλι επικοινωνίας θα είναι διαφορετικό κάθε φορά που επιλέγουμε έναν αριθμό, επειδή η διαδρομή του καναλιού θα είναι διαφορετική. Από τη στιγμή όμως που θα επιτευχθεί η σύνδεση, το κανάλι θα είναι χρονικά αμετάβλητο για ένα σχετικά μεγάλο χρονικό διάστημα. Αυτό είναι ένα παράδειγμα καναλιού, του οποίου τα χαρακτηριστικά δεν είναι εκ των προτέρων γνωστά. Σε αυτή την κατηγορία ανήκουν τα ραδιοκανάλια και γενικότερα τα κανάλια πολλαπλών διαδρομών. Όταν δεν υπάρχει διασυμβολική παρεμβολή, η σύνθετη κρουστική απόκριση του συστήματος (φίλτρο εκπομπής + κανάλι + φίλτρο λήψης) θα αποτελείται από μια μόνο μη μηδενική τιμή. Η τιμή αυτή εξαρτάται από την ποιότητα μετάδοσης και η καθυστέρηση που εισάγεται καθορίζεται από το χρόνο μετάδοσης. Όμως, όταν υπάρχει διασυμβολική παρεμβολή, ο αριθμός των μη μηδενικών τιμών της σύνθετης κρουστικής απόκρισης αυξάνεται. Το κανάλι μπορεί να μοντελοποιηθεί σαν ένα φίλτρο πεπερασμένης κρουστικής απόκρισης (FIR). Στόχος της χρήσης ισοσταθμιστή, είναι να δημιουργηθεί ένα φίλτρο στο δέκτη ώστε αυτό να αναιρέσει την επίδραση του καναλιού. Δηλαδή, είναι επιθυμητό η σύνθετη κρουστική απόκριση καναλιού - ισοσταθμιστή να έχει μια μόνο μη μηδενική τιμή. [2],[1]

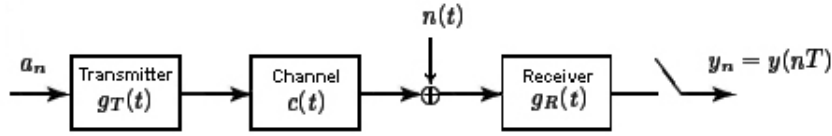
Έστω ότι έχουμε ένα σύστημα όπως αυτό του σχήματος 5.1, όπου  $g_T(t)$  το φίλτρο μορφοποίησης στον πομπό,  $c(t)$  το κανάλι μετάδοσης και  $g_R(t)$  το φίλτρο μορφοποίησης στο δέκτη. Η σύνθετη κρουστική απόκριση δίνεται από τη σχέση [10]

$$h^0(t) = g_T(t) * c(t) * g_R(t)$$

Τότε, εξαιτίας της παραμόρφωσης του καναλιού, η έξοδος του φίλτρου λήψης είναι

$$y(t) = \sum_{n=-\infty}^{+\infty} a_n h^0(t - nT) + v(t)$$

όπου  $a_n$  η ακολουθία εισόδου και  $v(t) = n(t) * g_R(t)$  η συνιστώσα θορύβου στην έξοδο του φίλτρου λήψης.



Σχήμα 5.1 Σύνθετο Κανάλι Συνεχούς Χρόνου

Η έξοδος του φίλτρου μπορεί να δειγματοληπτηθεί περιοδικά, δημιουργώντας την έξοδο του ισοδύναμου καναλιού διακριτού χρόνου, παράγοντας την ακολουθία [10]

$$y_m = \sum_{n=-\infty}^{+\infty} a_n h_{m-n}^0 + v_m = h_0^0 a_m + \underbrace{\sum_{\substack{n=-\infty \\ n \neq m}}^{+\infty} a_n h_{m-n}^0}_{ISI} + v_m$$

όπου  $h_n^0 = h^0(nT)$  και  $v_n = v(nT)$ .

Συνήθως, στην πράξη υποθέτουμε ότι η κρουστική απόκριση είναι μια πεπερασμένη και αιτιατή ακολουθία και συνεπώς παίρνει τις μη μηδενικές τιμές  $h_0^0, h_1^0, \dots, h_M^0$ . Το γραμμικό χρονικά αμετάβλητο σύστημα με κρουστική απόκριση  $\{h_n^0\}_{n=0}^M$  καλείται ισοδύναμο κανάλι διακριτού χρόνου και η σχέση εισόδου-εξόδου του δίνεται από την σχέση [10]

$$y_m = \sum_{n=0}^M h_n^0 a_{m-n} + v_m$$

Έστω ότι η επικοινωνία πομπού-δέκτη γίνεται με πακέτα δεδομένων. Κάθε πακέτο που αποστέλλεται έχει την εξής μορφή

$$a_1 \dots a_{n_1-1} \quad \underbrace{a_{n_1} \dots a_{n_2}}_{\text{training symbols}} \quad a_{n_2+1} \dots a_N$$

όπου τα σύμβολα που χαρακτηρίζονται “*training symbols*” είναι γνωστά στο δέκτη (δεν περιέχουν πληροφορία) και χρησιμοποιούνται για συγχρονισμό και εκτίμηση/ισοστάθμιση καναλιού.

Στη συνέχεια θα ασχοληθούμε με

- Γραμμική Ισοστάθμιση
  - Ελαχίστων Τετραγώνων
  - LMS
- Μη Γραμμική Ισοστάθμιση
  - Αλγόριθμος Viterbi

## 5.2 Γραμμική Ισοστάθμιση

### 5.2.i Γενικά

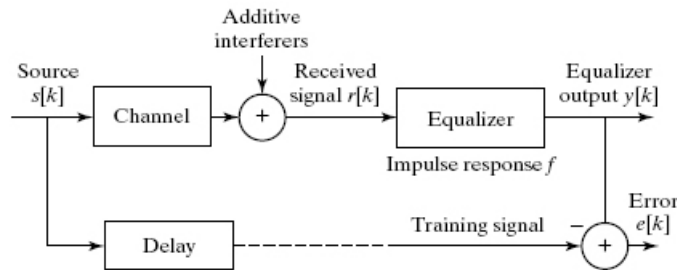
Το πρόβλημα της γραμμικής ισοστάθμισης είναι να υπολογίσουμε ένα γραμμικό φίλτρο πεπερασμένης χρονικής απόκρισης  $f_i$ , όπως φαίνεται στο σχήμα 5.3, το οποίο θα αναιρέσει την επίδραση του καναλιού και παράλληλα θα ελαχιστοποιήσει άλλες παρεμβολές. Έτσι, στην έξοδο του φίλτρου θα πάρουμε προσεγγιστικά μια καθυστερημένη εκδοχή της ακολουθίας εισόδου. Δηλαδή

$$y[k] \approx s[k - \delta]$$

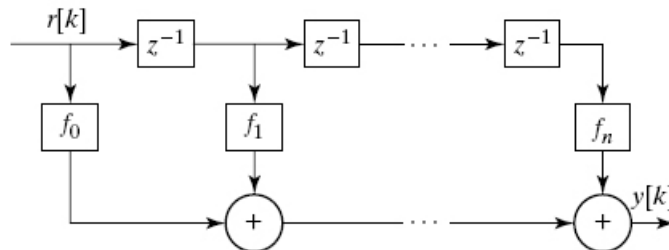
για συγκεκριμένο  $\delta$ , όπου  $s[k]$  είναι η ακολουθία εισόδου. Αν θεωρήσουμε ότι το λαμβανόμενο σήμα στο δέκτη είναι το  $r[k]$  όπως φαίνεται στο σχήμα 5.2, η έξοδος του FIR γραμμικού ισοσταθμιστή μπορεί να περιγραφεί από την παρακάτω σχέση [1]

$$y[k] = \sum_{j=0}^n f_j r[k - j] \quad (1)$$

με  $f_j, j = 0, 1, \dots, n$  να είναι οι συντελεστές του φίλτρου του ισοσταθμιστή (χρονική απόκριση), όπως περιγράφεται σχηματικά στο σχήμα 5.3. Τονίζουμε ότι το  $j$  δεν μπορεί να είναι μικρότερο του μηδενός διαφορετικά ο ισοσταθμιστής είναι μη αιτιατός.



Σχήμα 5.2 Αναπαράσταση Γραμμικής Ισοστάθμισης



Σχήμα 5.3 Φίλτρο Ισοσταθμιστή

Για να υπολογίσουμε την έξοδο του ισοσταθμιστή την χρονική στιγμή  $n+1$ , χρειαζόμαστε τα  $n+1$  πιο πρόσφατα δείγματα, δηλαδή

$$y[n+1] = [r[n+1] \dots r[1]] \begin{bmatrix} f_0 \\ \vdots \\ f_n \end{bmatrix}$$

Αν θέλουμε να υπολογίσουμε την έξοδο του ισοσταθμιστή σε ένα παράθυρο χρονικών στιγμών από  $n+1$  ως  $p$  τότε έχουμε [1]

$$\begin{bmatrix} y[n+1] \\ y[n+2] \\ y[n+3] \\ \vdots \\ y[p] \end{bmatrix} = \begin{bmatrix} r[n+1] & r[n] & \dots & r[1] \\ r[n+2] & r[n+1] & \dots & r[2] \\ r[n+3] & r[n+2] & \dots & r[3] \\ \vdots & \vdots & \dots & \vdots \\ r[p] & r[p-1] & \dots & r[p-n] \end{bmatrix} \begin{bmatrix} f_0 \\ f_1 \\ \vdots \\ f_n \end{bmatrix}$$

Με τη μορφή πινάκων έχουμε την παρακάτω σχέση

$$\mathbf{Y} = \mathbf{R}\mathbf{F} \quad (2)$$

όπου ο πίνακας  $\mathbf{R}$  είναι πίνακας *Toeplitz*.

Η έξοδος του ισοσταθμιστή θέλουμε να προσεγγίζει όσο το δυνατόν καλύτερα μια καθυστερημένη εκδοχή της εισόδου. Το λάθος εκτίμησης είναι

$$e[k] = s[k-\delta] - y[k] \quad (3)$$

για συγκεκριμένη καθυστέρηση  $\delta$ .

Ορίζουμε τον πίνακα της ακολουθίας εισόδου και του λάθους αντίστοιχα [1]

$$S = \begin{bmatrix} s[n+1-\delta] \\ s[n+2-\delta] \\ s[n+3-\delta] \\ \vdots \\ s[p-\delta] \end{bmatrix}$$

$$E = \begin{bmatrix} e[n+1] \\ e[n+2] \\ e[n+3] \\ \vdots \\ e[p] \end{bmatrix}$$

Χρησιμοποιώντας την σχέση (3) και γράφοντάς την με μορφή πινάκων έχουμε

$$\mathbf{E} = \mathbf{S} - \mathbf{Y} = \mathbf{S} - \mathbf{R}\mathbf{F} \quad (4)$$

## 5.2.ii Γραμμική Ισοστάθμιση Ελαχίστων Τετραγώνων

Σε αυτή την ενότητα θα δείξουμε πως το λάθος εκτίμησης μπορεί να χρησιμοποιηθεί για να ορίσουμε μια συνάρτηση κόστους, η οποία εξαρτάται από τις άγνωστες παραμέτρους  $f_i$ . Υπολογίζοντας τις παραμέτρους αυτές ώστε να ελαχιστοποιήσουμε τη συνάρτηση κόστους, τελικά έχουμε μια ικανοποιητική λύση στο πρόβλημα της ισοστάθμισης. Ορίζουμε σαν συνάρτηση κόστους [1]

$$J_{LS} = \sum_{i=n+1}^p e^2[i]$$

Από τον ορισμό του πίνακα  $\mathbf{E}$  και από τη σχέση (4), η συνάρτηση κόστους μπορεί να γραφεί ως [1]

$$\begin{aligned} J_{LS} &= \mathbf{E}^T \mathbf{E} = (\mathbf{S} - \mathbf{R}\mathbf{F})^T (\mathbf{S} - \mathbf{R}\mathbf{F}) \\ &= \mathbf{S}^T \mathbf{S} - (\mathbf{R}\mathbf{F})^T \mathbf{S} - \mathbf{S}^T \mathbf{R}\mathbf{F} + (\mathbf{R}\mathbf{F})^T \mathbf{R}\mathbf{F} \end{aligned}$$

Επειδή, η συνάρτηση κόστους είναι βαθμωτή,  $(\mathbf{R}\mathbf{F})^T \mathbf{S}$  και  $\mathbf{S}^T \mathbf{R}\mathbf{F}$  είναι επίσης βαθμωτές. Καθώς, ο ανάστροφος μιας βαθμωτής είναι ο εαυτός της, είναι

$$\mathbf{F}^T \mathbf{R}^T \mathbf{S} = (\mathbf{R}\mathbf{F})^T \mathbf{S} = \mathbf{S}^T \mathbf{R}\mathbf{F} \quad (5)$$

Έτσι, η παραπάνω σχέση γράφεται ως

$$J_{LS} = \mathbf{S}^T \mathbf{S} - 2\mathbf{S}^T \mathbf{R}\mathbf{F} + (\mathbf{R}\mathbf{F})^T \mathbf{R}\mathbf{F} = \mathbf{S}^T \mathbf{S} - 2\mathbf{S}^T \mathbf{R}\mathbf{F} + \mathbf{F}^T (\mathbf{R}^T \mathbf{R}) \mathbf{F} \quad (6)$$

Στόχος μας είναι να υπολογίσουμε τον πίνακα  $\mathbf{F}$ , έτσι ώστε να ελαχιστοποιήσουμε τη συνάρτηση κόστους. Ορίζουμε τον πίνακα [1]

$$\begin{aligned} \mathbf{A} &= [\mathbf{F} - (\mathbf{R}^T \mathbf{R})^{-1} \mathbf{R}^T \mathbf{S}]^T (\mathbf{R}^T \mathbf{R}) [\mathbf{F} - (\mathbf{R}^T \mathbf{R})^{-1} \mathbf{R}^T \mathbf{S}] \\ &= \mathbf{F}^T (\mathbf{R}^T \mathbf{R}) \mathbf{F} - \mathbf{S}^T \mathbf{R}\mathbf{F} - \mathbf{F}^T \mathbf{R}^T \mathbf{S} + \mathbf{S}^T \mathbf{R} (\mathbf{R}^T \mathbf{R})^{-1} \mathbf{R}^T \mathbf{S} \\ &\stackrel{(5)}{=} \mathbf{F}^T (\mathbf{R}^T \mathbf{R}) \mathbf{F} - 2\mathbf{S}^T \mathbf{R}\mathbf{F} + \mathbf{S}^T \mathbf{R} (\mathbf{R}^T \mathbf{R})^{-1} \mathbf{R}^T \mathbf{S} \\ &\Rightarrow -2\mathbf{S}^T \mathbf{R}\mathbf{F} + \mathbf{F}^T (\mathbf{R}^T \mathbf{R}) \mathbf{F} = \mathbf{A} - \mathbf{S}^T \mathbf{R} (\mathbf{R}^T \mathbf{R})^{-1} \mathbf{R}^T \mathbf{S} \end{aligned}$$

Τότε, αντικαθιστώντας την παραπάνω σχέση στην (6) έχουμε

$$\begin{aligned} J_{LS} &= \mathbf{A} + \mathbf{S}^T \mathbf{S} - \mathbf{S}^T \mathbf{R} (\mathbf{R}^T \mathbf{R})^{-1} \mathbf{R}^T \mathbf{S} \\ &= \mathbf{A} + \mathbf{S}^T [\mathbf{I} - \mathbf{R} (\mathbf{R}^T \mathbf{R})^{-1} \mathbf{R}^T] \mathbf{S} \end{aligned}$$

Καθώς, η πόσότητα  $\mathbf{S}^T [\mathbf{I} - \mathbf{R} (\mathbf{R}^T \mathbf{R})^{-1} \mathbf{R}^T] \mathbf{S}$  δεν είναι συνάρτηση του  $\mathbf{F}$ , το ελάχιστο του  $J_{LS}$  επιτυγχάνεται εκεί όπου το  $\mathbf{F}$  ελαχιστοποιεί το  $\mathbf{A}$ . Αυτό συμβαίνει όταν

$$\mathbf{F} = (\mathbf{R}^T \mathbf{R})^{-1} \mathbf{R}^T \mathbf{S}$$

υποθέτοντας ότι ο πίνακας  $(\mathbf{R}^T \mathbf{R})^{-1}$  υπάρχει.

Άρα, το ελάχιστο επιτυγχάνεται από την παρακάτω σχέση

$$J_{LS}^{min} = \mathbf{S}^T [\mathbf{I} - \mathbf{R}(\mathbf{R}^T \mathbf{R})^{-1} \mathbf{R}^T] \mathbf{S}$$

Για να ολοκληρωθεί η διαδικασία, θα πρέπει να επιλέξουμε την βέλτιστη καθυστέρηση  $\delta$ . Μια προσέγγιση είναι να κάνουμε μια σειρά από υπολογισμούς  $\mathbf{S} = \mathbf{R}\mathbf{F}$  για κάθε τιμή της καθυστέρησης  $\delta$ , κατόπιν να υπολογίσουμε το αντίστοιχο  $J_{LS}^{min}$  και να διαλέξουμε την καθυστέρηση που αντιστοιχεί στην μικρότερη τιμή του. Τελικά, η εξίσωση [1]

$$\mathbf{F} = (\mathbf{R}^T \mathbf{R})^{-1} \mathbf{R}^T \mathbf{S}$$

κι εφόσον ο πίνακας  $(\mathbf{R}^T \mathbf{R})^{-1}$  υπάρχει, μας δίνει τη βέλτιστη λύση εύρεσης των συντελεστών  $f_i$ , με  $i = 0, \dots, n$  που μπορούν να χρησιμοποιηθούν στο φίλτρο του ισοσταθμιστή, βασιζόμενοι στη Λύση Ελαχίστων Τετραγώνων.



### 5.2.iii Προσαρμοστική Ισοστάθμιση - Αλγόριθμος LMS

Στην προηγούμενη ενότητα είδαμε ότι για να υπολογίσουμε τον βέλτιστο ισοσταθμιστή χρειάστηκε να κάνουμε μια σειρά από υπολογισμούς πινάκων όταν η καθυστέρηση που εισάγεται στο σύστημα είναι γνωστή. Αυτό προϋποθέτει τον υπολογισμό του αντίστροφου ενός  $(N+1) \times (N+1)$  πίνακα, όπου  $N$  είναι η μεγαλύτερη καθυστέρηση στον FIR γραμμικό ισοσταθμιστή. Από την άλλη, σε αυτή την ενότητα θα χρησιμοποιήσουμε έναν προσαρμοστικό αλγόριθμο προκειμένου να ελαχιστοποιήσουμε το μέσο τετραγωνικό σφάλμα (MSE) και τελικά να υπολογίσουμε τους συντελεστές του FIR φίλτρου του ισοσταθμιστή. Ως συνάρτηση κόστους που θα ελαχιστοποιεί το παραπάνω σφάλμα, ορίζουμε την [1]

$$J_{LMS} = \frac{1}{2} \text{avg} \{e^2[k]\} \quad (7)$$

Από (1),(3) έχουμε

$$e[k] = s[k - \delta] - \sum_{j=0}^n f_j r[k - j] \quad (8)$$

όπου  $r[k]$  είναι το λαμβανόμενο σήμα στη βασική ζώνη μετά τη δειγματοληψία. Ένας αλγόριθμος ο οποίος ελαχιστοποιεί την  $J_{LMS}$  είναι ο παρακάτω [1]

$$f_i[k+1] = f_i[k] - \mu \left. \frac{dJ_{LMS}}{df_i} \right|_{f_i=f_i[k]} \quad (9)$$

Κατόπιν θα πρέπει να υπολογιστεί η παράγωγος που εμφανίζεται στην σχέση (9). Έτσι, έχουμε [1]

$$\frac{dJ_{LMS}}{df_i} = \frac{d \text{avg} \left\{ \frac{1}{2} e^2[k] \right\}}{df_i} \approx \text{avg} \left\{ \frac{\frac{1}{2} de^2[k]}{df_i} \right\} = \text{avg} \left\{ e[k] \frac{de[k]}{df_i} \right\} \quad (10)$$

Χρησιμοποιώντας τη σχέση (8) στον υπολογισμό της παραγώγου έχουμε

$$\frac{de[k]}{df_i} = \frac{ds[k - \delta]}{df_i} - \sum_{j=0}^n \frac{d(f_j r[k - j])}{df_i} = -r[k - i] \quad (11)$$

όπου  $\frac{ds[k - \delta]}{df_i} = 0$  και  $\frac{d(f_j r[k - j])}{df_i} = 0, \forall i \neq j$ .

Αντικαθιστώντας την σχέση (11) στην (10) έχουμε

$$\frac{dJ_{LMS}}{df_i} = -\text{avg} \{e[k] r[k - i]\}$$

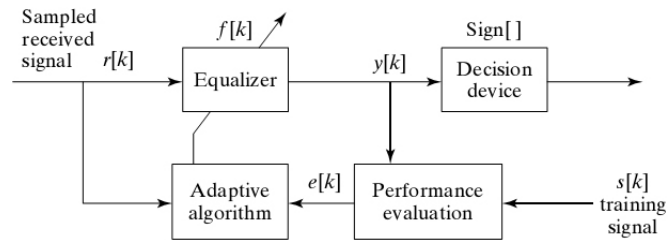
Αντικαθιστώντας την παραπάνω σχέση στην (9) παίρνουμε τον παρακάτω προσαρμοστικό αλγόριθμο, που είναι της μορφής πιο Απότομης Κατάβασης (*Steepest Descent*). Είναι: [1]

$$f_i[k+1] = f_i[k] + \mu \text{avg}\{e[k] r[k-i]\}$$

Συνήθως παραλείπεται ο τελεστής *avg*, καθώς η επανάληψη με ένα μικρό  $\mu$  έχει lowpass (averaging) χαρακτηριστικά. Το αποτέλεσμα που προκύπτει καλείται Προσαρμοστικός Αλγόριθμος *LMS* για γραμμική ισοστάθμιση και είναι [1]

$$f_i[k+1] = f_i[k] + \mu e[k] r[k-i]$$

Η προσαρμοστική ισοστάθμιση αναπαρίσταται στο σχήμα 5.4. Για να εγγυηθεί η σύγκλιση του αλγορίθμου, το  $\mu$  επιλέγεται να είναι ένας μικρός θετικός αριθμός. Όμως, αυτό προκαλεί αργή σύγκλιση του αλγορίθμου, αλλά τελικά επιτυγχάνεται μικρό μέσο τετραγωνικό σφάλμα πράγμα που δείχνει ότι οι εκτιμώμενοι συντελεστές του φίλτρου ισοστάθμισης βρίσκονται πολύ κοντά στη βέλτιστη τιμή τους. Η επιλογή μεγάλου  $\mu$ , προκαλεί γρήγορη σύγκλιση, ωστόσο ο αλγόριθμος εκδηλώνει μεγάλα άλματα στο μέσο τετραγωνικό σφάλμα. [2]



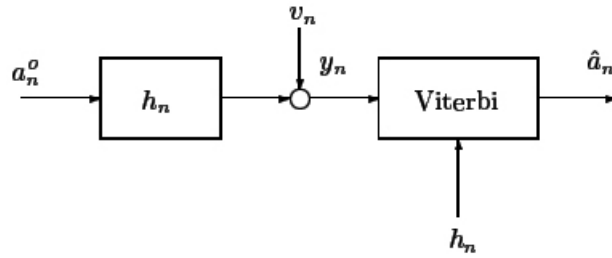
Σχήμα 5.4 Εκπαιδευόμενη Προσαρμοστική Γραμμική Ισοστάθμιση

## 5.3 Μη Γραμμική Ισοστάθμιση

### 5.3.i Γενικά

Οι γραμμικοί ισοσταθμιστές ενισχύουν τον θόρυβο όταν το κανάλι έχει φασματικά κενά (spectral nulls), δηλαδή,  $|H(f)| \approx 0$  για κάποιες συχνότητες  $f$ . Αυτό συμβαίνει διότι σε αυτές τις συχνότητες το πλάτος του φάσματος του ισοσταθμιστή παίρνει μεγάλες τιμές. Συνήθως, σε αυτές τις περιπτώσεις οι γραμμικοί ισοσταθμιστές δεν είναι αποδοτικοί και τα προβλήματα που εισαγάγουν αντιμετωπίζονται με μη γραμμικές διατάξεις, όπως ο αλγόριθμος Viterbi, ο ισοσταθμιστής ανάδρασης απόφασης (decision-feedback equalizer), κλπ. Ακολουθεί η ανάλυση του αλγορίθμου Viterbi για ισοστάθμιση καναλιού που μας απασχόλησε στην παρούσα εργασία. [10]

### 5.3.ii Trellis Αναπαράσταση<sup>3</sup>



Σχήμα 5.5 Ισοστάθμιση Viterbi

Η σχέση εισόδου-εξόδου του καναλιού περιγράφεται από τη συνέλιξη

$$y_n = \sum_{k=0}^L h_k a_{n-k}^o + u_n$$

όπου τα  $a_n^o$  αποτελούν την ακολουθία συμβόλων εισόδου, η οποία είναι άγνωστη στον δέκτη,  $h_n$  η κρουστική απόκριση του καναλιού και  $u_n$  η συνιστώσα θορύβου. Τα σύμβολα αυτά ανήκουν γενικά σε έναν αστερισμό, έστω  $M$ -PAM. Επίσης, στον δέκτη υποθέτουμε ότι γνωρίζουμε την κρουστική απόκριση του καναλιού, ωστόσο στην πράξη διαθέτουμε μια εκτίμηση. Για το λόγο αυτό, αναζητούμε μια ακολουθία  $\{a_1^*, \dots, a_N^*\}$ , η οποία είναι λύση του προβλήματος

<sup>3</sup>Λιάβας, Αθανάσιος. Πανεπιστημιακές Σημειώσεις Μαθήματος “Ψηφιακά Τηλεπικοινωνιακά Συστήματα ΙΙ”

ελαχιστοποίησης

$$\min_{\substack{a_1, \dots, a_N \\ a_i \in M\text{-PAM}}} \sum_{n=1}^{N+L} \left( y_n - \sum_{k=0}^L h_k a_{n-k} \right)^2 \quad (1)$$

Όταν τα σύμβολα εισόδου είναι ισοπίθανα και ο θόρυβος λευκός Gaussian, τότε μπορεί να αποδειχθεί ότι η ακολουθία αυτή ελαχιστοποιεί τη μέση πιθανότητα σφάλματος.

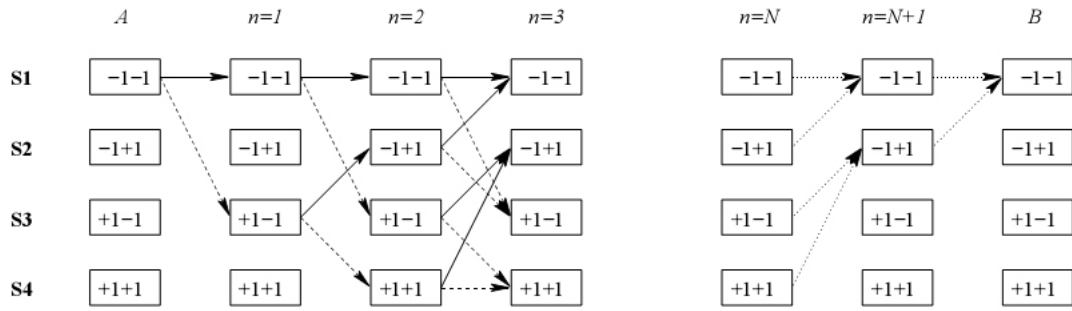
Αφού κάθε σύμβολο  $a_n$  μπορεί να πάρει  $M$  τιμές, το πλήθος των διαφορετικών ακολουθιών μήκους  $N$  είναι  $M^N$ . Λόγω της τεράστιας υπολογιστικής πολυπλοκότητας, η εύρεση της βέλτιστης ακολουθίας είναι πρακτικά αδύνατη. Για να επιλυθεί το παραπάνω πρόβλημα, θα δημιουργήσουμε μια έξυπνη αναπαράσταση όλων των δυνατών καταστάσεων του καναλιού που ονομάζεται *Trellis Αναπαράσταση* πάνω στην οποία θα εφαρμοστεί ο αλγόριθμος Viterbi.

Ας υποθέσουμε ότι το κανάλι έχει 3 συντελεστές  $h_0, h_1, h_2$  και ο αστερισμός εισόδου είναι 2-PAM με αλφάβητο  $(-1, +1)$ . Γενικά, το κανάλι έχει  $L+1$  συντελεστές  $h_0, \dots, h_L$  και ο αστερισμός είναι  $M$ -PAM. Για τον υπολογισμό της εξόδου τη χρονική στιγμή  $n_0$ , το κανάλι χρησιμοποιεί το τρέχον δείγμα εισόδου  $a_{n_0}$  καθώς και 2 προηγούμενα δείγματα  $a_{n_0-1}$  και  $a_{n_0-2}$ . Συνεπώς, κάθε χρονική στιγμή, το κανάλι διαθέτει στη μνήμη του τα δείγματα εισόδου των 2 (γενικά  $L$ ) προηγούμενων χρονικών στιγμών. Κάθε χρονική στιγμή, τα δείγματα που διαθέτει το κανάλι στη μνήμη του, καθορίζουν την **κατάσταση** του καναλιού. Στην περίπτωση μας, κάθε κατάσταση  $\mathbf{S}$  αποτελείται από 2 στοιχεία  $s_1, s_2$  τα οποία τη χρονική στιγμή  $n_0$  είναι

$$s_1 = a_{n_0-1}, \quad s_2 = a_{n_0-2}$$

Το πλήθος των διαφορετικών καταστάσεων στις οποίες μπορεί να βρεθεί το κανάλι είναι  $M^L$ . Για το παράδειγμά μας, είναι  $2^2 = 4$ . Το κανάλι παράγει μια έξοδο για κάθε χρονική στιγμή και αλλάζει κατάσταση. Η νέα κατάσταση υπολογίζεται ως εξής: Αν το κανάλι βρίσκεται τη χρονική στιγμή  $n_0$  στην κατάσταση  $\mathbf{S} = s_1 s_2 \dots s_L$  και το τρέχον σύμβολο είναι το  $s_0$ , τότε τη χρονική στιγμή  $n_0 + 1$  η κατάσταση του καναλιού θα είναι  $\mathbf{S} = s_0 s_1 \dots s_{L-1}$ . Στο σχήμα 5.6 έχει σχεδιαστεί η αλληλεπίδραση ανάμεσα στα πιθανά δείγματα εισόδου και στις καταστάσεις του καναλιού για τις χρονικές στιγμές  $n = 1$  μέχρι  $n = N + 2$ . Με συνεχή γραμμή σχεδιάστηκαν οι ακμές που αντιστοιχούν σε σύμβολο εισόδου  $-1$  και με διακεκομμένη, αυτές που αντιστοιχούν σε  $+1$ . Οι ακμές των 2 τελευταίων χρονικών στιγμών διαφέρουν από τις υπόλοιπες, διότι δεν υπάρχει δείγμα εισόδου. Συνήθως, ξεκινάμε από μια κατάσταση  $\mathbf{S}$ , η οποία επιλέγεται αυθαίρετα αφού αρχικά το κανάλι δεν διαθέτει τίποτα στην μνήμη του. Στο παράδειγμά μας, αρχίζουμε από την κατάσταση

$S_1 = -1 - 1$ . Αντίστοιχα γίνεται και στο τέλος της Trellis αναπαράστασης. Οι παραπάνω συμβάσεις δεν οδηγούν σε σφάλματα, αρκεί ο υπολογισμός του τοπικού κόστους να γίνεται χρησιμοποιώντας την πραγματική και όχι την συμβατική κατάσταση του καναλιού. Στην πραγματικότητα, θα μπορούσαμε να είχαμε αρχίσει από οποιαδήποτε κατάσταση της πρώτης στήλης του Trellis και να είχαμε καταλήξει σε οποιαδήποτε κατάσταση της τελευταίας στήλης, αρκεί να είχαμε υπολογίσει τα τοπικά κόστη σωστά. Καθεμία από τις  $2^N$  δυνατές ακολουθίες εισόδου, καθορίζει ένα μονοπάτι στο Trellis και αντίστροφα.



Σχήμα 5.6 Trellis Αναπαράσταση για  $L=2$  και σύμβολα εισόδου 2-PAM

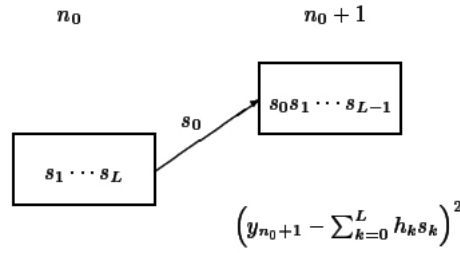
Όπως φαίνεται και από το σχήμα, σε κάθε κατάσταση καταλήγουν 2 (γενικά  $M$ ) ακμές και από κάθε κατάσταση εξέρχονται 2 (γενικά  $M$ ) ακμές. Η ελαχιστοποίηση της σχέσης (1) μπορεί να μεταφραστεί σε εύρεση του μονοπατιού ελαχίστου κόστους από την αρχή  $A$  έως το τέλος  $B$  του Trellis, ορίζοντας τα βάρη (κόστη) των ακμών ως εξής: Υποθέτοντας ότι τη χρονική στιγμή  $n_0$  το κανάλι είναι στην κατάσταση  $\mathbf{S} = s_1 \dots s_L$  και το νέο σύμβολο εισόδου είναι το  $s_0$ , τότε η έξοδος του καναλιού θα είναι

$$\sum_{k=0}^L h_k s_k$$

Το τοπικό κόστος Viterbi, όπως φαίνεται και στο σχήμα 5.7, θα είναι η απόσταση αυτής της εξόδου από την πραγματική έξοδο του καναλιού  $y_{n_0+1}$ , με την απόσταση να αντιστοιχεί στο τετράγωνο της διαφοράς, δηλαδή

$$\left( y_{n_0+1} - \sum_{k=0}^L h_k s_k \right)^2 = \left( y_{n_0+1} - \sum_{k=0}^L h_k a_{n_0+1-k} \right)^2$$

Η τελευταία ισότητα ισχύει επειδή το  $s_0$  δηλώνει το τρέχον δείγμα εισόδου και τα  $s_1 \dots s_L$ , τα  $L$  πιο πρόσφατα δείγματα.



Σχήμα 5.7 Τοπικό Κόστος Viterbi

### 5.3.iii Αναδρομικός Αλγόριθμος Viterbi<sup>4</sup>

Σε κάθε χρονική στιγμή, όλα τα μονοπάτια από την αρχή A έως το τέλος B του Trellis, περνούν από τις  $M^L$  δυνατές καταστάσεις  $\mathbf{S}_i$  με  $i = 1, \dots, M^L$ . Αν μπορούσαμε να υπολογίσουμε αποδοτικά το βέλτιστο μονοπάτι από το A σε κάθε κατάσταση τη χρονική στιγμή  $n_0$ , τότε θα είχαμε τα  $M^L$  υποψήφια αρχικά τμήματα του βέλτιστου μονοπατιού μέχρι τη χρονική στιγμή  $n_0$ . Αυτός ο υπολογισμός αναδρομικά μπορεί να γίνει ως εξής:

Έστω ότι τη χρονική στιγμή  $n_0$  γνωρίζουμε για κάθε δυνατή κατάσταση  $\mathbf{S}_i$  με  $i = 1, \dots, M^L$ , το βέλτιστο μονοπάτι από το A στην κατάσταση αυτή,  $\mathcal{P}_{S_i}^{n_0}$ , καθώς και το κόστος του,  $C_{S_i}^{n_0}$ . Τότε, μπορούμε να υπολογίσουμε το βέλτιστο μονοπάτι για την κατάσταση  $\mathbf{S}_j$ ,  $j = 1, \dots, M^L$ , τη χρονική στιγμή  $n_0 + 1$ , συγκρίνοντας το άθροισμα

$$C_{S_i}^{n_0} + \text{τοπικό κόστος}_{ij}^{n_0+1}$$

για όλες τις καταστάσεις  $\mathbf{S}_i$ , της χρονικής στιγμής  $n_0$  για τις οποίες υπάρχει δυνατότητα μετάβασης στην  $\mathbf{S}_j$ , τη χρονική στιγμή  $n_0 + 1$ . Ως τοπικό κόστος  $_{ij}^{n_0+1}$  ορίζεται το βάρος της ακμής που συνδέει τις 2 καταστάσεις.

Το βέλτιστο μονοπάτι καθορίζεται από το ελάχιστο από τα παραπάνω αθροίσματα ως εξής: Αν η μετάβαση στην  $\mathbf{S}_j$  γίνεται μέσω του συμβόλου  $s_*$  και το παραπάνω άθροισμα ελαχιστοποιείται για  $i = i_*$ , τότε

$$\mathcal{P}_{S_j}^{n_0+1} = \mathcal{P}_{S_{i_*}}^{n_0} \cup \{s_*\}$$

και

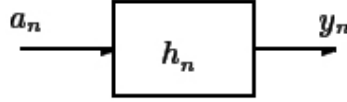
$$C_{S_j}^{n_0+1} = C_{S_{i_*}}^{n_0} + \text{τοπικό κόστος}_{i_*j}^{n_0+1}$$

Δουλεύοντας αντίστοιχα και για τις άλλες καταστάσεις της χρονικής στιγμής  $n_0 + 1$ , υπολογίζουμε τα επαυξημένα βέλτιστα μονοπάτια και τα κόστη τους και μπορούμε να συνεχίσουμε στην αναδρομή για τη χρονική στιγμή  $n_0 + 2$ . Στο τέλος, επιλέγουμε τα  $N$  πρώτα σύμβολα του μονοπατιού ελαχίστου κόστους από το A στο B και αυτή είναι η εκτίμησή μας για την ακολουθία εισόδου.

<sup>4</sup>Λιάβας, Αθανάσιος. Πανεπιστημιακές Σημειώσεις Μαθήματος “Ψηφιακά Τηλεπικοινωνιακά Συστήματα II”

## 5.4 Εκτίμηση Καναλιού

### 5.4.i Μέθοδος Ετεροσυσχέτισης



Σχήμα 5.8 Απλό Κανάλι

Έστω το κανάλι διακριτού χρόνου, κρουστικής απόκρισης  $h_n$ , μήκους  $L$ , όπως φαίνεται και στο σχήμα 5.8. Θεωρούμε ότι τα γνωστά σύμβολα εισόδου είναι τα  $a_1, a_2, \dots, a_k$ . Η ακολουθία ετεροσυσχέτισης μεταξύ των γνωστών συμβόλων εισόδου  $a_k$  και της εξόδου του καναλιού δίνεται από την εξίσωση

$$\begin{aligned} r_n &= \mathcal{E} [y_{n+i} a_i] \\ &= \mathcal{E} \left[ \left( \sum_{j=0}^{L-1} h_j a_{n+i-j} \right) a_i \right] \\ &= \sum_{j=0}^{L-1} h_j \mathcal{E} [a_{n+i-j} a_i] \end{aligned}$$

όπου τα  $a_{n+i-j}, a_i$  είναι όμοια όταν  $n+i-j = i \Rightarrow n = j$ . Επειδή τα σύμβολα εισόδου είναι ανεξάρτητα, όμοια κατανομημένα, με μέση τιμή 0 έχουμε

$$\mathcal{E} [a_i a_{n+i-j}] = \begin{cases} 0 & , \quad n \neq j \\ \sigma^2 & , \quad n = j \end{cases}$$

Τελικά, επιζεί ένας μόνο όρος από το άθροισμα (δηλαδή, όταν  $n = j$ ) και όλοι οι άλλοι είναι 0. Άρα, έχουμε

$$r_n = h_n \sigma^2$$

όπου στη περίπτωση μας (4-PAM ισοπίθανα σύμβολα  $[-3, -1, +1, +3]$ ) θα έχουμε

$$\sigma^2 = \sum_{i=1}^4 p_i x_i^2 = \frac{1}{4}(-3)^2 + \frac{1}{4}(-1)^2 + \frac{1}{4}(1)^2 + \frac{1}{4}(3)^2 = 5$$

Μια εκτίμηση του  $r_n$  είναι η

$$\hat{r}_n = \frac{1}{k} \sum_{i=1}^k y_{n+i} a_i$$

Άρα μια εκτίμηση του καναλιού είναι η

$$\hat{h}_n = \frac{\hat{r}_n}{\sigma^2}$$



### 5.4.ii Μέθοδος Ελαχίστων Τετραγώνων<sup>5</sup>

Έστω  $n_2 - n_1 > M$ . Οι έξοδοι του καναλιού  $y_n$  που εξαρτώνται μόνο από τα σύμβολα εκπαίδευσης  $a_{n_1}, \dots, a_{n_2}$  μπορούν να εκφραστούν ως εξής

$$\begin{aligned} y_{n_1+M} &= h_0^o a_{n_1+M} + h_1^o a_{n_1+M-1} + \dots + h_M^o a_{n_1} + u_{n_1+M} \\ &\vdots \\ y_{n_2} &= h_0^o a_{n_2} + h_1^o a_{n_2-1} + \dots + h_M^o a_{n_2-M} + u_{n_2} \end{aligned}$$

Σε μορφή πινάκων-διανυσμάτων, οι παραπάνω σχέσεις γράφονται

$$\begin{bmatrix} y_{n_1+M} \\ \vdots \\ y_{n_2} \end{bmatrix} = \begin{bmatrix} a_{n_1+M} & a_{n_1+M-1} & \dots & a_{n_1} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n_2} & a_{n_2-1} & \dots & a_{n_2-M} \end{bmatrix} \begin{bmatrix} h_0^o \\ \vdots \\ h_M^o \end{bmatrix} + \begin{bmatrix} u_{n_1+M} \\ \vdots \\ u_{n_2} \end{bmatrix}$$

δηλαδή

$$y = Ah^o + u$$

όπου  $y$  και  $A$  είναι γνωστά,  $h^o$  και  $u$ , άγνωστα, με το διάνυσμα  $y$  να αντιστοιχεί στην έξοδο του καναλιού, τον πίνακα  $A$  στα γνωστά σύμβολα, το διάνυσμα  $h^o$  στην κρουστική απόκριση του καναλιού και  $u$  το διάνυσμα θορύβου. Αναζητώντας το  $h^o$ , θα υπολογίσουμε το  $h_{LS}$  που ελαχιστοποιεί τη συνάρτηση κόστους

$$C(h) = \|y - Ah\|_2^2 = (y - Ah)^T (y - Ah)$$

όπου  $h \triangleq [h_0 \dots h_M]^T$ . Δηλαδή, αναζητούμε την ποσότητα

$$h_{LS} = \arg \min_h C(h)$$

Η  $C(h)$  είναι τετραγωνική συνάρτηση των στοιχείων του  $h$ , δηλαδή είναι γραμμικός συνδυασμός γινομένων της μορφής  $h_i h_j$ . Μπορεί να αποδειχθεί ότι αν ο πίνακας  $A$  είναι πλήρους βαθμού, δηλαδή οι στήλες του είναι γραμμικά ανεξάρτητες, τότε το παραπάνω πρόβλημα ελαχιστοποίησης έχει μοναδική λύση, η οποία δίνεται από τη λύση της εξίσωσης

$$\left. \frac{\partial C(h)}{\partial h} \right|_{h=h_{LS}} = 0$$

---

<sup>5</sup> Λιάβας, Αθανάσιος. Πανεπιστημιακές Σημειώσεις Μαθήματος “Ψηφιακά Τηλεπικοινωνιακά Συστήματα ΙΙ”

Έχουμε

$$\begin{aligned}\frac{\partial C(h)}{\partial h} &= \frac{\partial}{\partial h} (y^T y - y^T A h - h^T A^T y + h^T A^T A h) \\ &= -2 \frac{\partial}{\partial h} (y^T A h) + \frac{\partial}{\partial h} (h^T A^T A h) = -2 A^T y + 2 A^T A h\end{aligned}$$

όπου έχουμε χρησιμοποιήσει τη συμμετρία του πίνακα  $A^T A$  και το ότι  $y^T A h = h^T A^T y$ , διότι κάθε βαθμωτό είναι ίσο με το ανάστροφό του. Τελικά, λαμβάνουμε τη λύση ελαχίστων τετραγώνων

$$h_{LS} = (A^T A)^{-1} A^T y$$

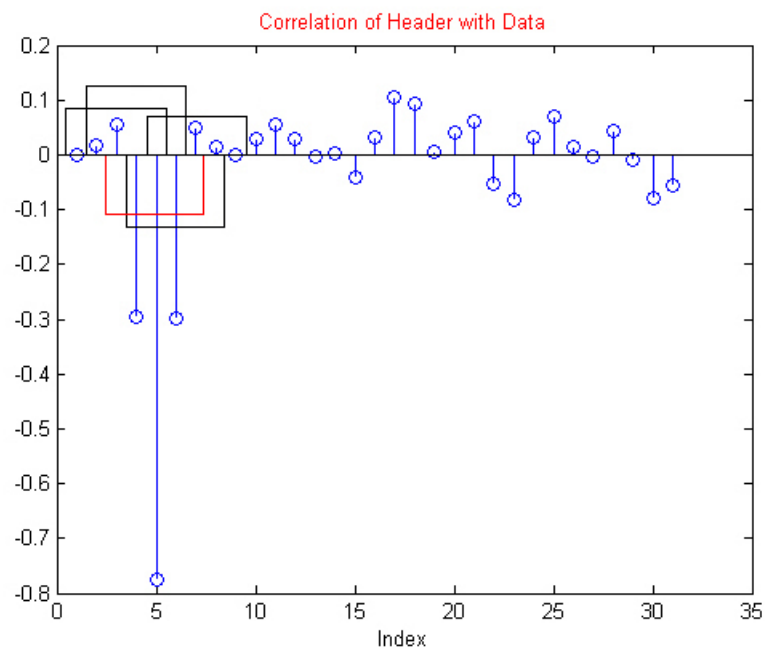
Έτσι, η παραπάνω λύση  $h_{LS}$ , μας δίνει μια εκτίμηση για το κανάλι.

## 5.5 Συγχρονισμός Πλαισίου

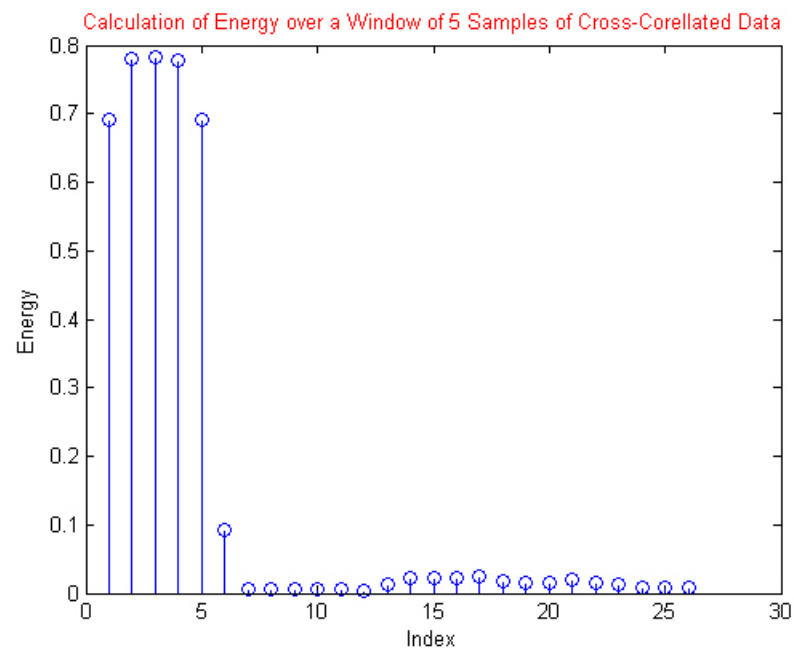
Συνήθως, η πληροφορία στο μεταδιδόμενο σήμα είναι ομαδοποιημένες ακολουθίες. Κάθε τέτοια ομάδα ονομάζεται *πλαίσιο*. Αυτό αποτελείται από 2 μέρη:[1]

1. την επικεφαλίδα, που μας δείχνει την έναρξη κάθε πλαισίου και αποτελείται από γνωστά σύμβολα που δίνονται από τις προδιαγραφές του συστήματος και τα οποία μπορούν επίσης να χρησιμοποιηθούν για εκτίμηση ή και ισοστάθμιση καναλιού
2. την μεταδιδόμενη πληροφορία, η οποία τοποθετείται αμέσως μετά την επικεφαλίδα

Σκοπός του *Συγχρονισμού Πλαισίου* είναι να βρούμε τη θέση έναρξης του πλαισίου και να αφαιρέσουμε την επικεφαλίδα από κάθε πλαίσιο, λαμβάνοντας τελικά μόνο την ωφέλιμη πληροφορία. Πιθανώς, το λαμβανόμενο σήμα να μην αρχίζει με ένα πλαίσιο, οπότε με τη βοήθεια ενός συσχετιστή μπορούμε να εντοπίσουμε την αρχή του πλαισίου χρησιμοποιώντας τη γνωστή ακολουθία και μετακινώντας την πάνω στην ακολουθία εισόδου, με τρόπο όπως υποδεικνύεται στη συνέχεια. Θεωρούμε αυθαίρετα ένα παράθυρο μήκους  $M$ , στην περίπτωση μας  $M = 5$ , το οποίο το μετακινούμε κατά μια θέση κάθε φορά πάνω στα στοιχεία που έχουν προκύψει από την ετεροσυσχέτιση του λαμβανόμενου σήματος με την επικεφαλίδα ενός πλαισίου, όπου η ακολουθία ετεροσυσχέτισης είναι μια εκτίμηση για το κανάλι και υπολογίζουμε την ενέργεια κάθε παραθύρου. Στο σημείο που η ενέργεια είναι μέγιστη έχουμε την έναρξη του πλαισίου. Στα σχήματα που ακολουθούν, παρουσιάζουμε ένα παράδειγμα, όπου στο σχήμα 5.9 φαίνεται η ακολουθία ετεροσυσχέτισης. Αυτή προέκυψε χρησιμοποιώντας ένα παράθυρο 245 4-PAM συμβόλων με αλφάβητο  $(-3, -1, +1, +3)$ , τα οποία αποτελούν την επικεφαλίδα ενός πλαισίου και είναι γνωστά στον δέκτη, συσχετίζοντάς τα με την λαμβανόμενη στον δέκτη ακολουθία συμβόλων. Στη συνέχεια, θεωρούμε αυθαίρετα ένα παράθυρο μήκους  $M = 5$  (όσο δηλαδή είναι θεωρητικά το μήκος του καναλιού μας) πάνω στην ακολουθία ετεροσυσχέτισης και υπολογίζουμε την ενέργεια αυτού του παραθύρου. Το παράθυρο κινείται κατά μία θέση προς τα δεξιά και κάθε φορά υπολογίζουμε εκ νέου την ενέργειά του. Στο σχήμα 5.10, δείχνουμε την ενέργεια καθενός από τα 26 παράθυρα μήκους  $M = 5$  πάνω στην ακολουθία ετεροσυσχέτισης του σχήματος 5.9. Στο σημείο όπου η ενέργεια είναι μέγιστη (στο παράδειγμα,  $\text{index}=3$ ), έχουμε την έναρξη του πλαισίου (*Frame Synchronization*) και ταυτόχρονα αν μετακινήσουμε το παράθυρο πάνω στην ακολουθία ετεροσυσχέτισης στη θέση μέγιστης ενέργειας (στο παράδειγμα,  $\text{index}=3$ ) και παίρνοντας τις  $M - 1$  επόμενες τιμές της, διαθέτουμε με τον τρόπο αυτό μια εκτίμηση για το κανάλι.



**Σχήμα 5.9** Το παράθυρο μήκους  $M = 5$  κινείται πάνω στα δεδομένα που προκύπτουν από την ετεροσυσχέτιση



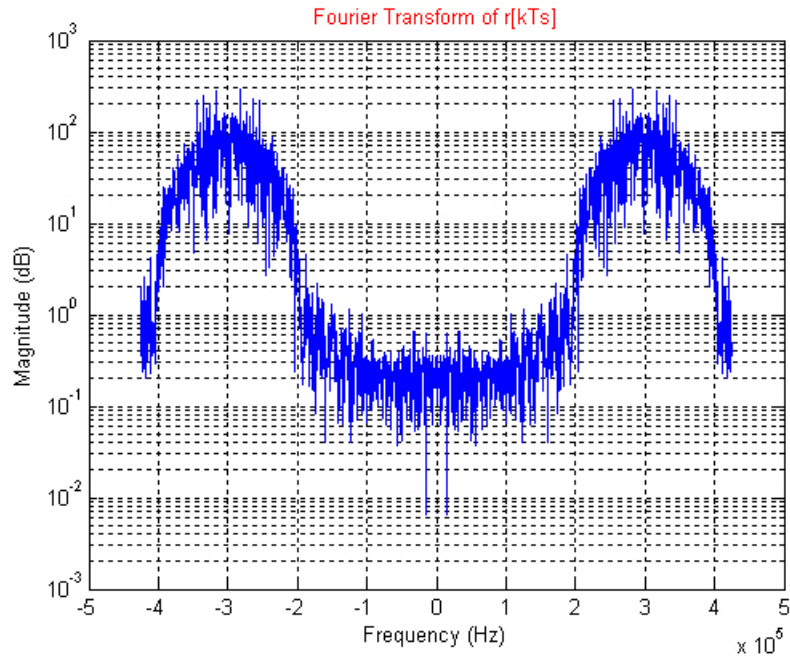
**Σχήμα 5.10** Η θέση της μέγιστης ενέργειας δείχνει την αρχή του πλαισίου

## 6. Προσομοίωση Συστήματος

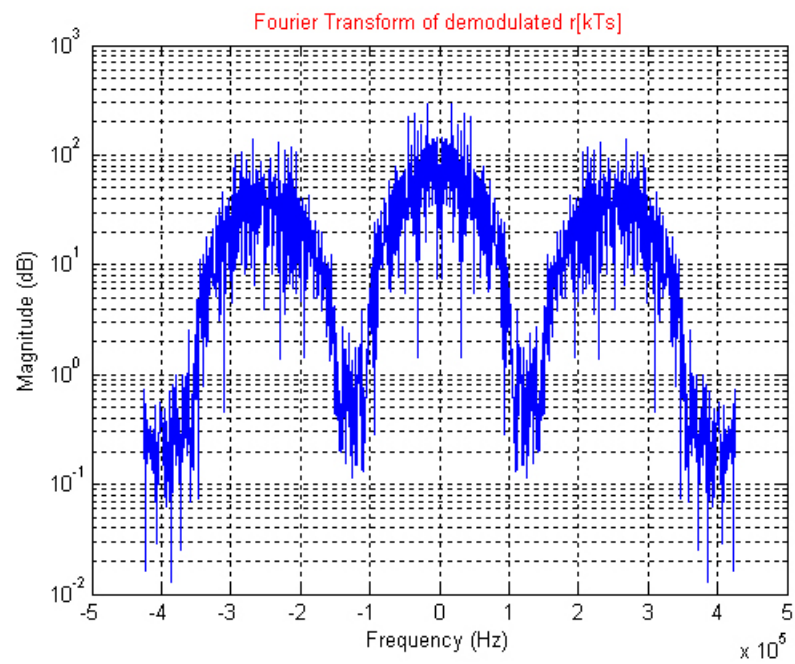
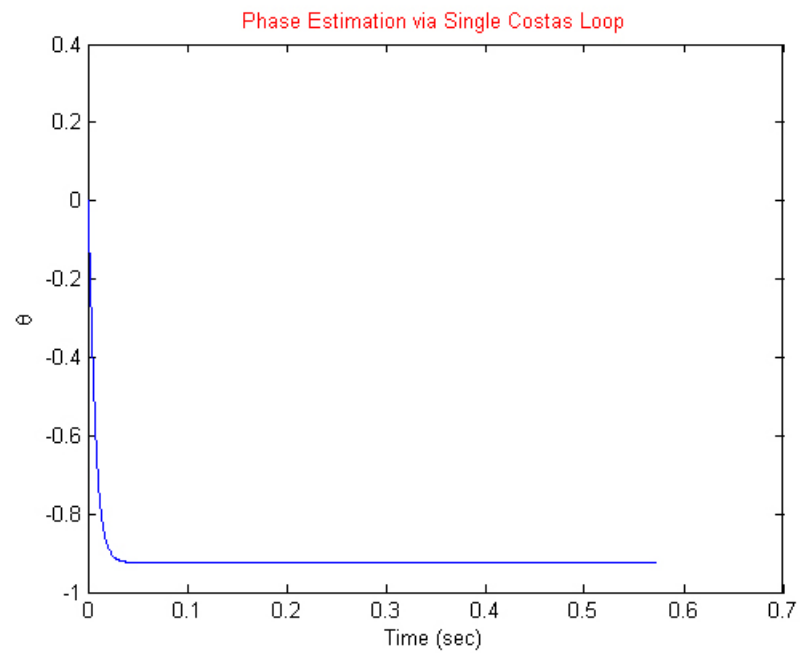
Σε αυτό το κεφάλαιο παρατίθενται αποτελέσματα και γραφήματα όπως αυτά προέκυψαν από την προσομοίωση, όλων των μεθόδων που χρησιμοποιήθηκαν. Οι προδιαγραφές του συστήματος έχουν αναφερθεί στο πρώτο κεφάλαιο της εργασίας και βρίσκονται συγκεντρωμένες στο σχήμα 1.3. Ως δεδομένα εισόδου στο δέκτη χρησιμοποιήθηκαν 3 διαφορετικά mat αρχεία (easy1.mat, medium1.mat, hard1.mat), που αντιστοιχούν σε διαφορετικά κανάλια, επίπεδα θορύβου, παρεμβολές από άλλους χρήστες, αποκλίσεις φάσης-συχνότητας, συγχρονισμού. Στο τέλος της προσομοίωσης παρατίθεται κώδικας στον οποίο γίνεται χρήση του αλγόριθμου Viterbi για ισοστάθμιση τυχαίων καναλιών με τυχαία δεδομένα εισόδου. Στο σχήμα 6.1 φαίνονται τα αποτελέσματα της προσομοίωσης, όπου χρησιμοποιήσαμε κανάλια με μήκος  $L=5$  και δεδομένα εισόδου 1000 4-PAM συμβόλων.

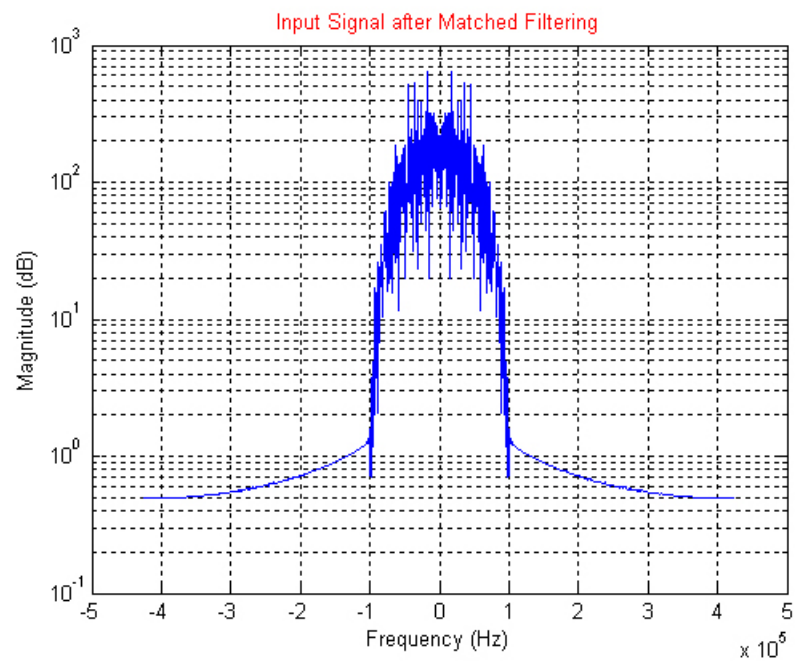
- easy1.mat

- Φάσμα Λαμβανόμενου Σήματος

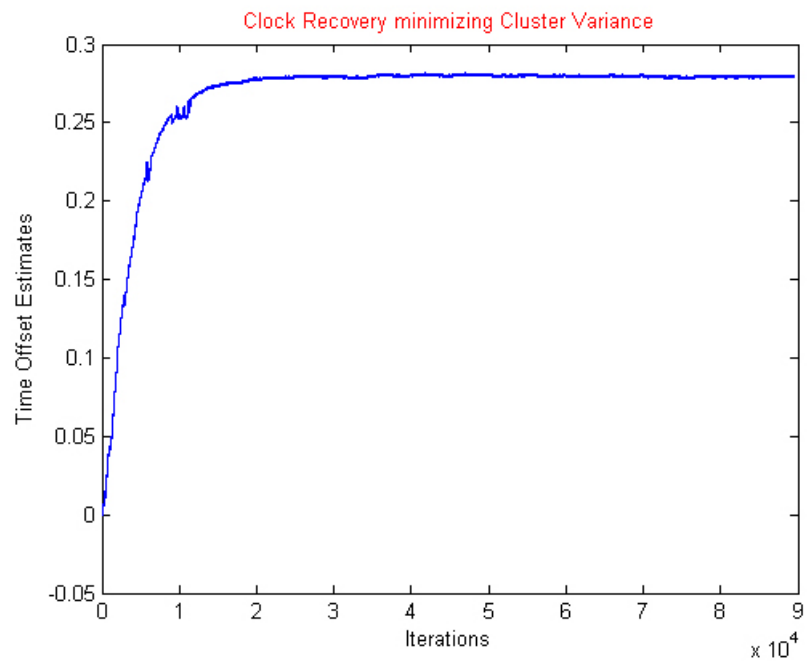


- Ανάκτηση Φορέα με χρήση Βρόχου Costas & Προσαρμοσμένο Φιλτράρισμα

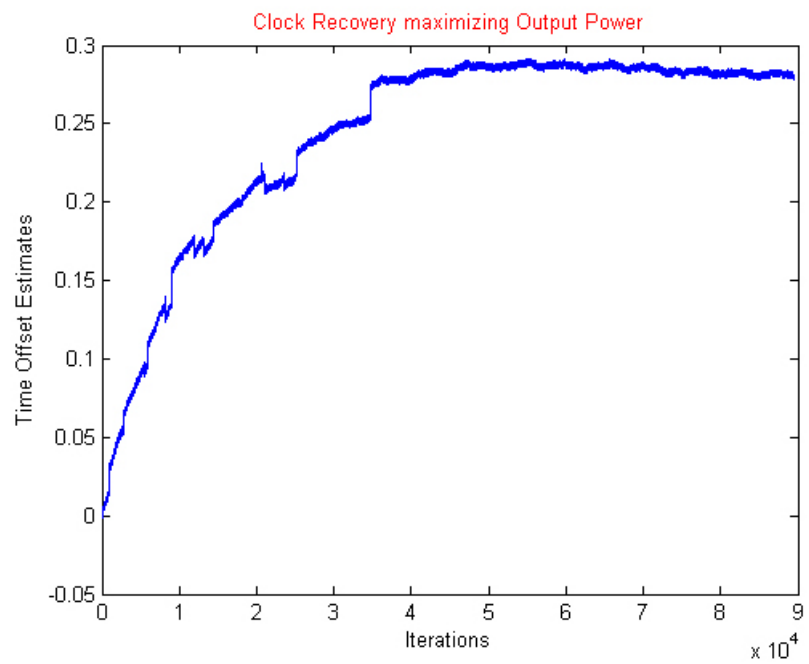




- Ανάκτηση Συγχρονισμού
  - \* Ελαχιστοποίηση του Cluster Variance

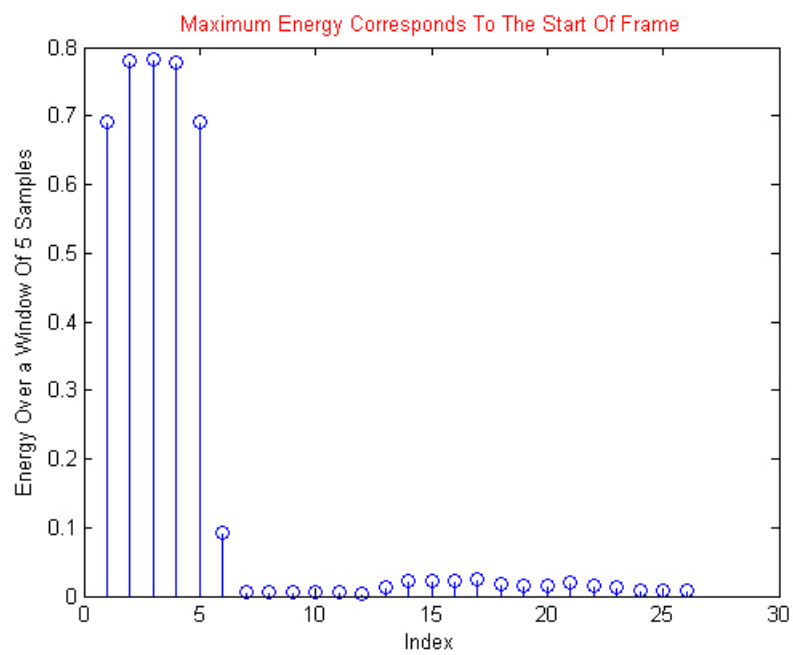
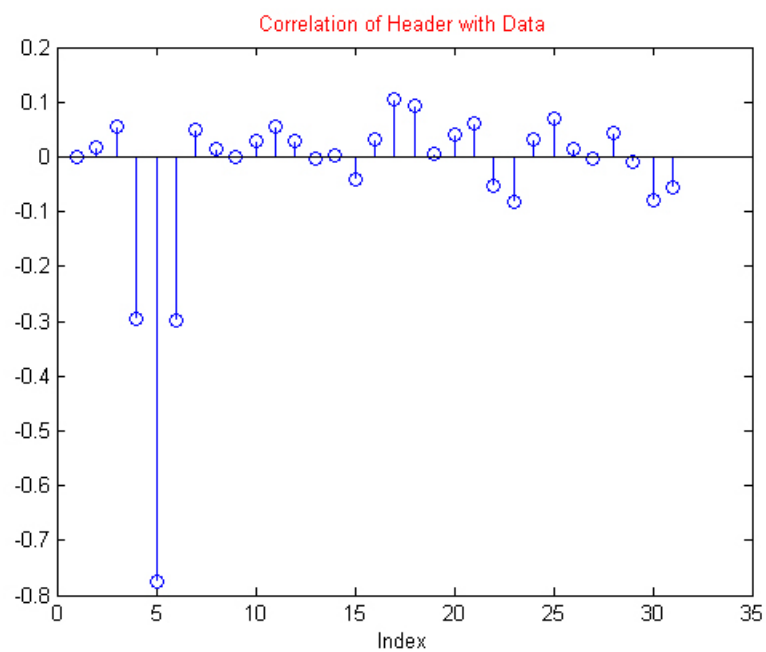


- \* Μεγιστοποίηση της Ισχύος Εξόδου

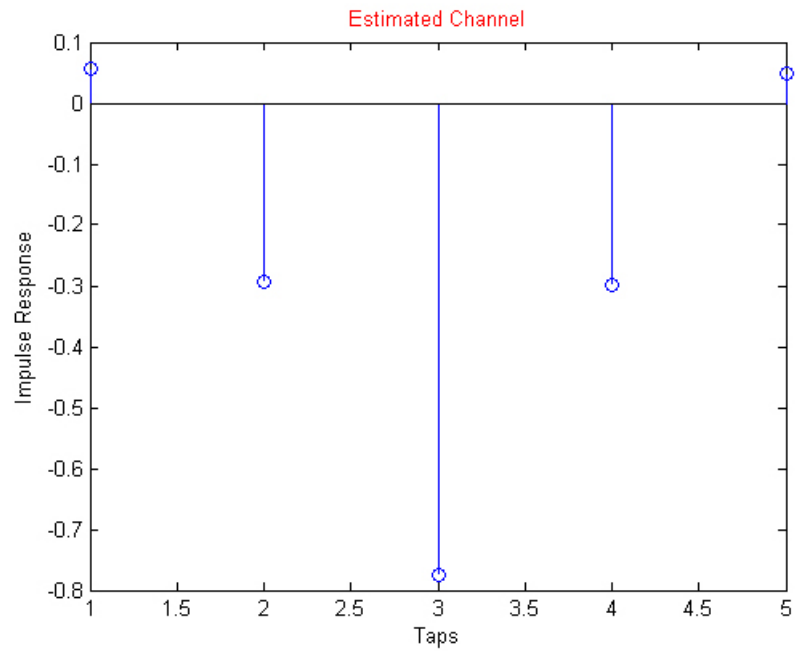




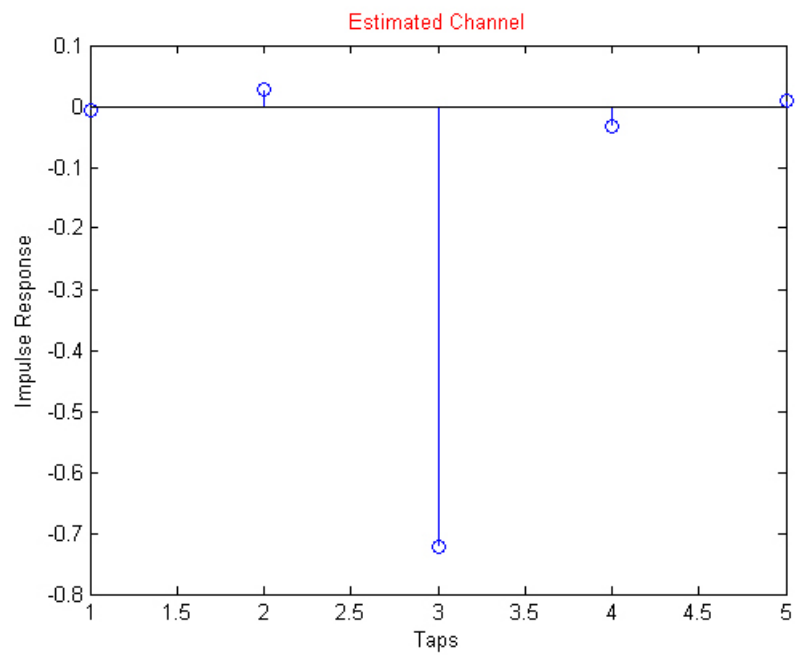
## – Εκτίμηση Καναλιού



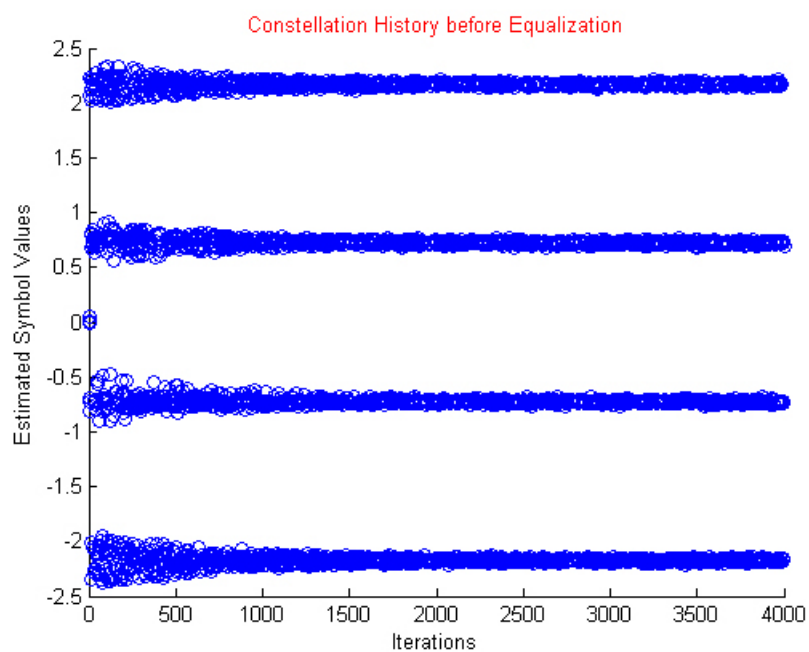
## \* Μέθοδος Ετεροσυσχέτισης



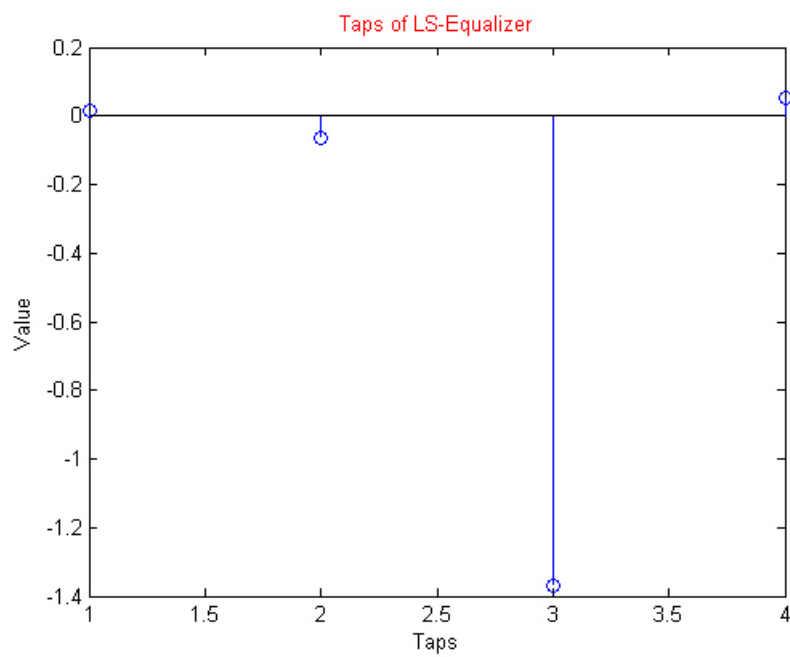
## \* Μέθοδος Ελαχίστων Τετραγώνων (LS)



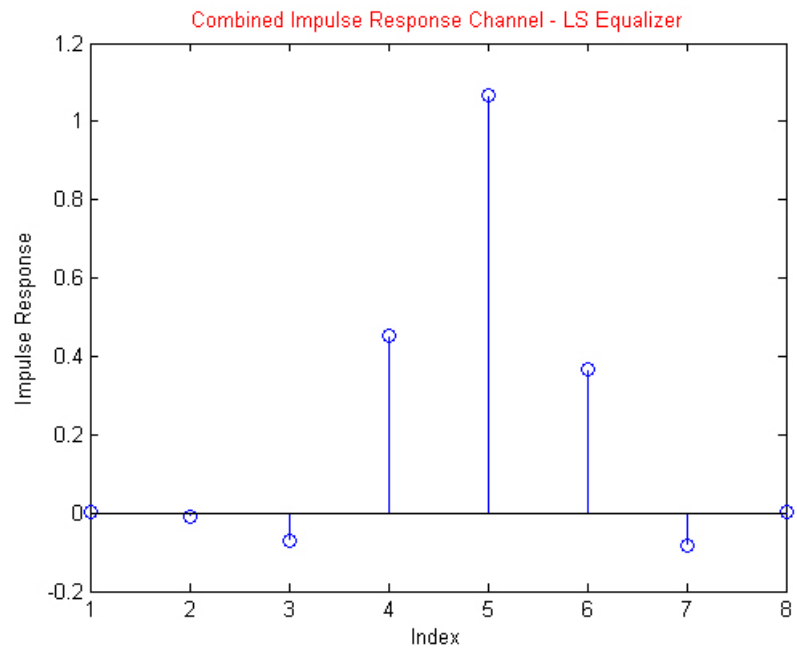
– Ισοστάθμιση Καναλιού



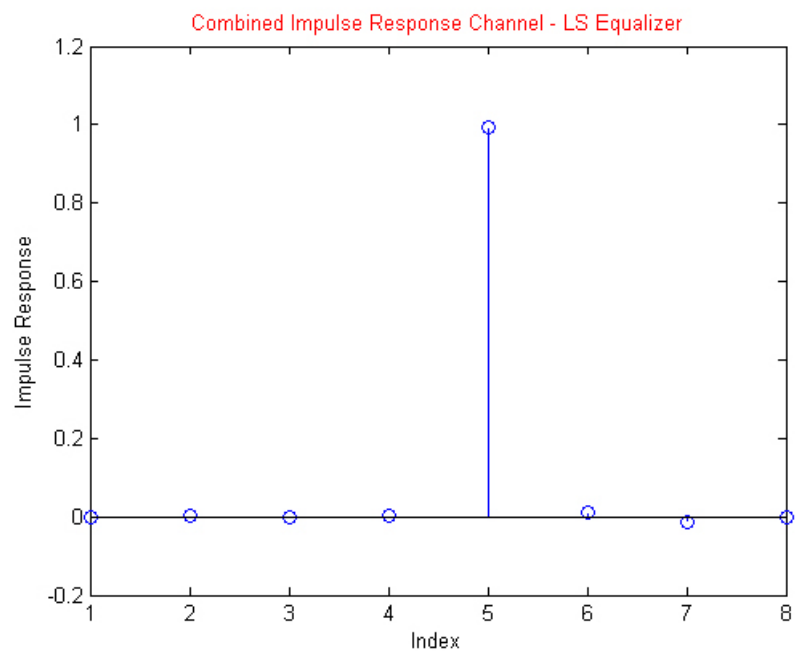
\* Μέθοδος Ελαχίστων Τετραγώνων (LS)

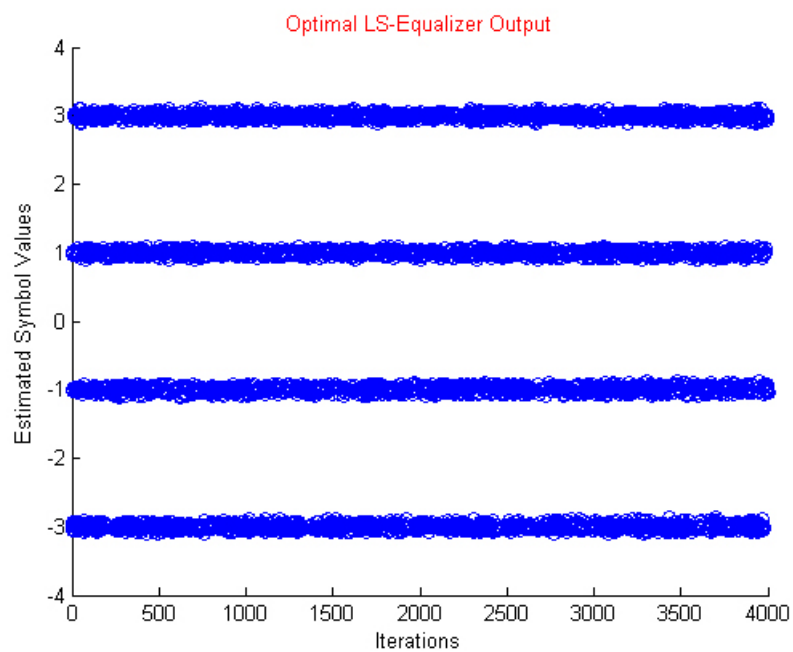


\* Για Εκτίμηση Καναλιού με την Μέθοδο της Ετεροσυσχέτισης

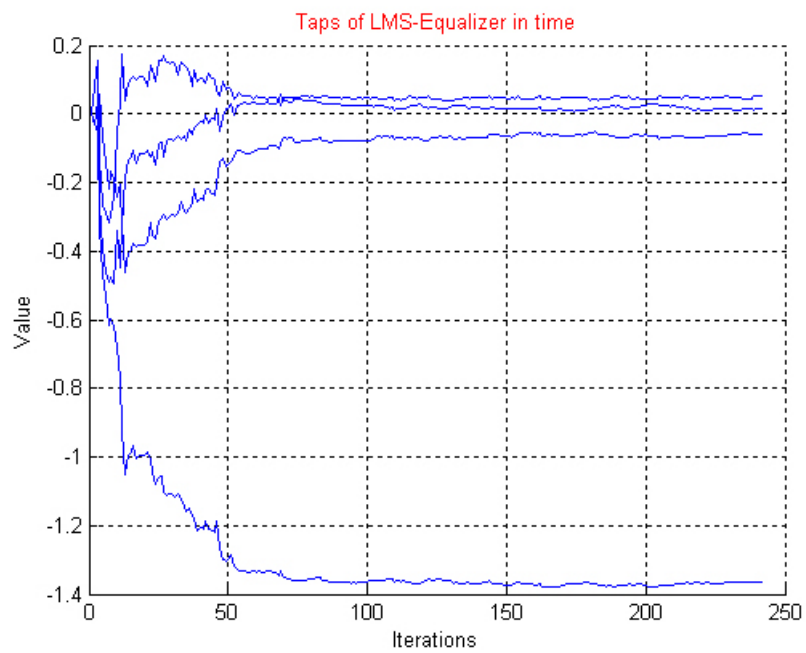


\* Για Εκτίμηση Καναλιού με Λύση Ελαχίστων Τετραγώνων

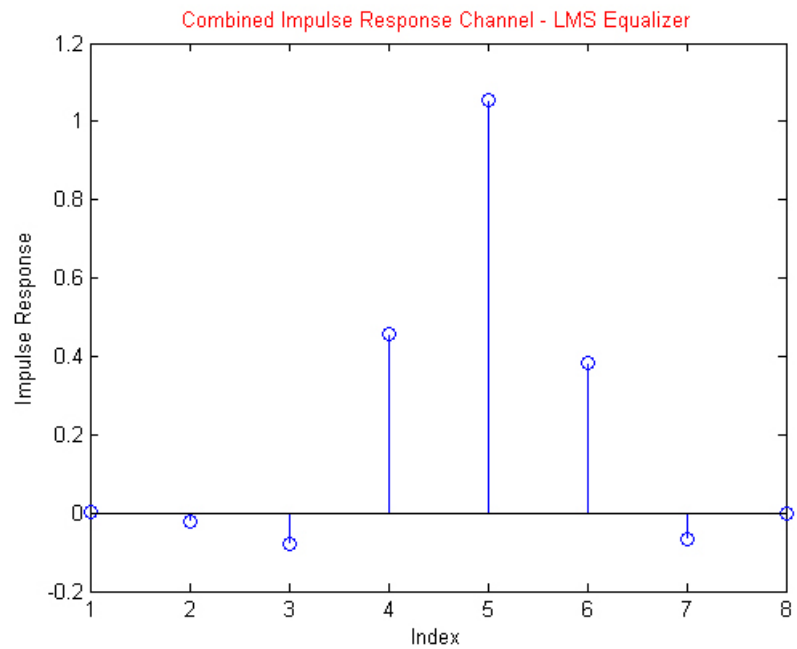




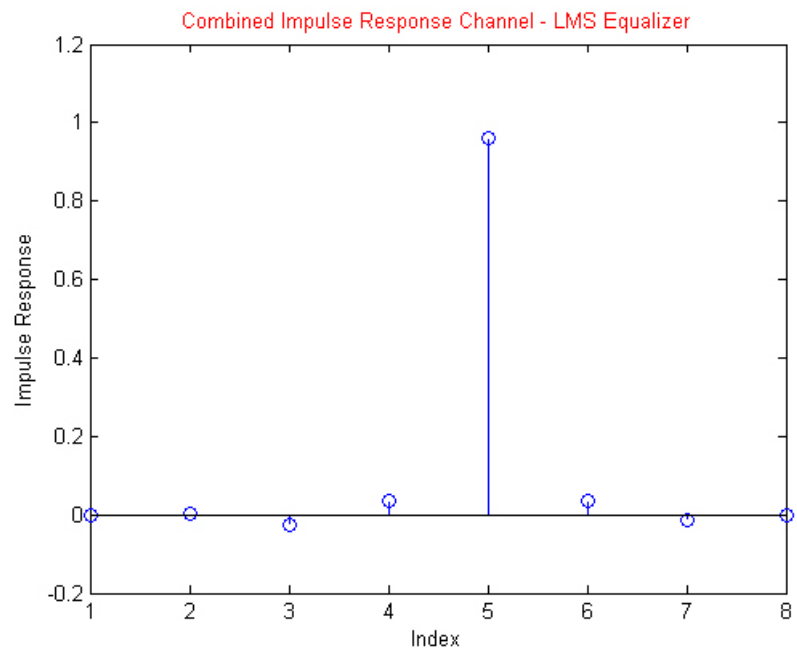
\* Προσαρμοστική Ισοστάθμιση - Αλγόριθμος LMS

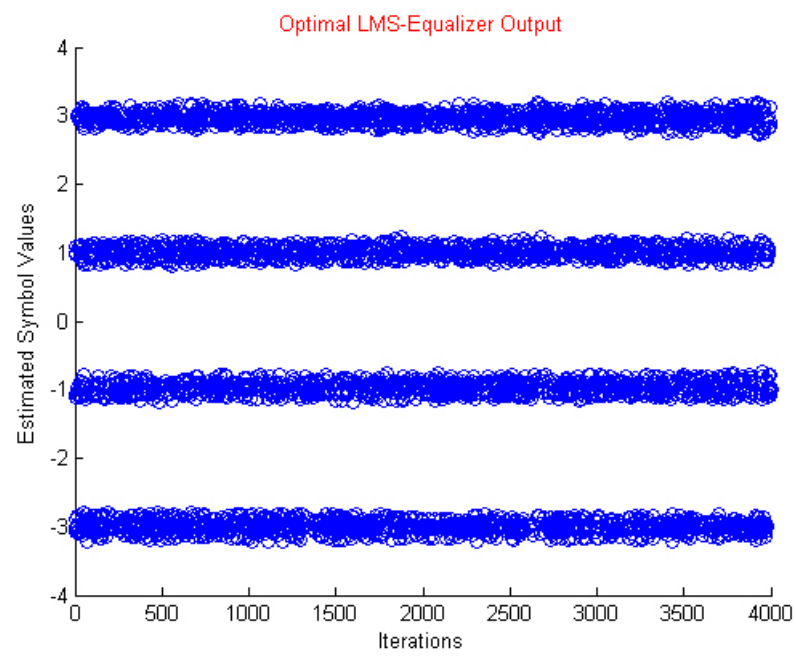


\* Για Εκτίμηση Καναλιού με την Μέθοδο της Ετεροσυσχέτισης

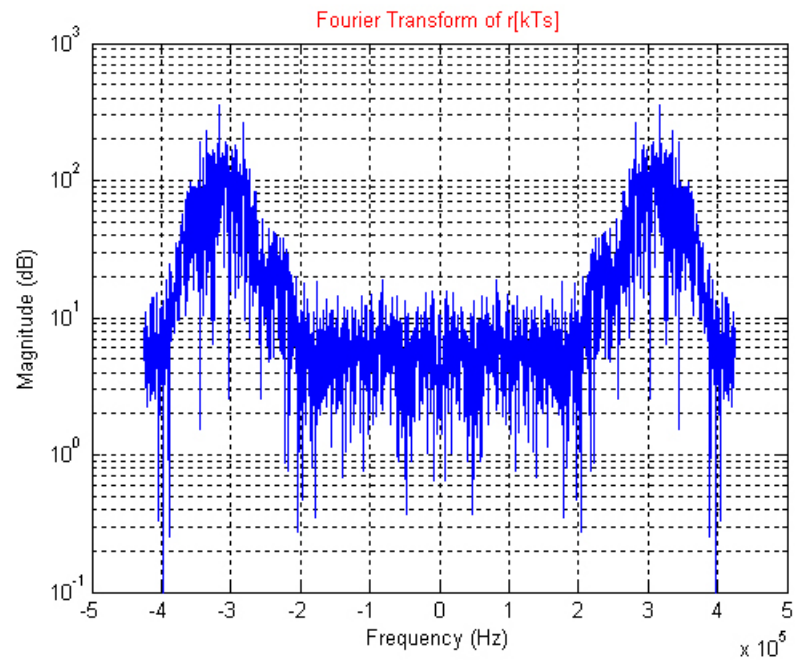


\* Για Εκτίμηση Καναλιού με Λύση Ελαχίστων Τετραγώνων



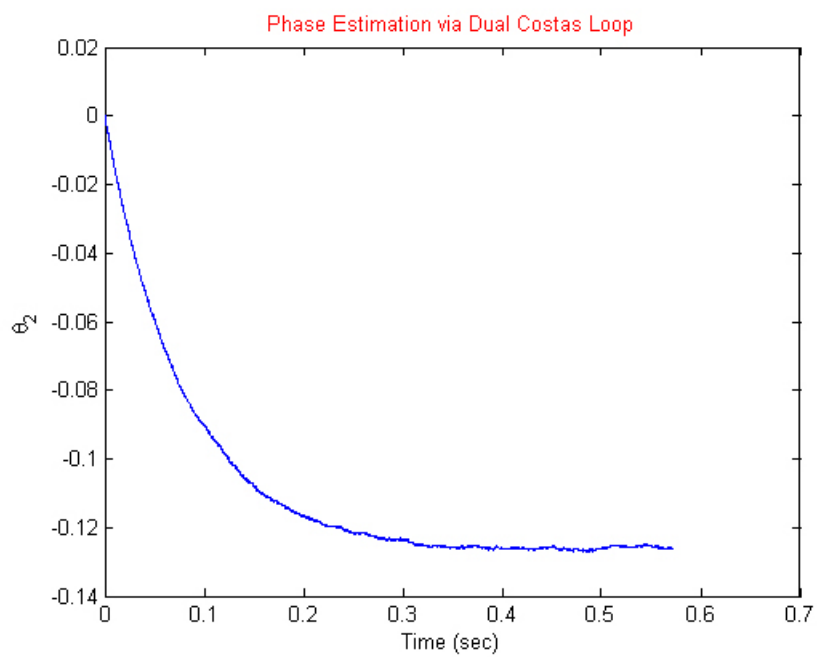
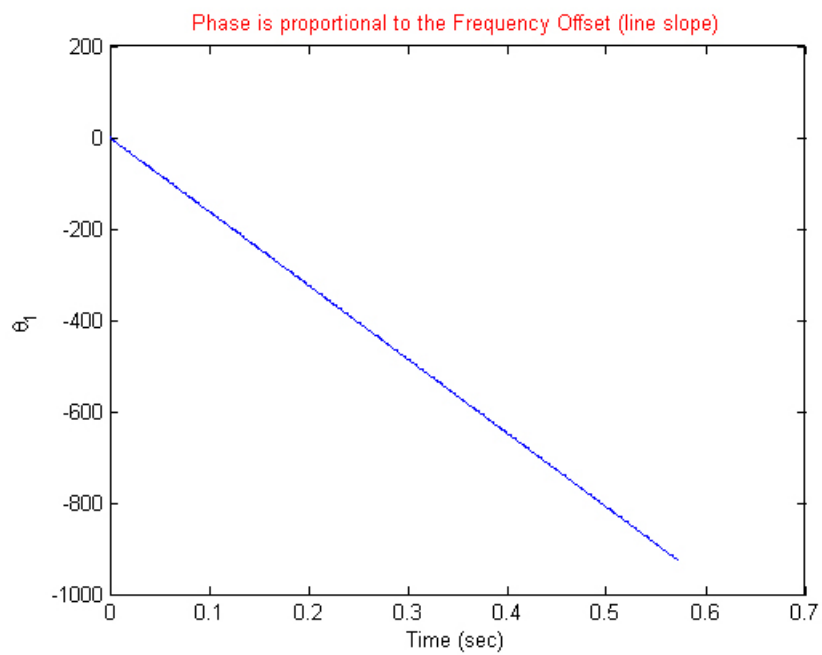


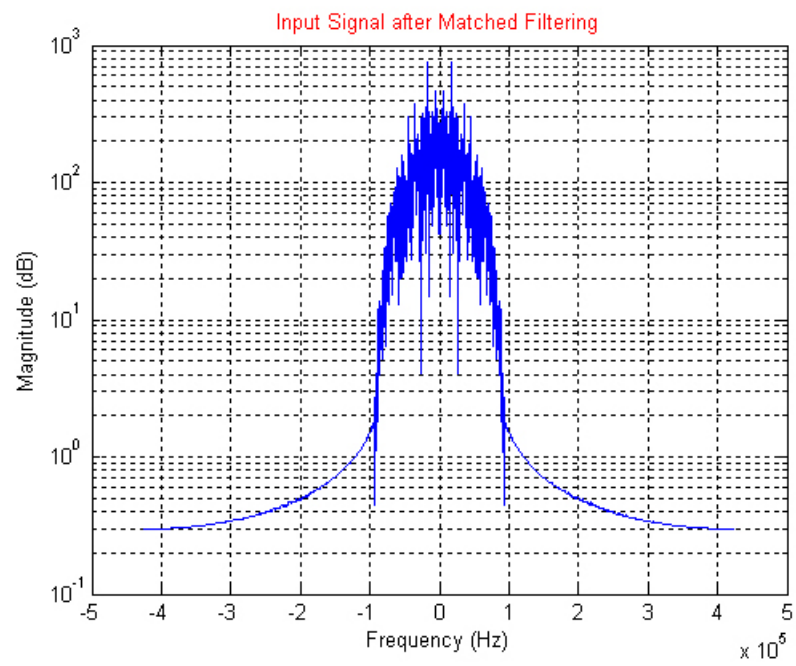
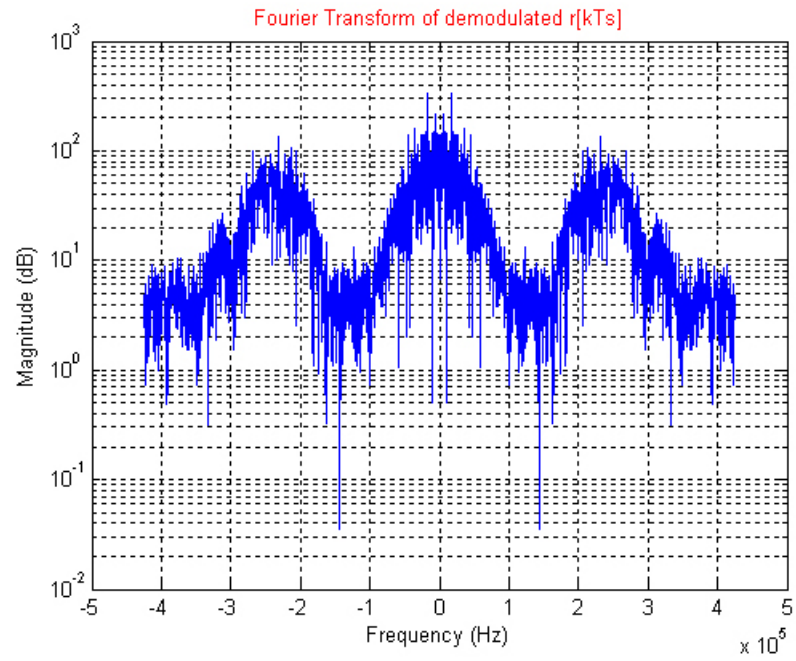
- medium1.mat
  - Φάσμα Λαμβανόμενου Σήματος





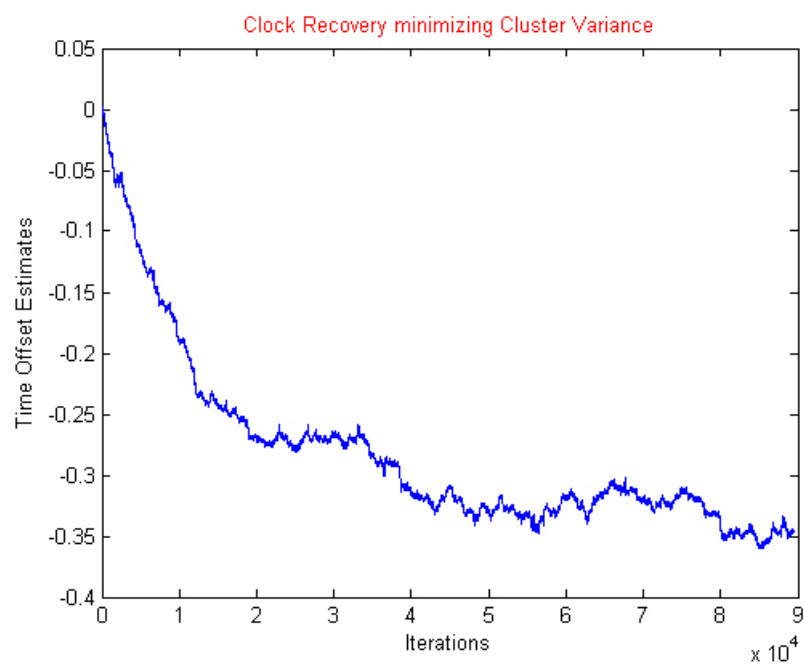
- Ανάκτηση Φορέα με χρήση Βρόχου Costas & Προσαρμοσμένο Φιλτράρισμα



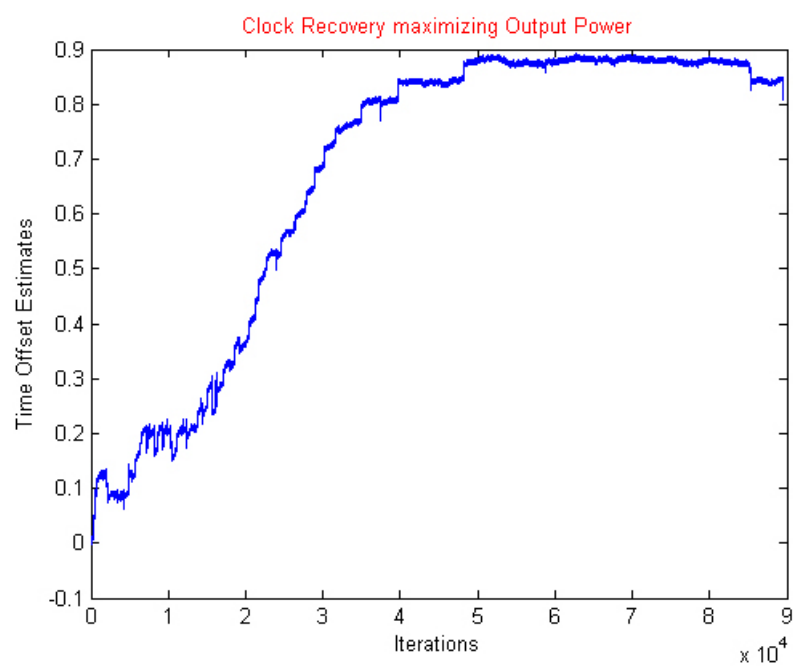


– Ανάκτηση Συγχρονισμού

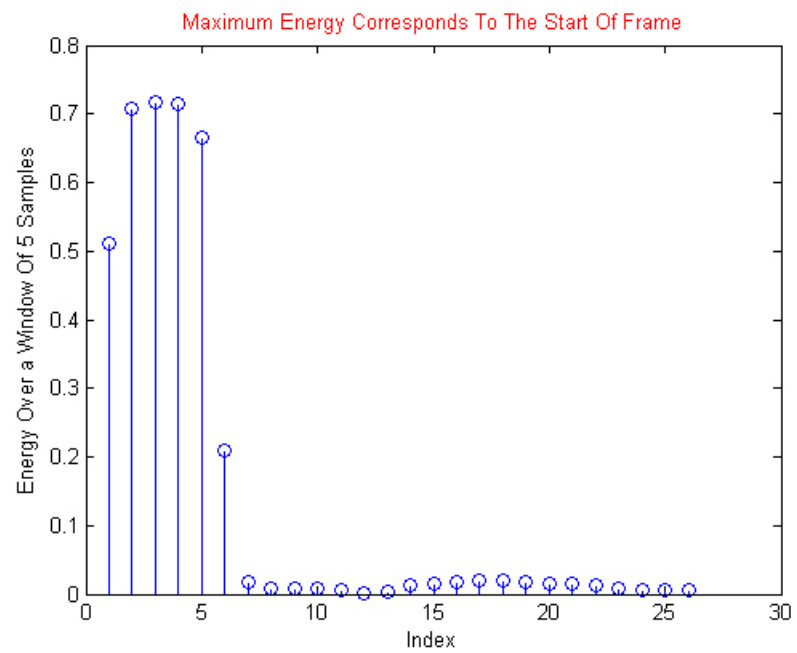
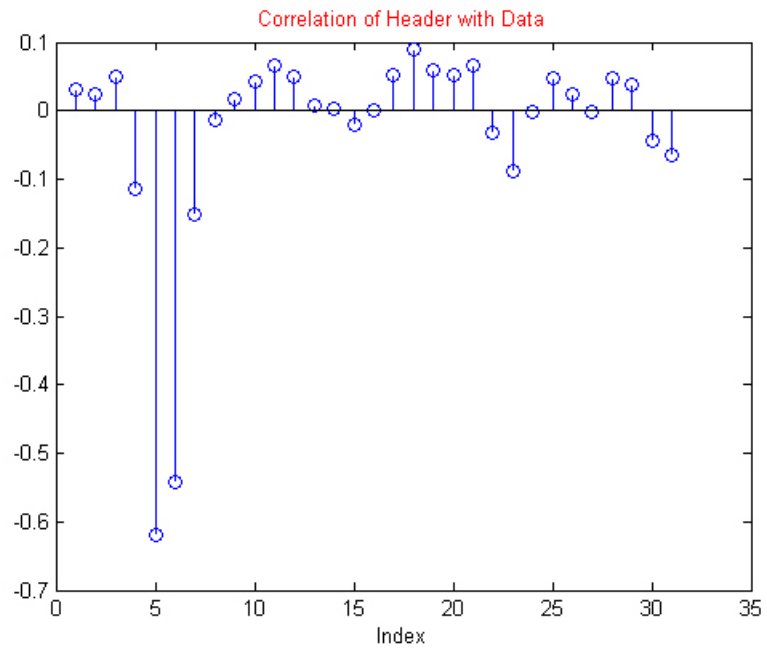
\* Ελαχιστοποίηση του Cluster Variance



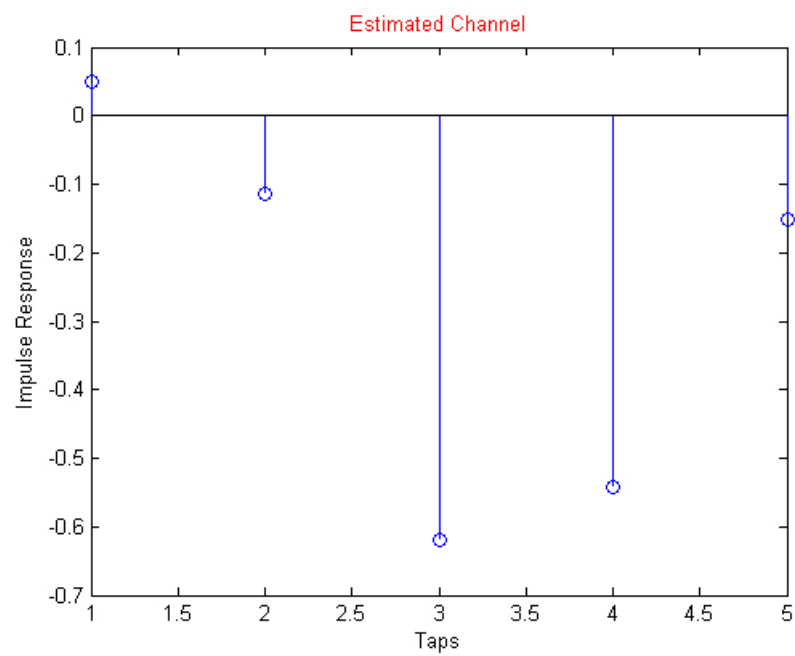
\* Μεγιστοποίηση της Ισχύος Εξόδου



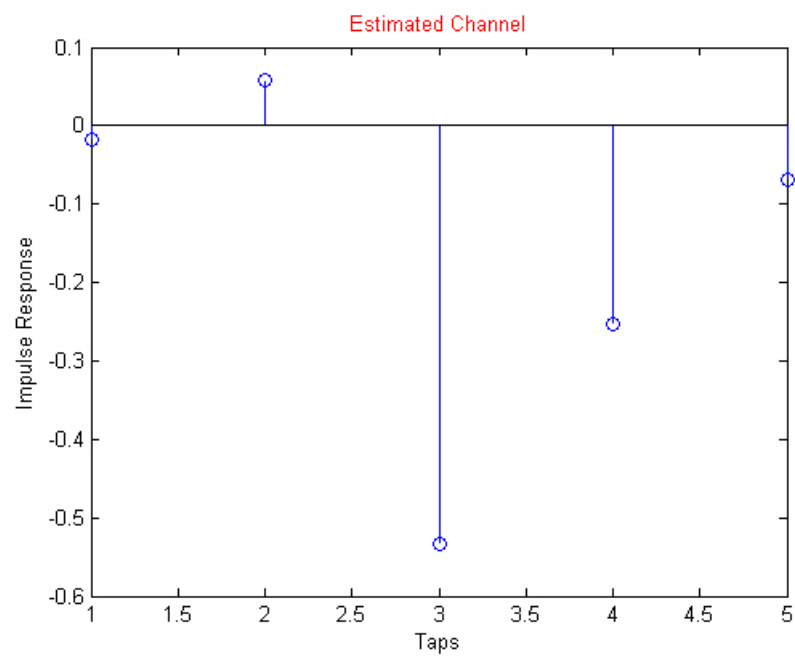
## – Εκτίμηση Καναλιού



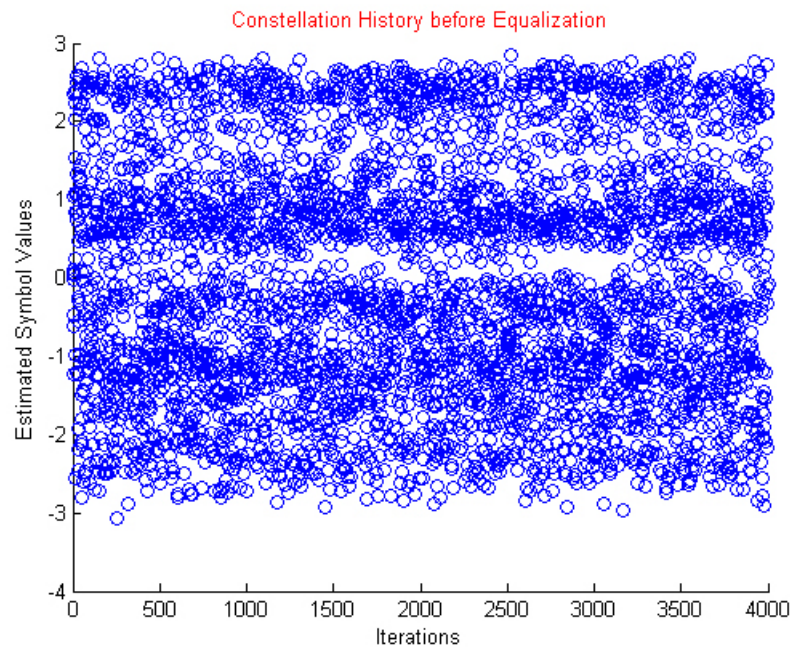
\* Μέθοδος Ετεροσυσχέτισης



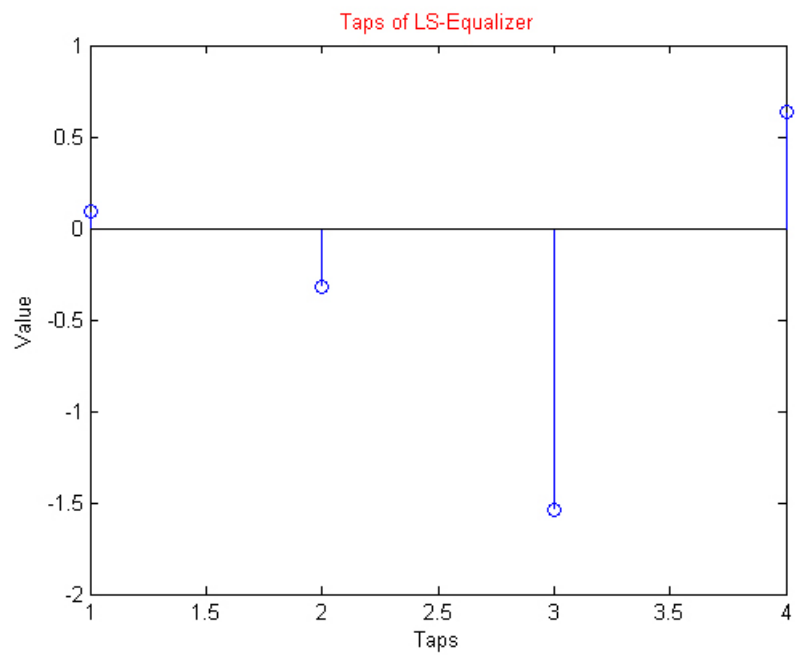
\* Μέθοδος Ελαχίστων Τετραγώνων (LS)



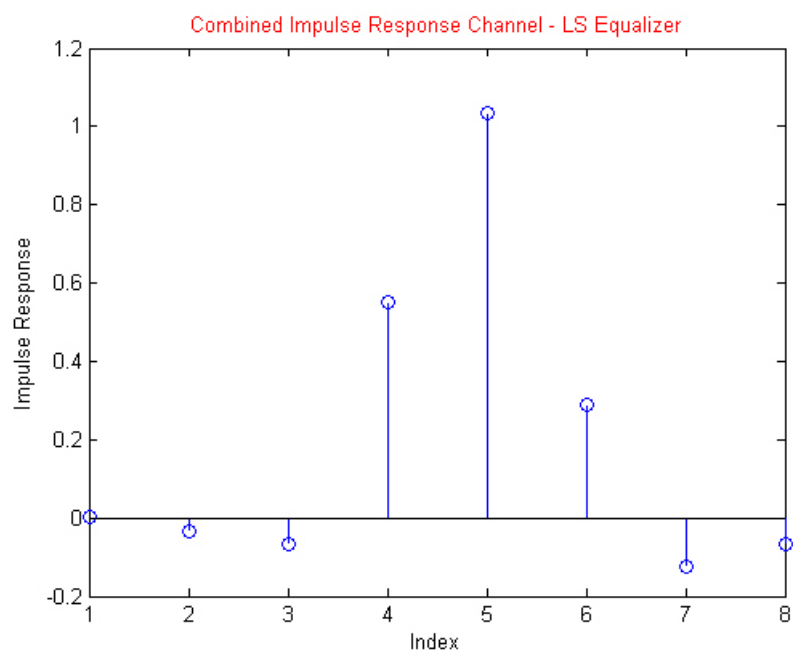
– Ισοστάθμιση Καναλιού



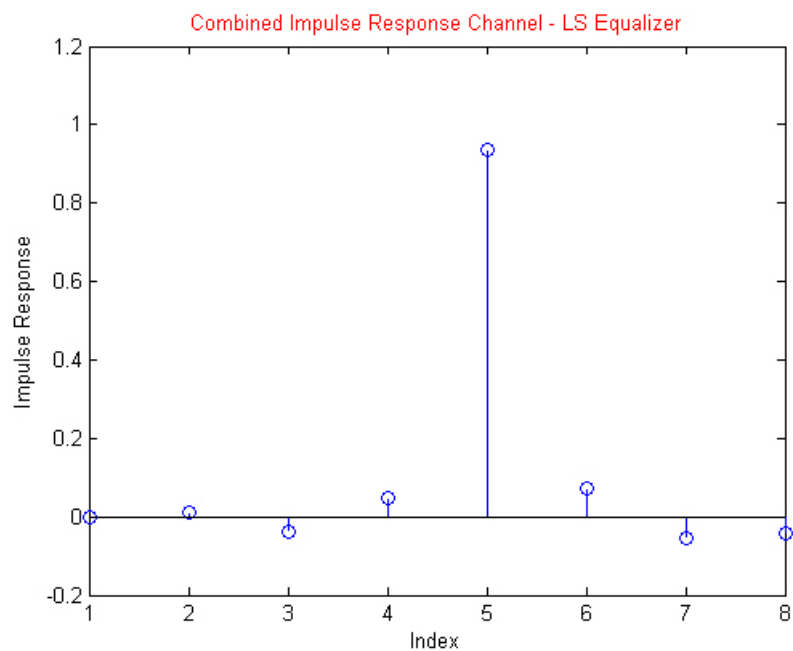
\* Μέθοδος Ελαχίστων Τετραγώνων (LS)

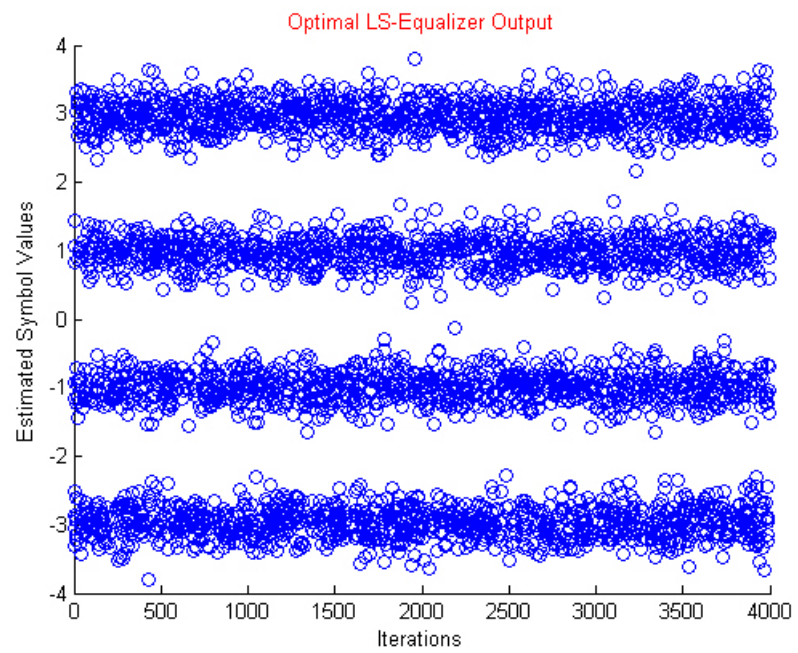


\* Για Εκτίμηση Καναλιού με την Μέθοδο της Ετεροσυσχέτισης

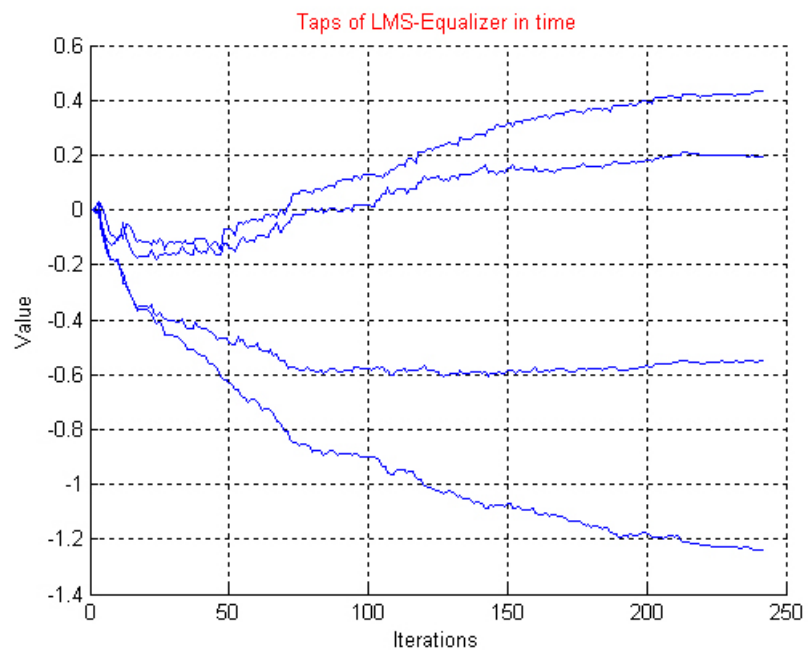


\* Για Εκτίμηση Καναλιού με Λύση Ελαχίστων Τετραγώνων



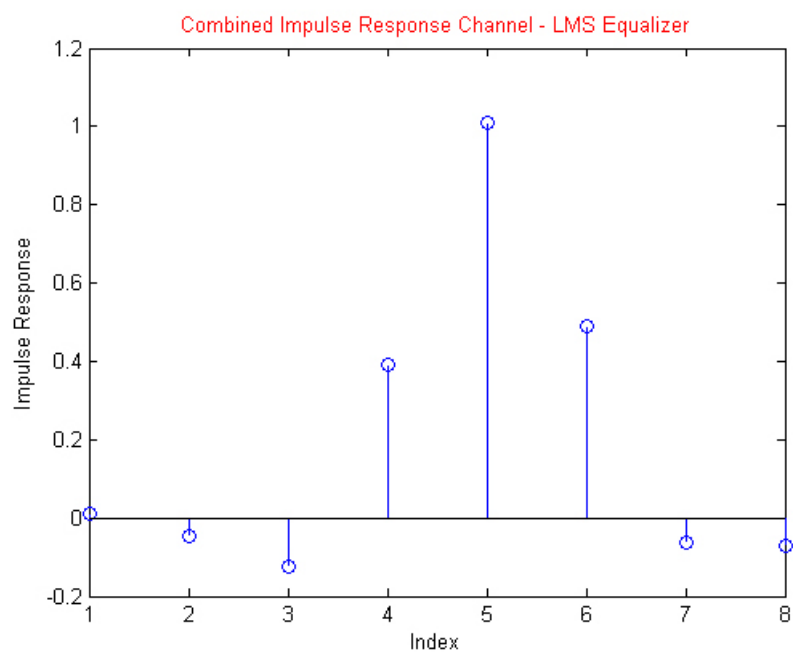


\* Προσαρμοστική Ισοστάθμιση - Αλγόριθμος LMS

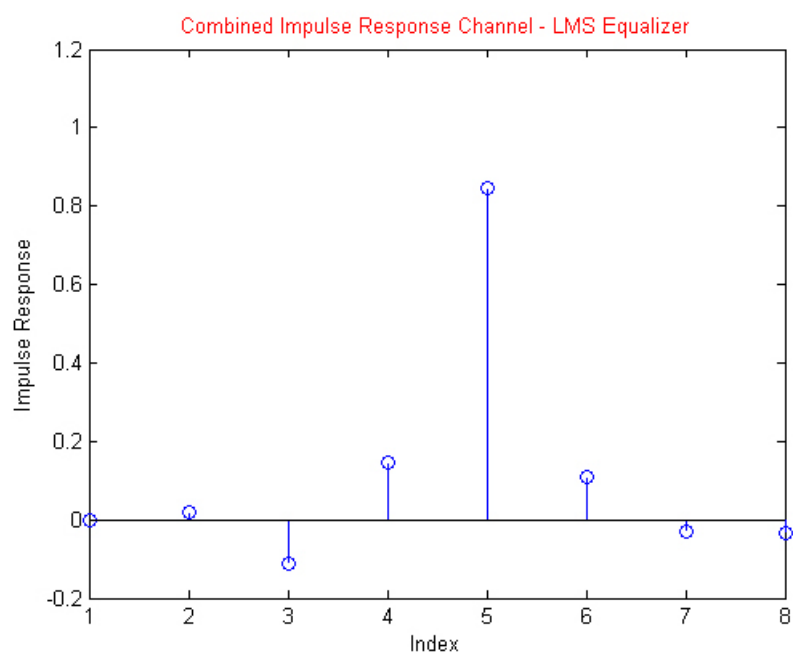


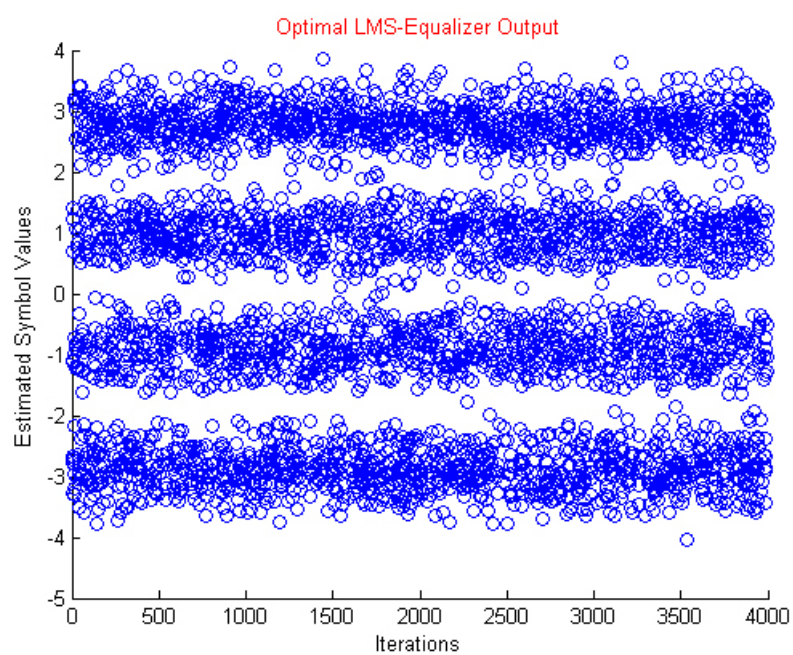


\* Για Εκτίμηση Καναλιού με την Μέθοδο της Ετεροσυσχέτισης



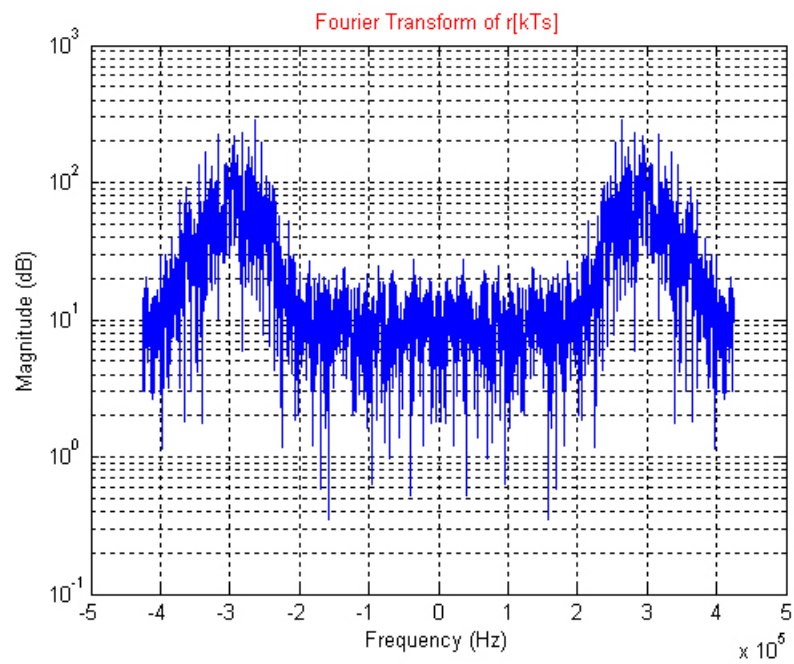
\* Για Εκτίμηση Καναλιού με Λύση Ελαχίστων Τετραγώνων



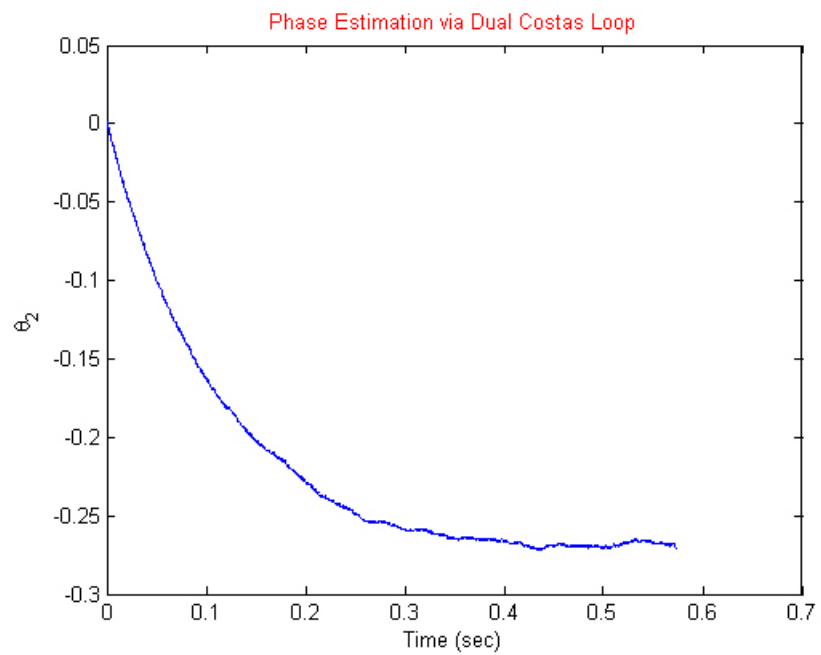
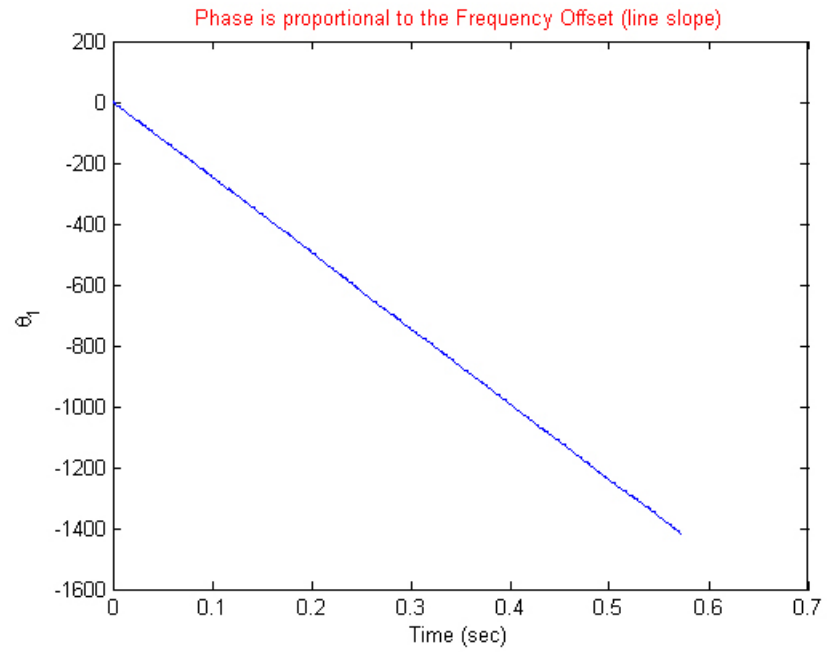


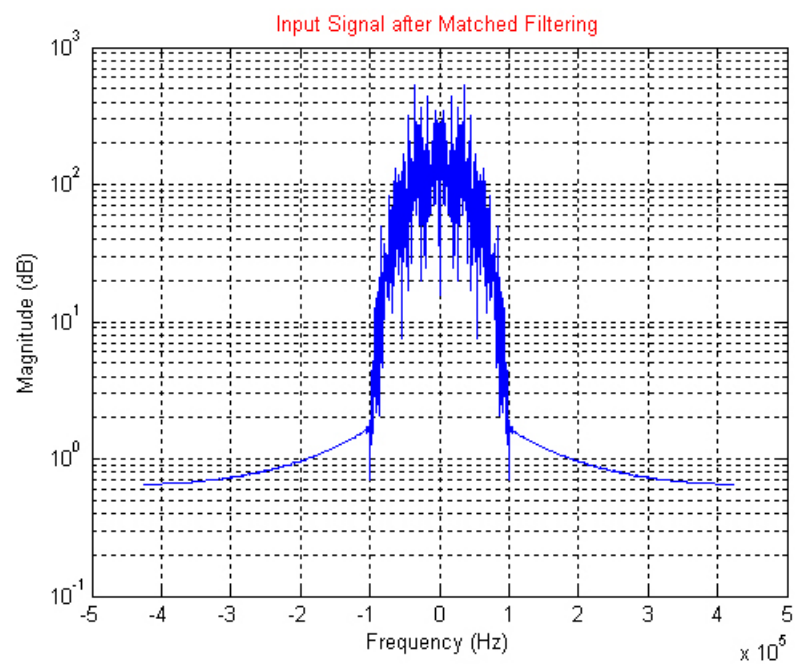
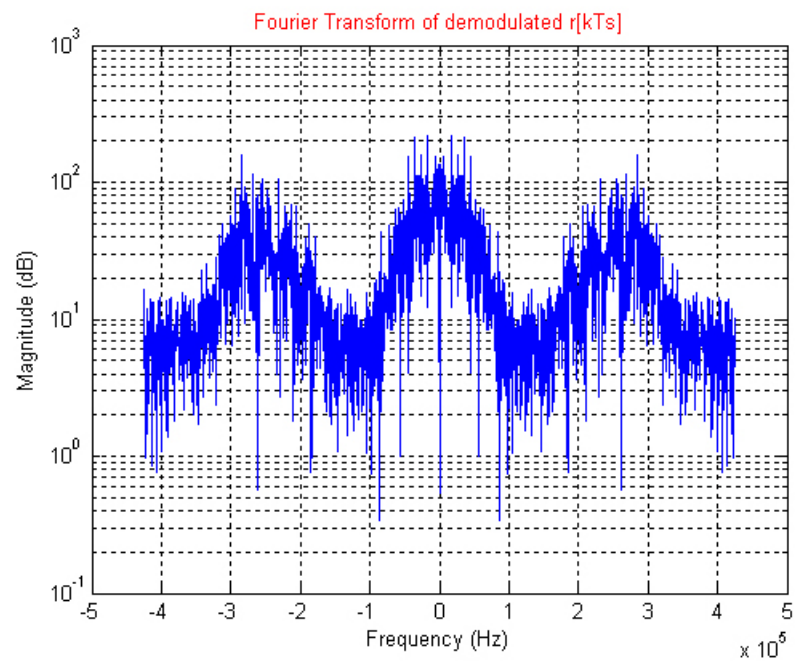
- hard1.mat

- Φάσμα Λαμβανόμενου Σήματος

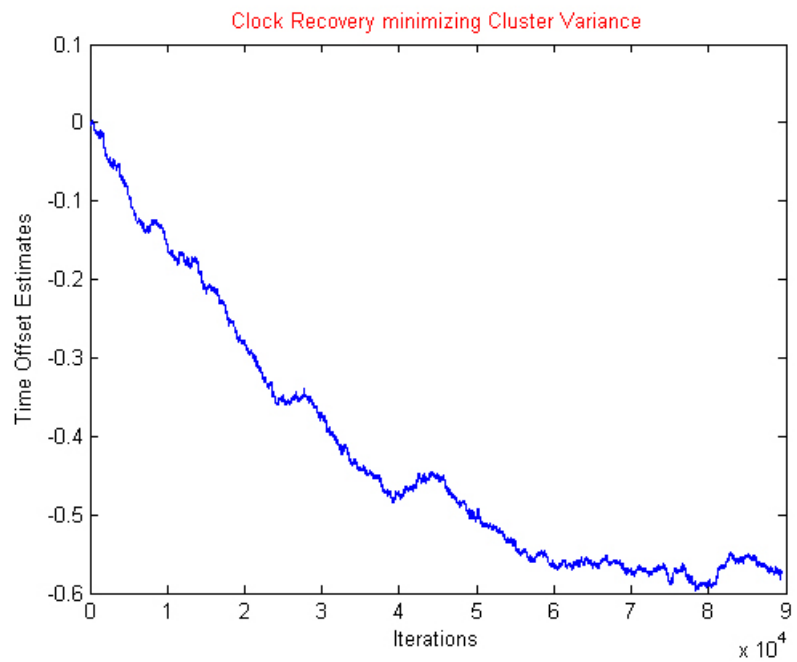


- Ανάκτηση Φορέα με χρήση Βρόχου Costas & Προσαρμοσμένο Φιλτράρισμα

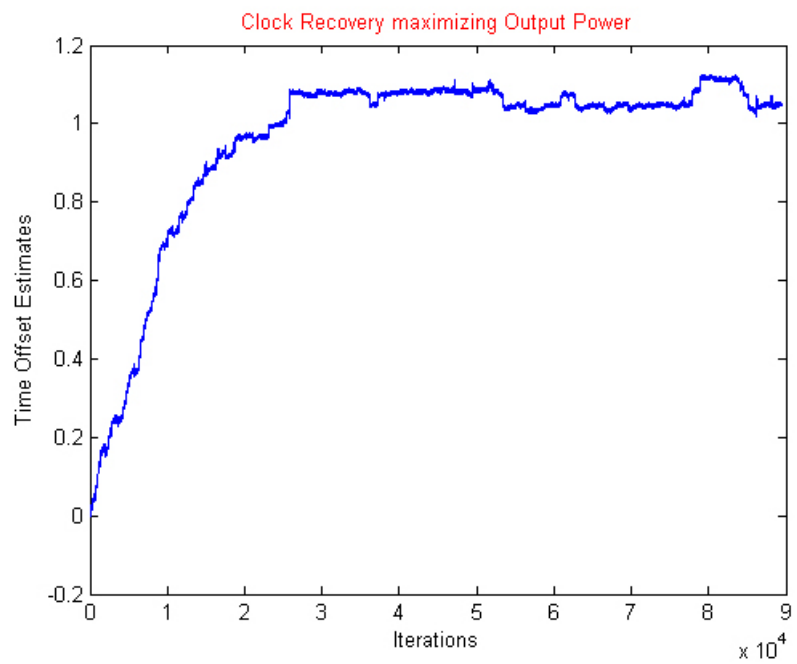




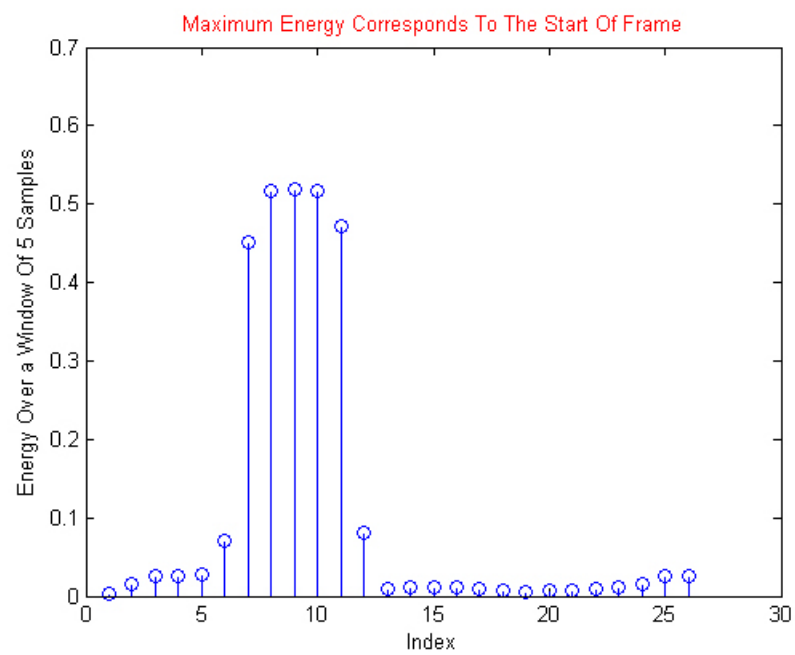
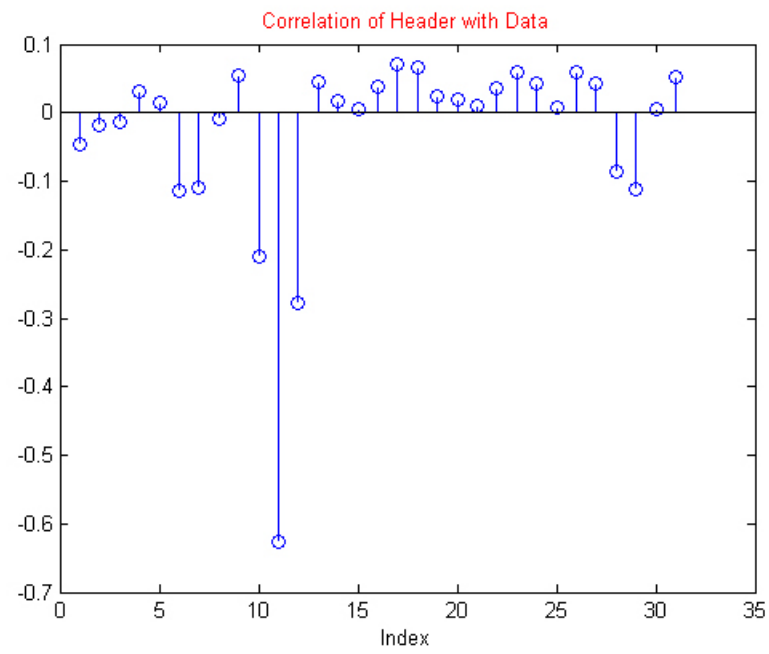
- Ανάκτηση Συγχρονισμού
  - \* Ελαχιστοποίηση του Cluster Variance



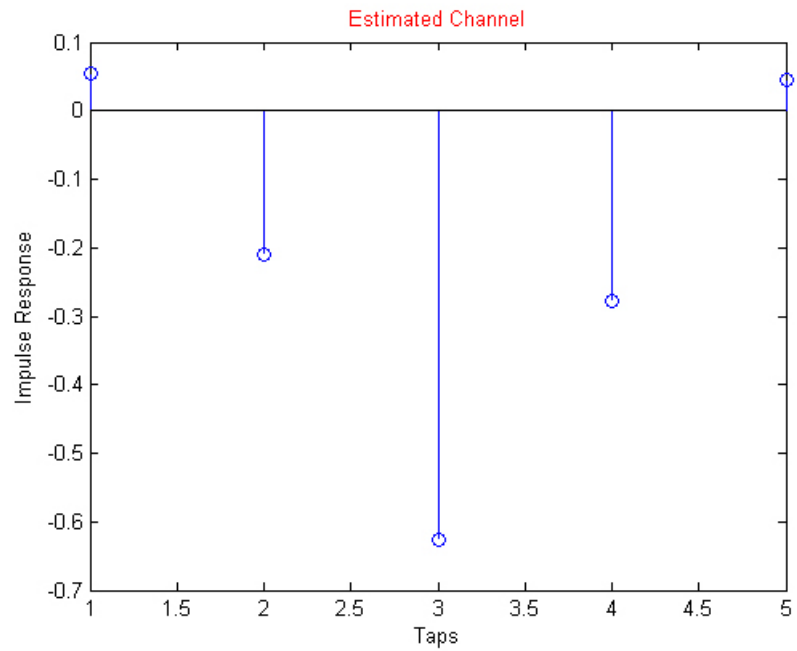
- \* Μεγιστοποίηση της Ισχύος Εξόδου



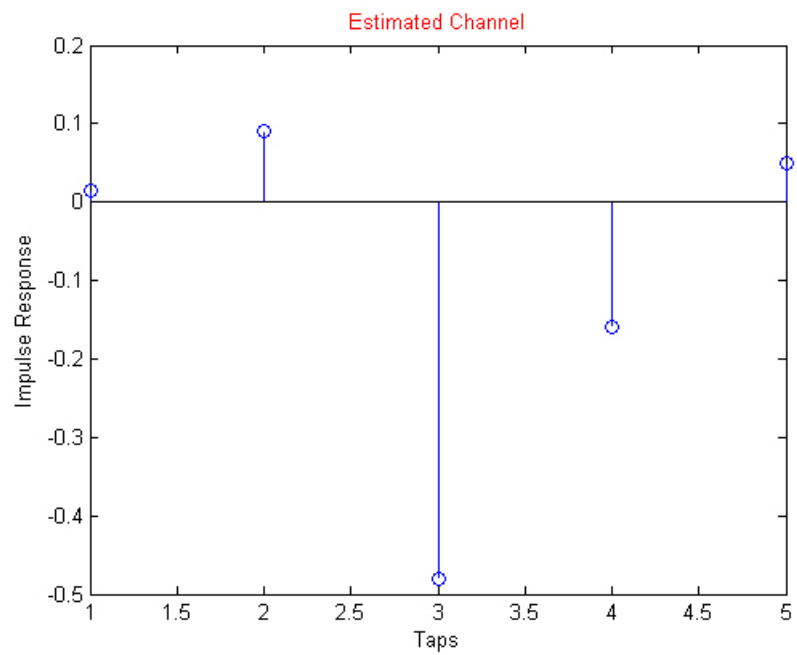
## – Εκτίμηση Καναλιού



## \* Μέθοδος Ετεροσυσχέτισης

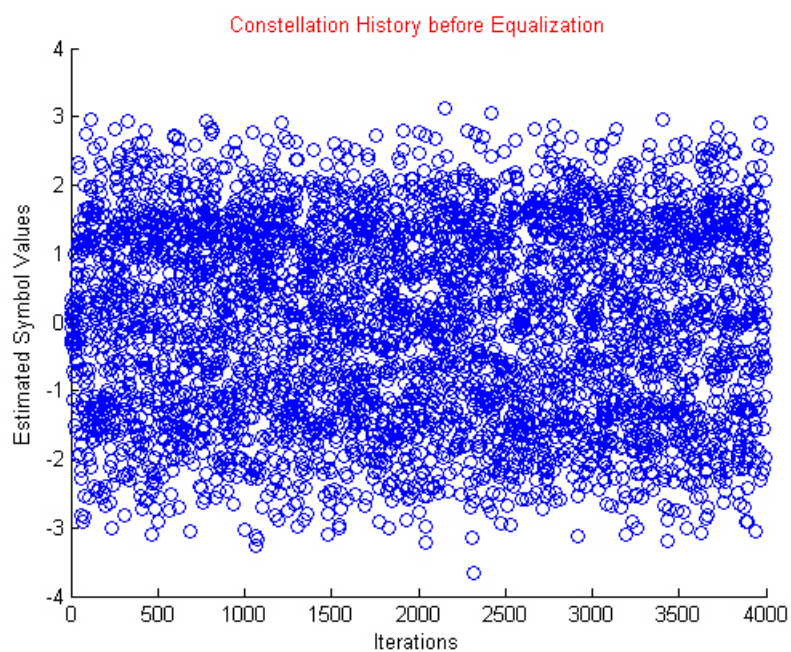


## \* Μέθοδος Ελαχίστων Τετραγώνων (LS)

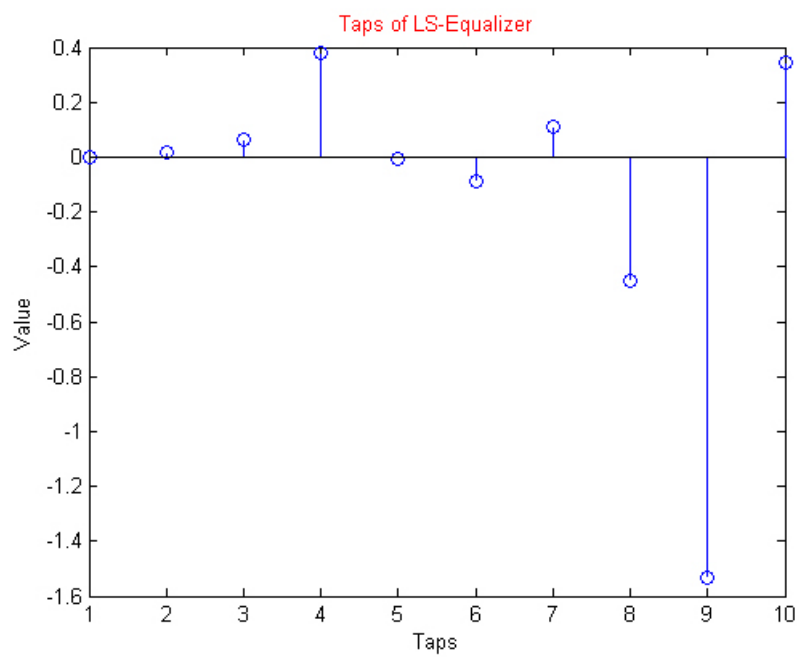




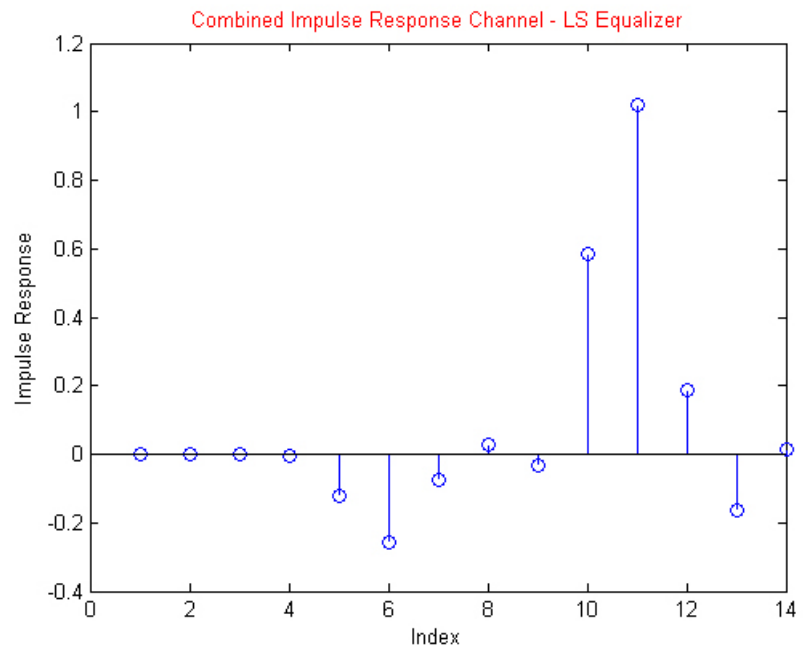
– Ισοστάθμιση Καναλιού



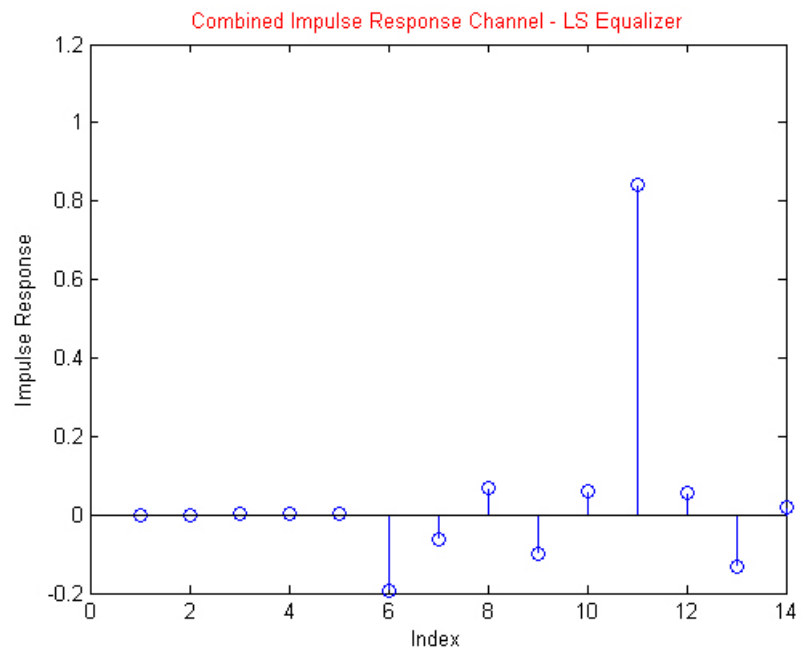
\* Μέθοδος Ελαχίστων Τετραγώνων (LS)

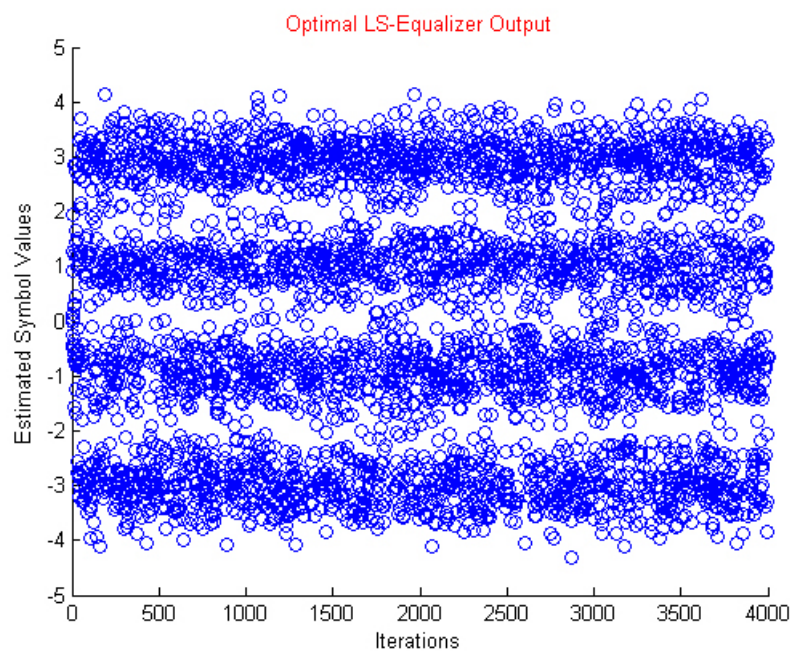


\* Για Εκτίμηση Καναλιού με την Μέθοδο της Ετεροσυσχέτισης

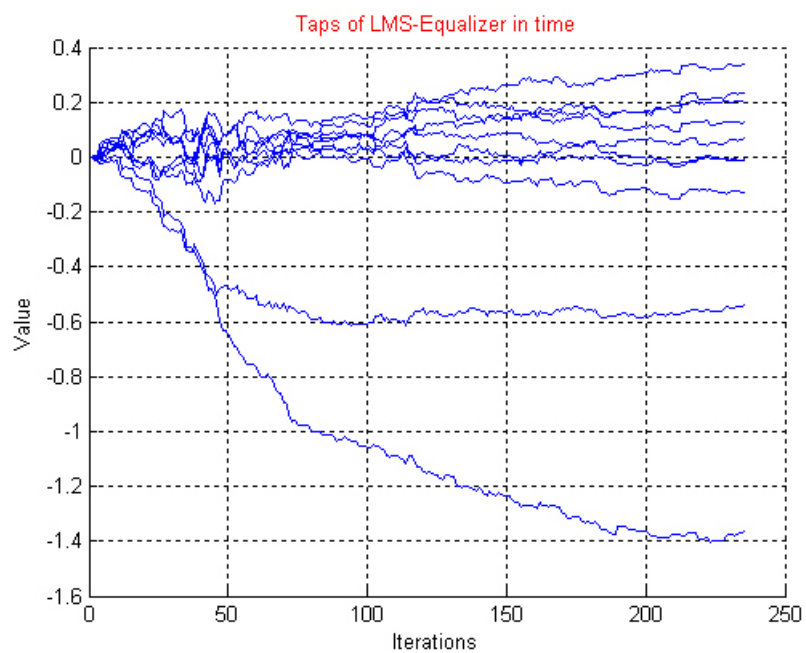


\* Για Εκτίμηση Καναλιού με Λύση Ελαχίστων Τετραγώνων

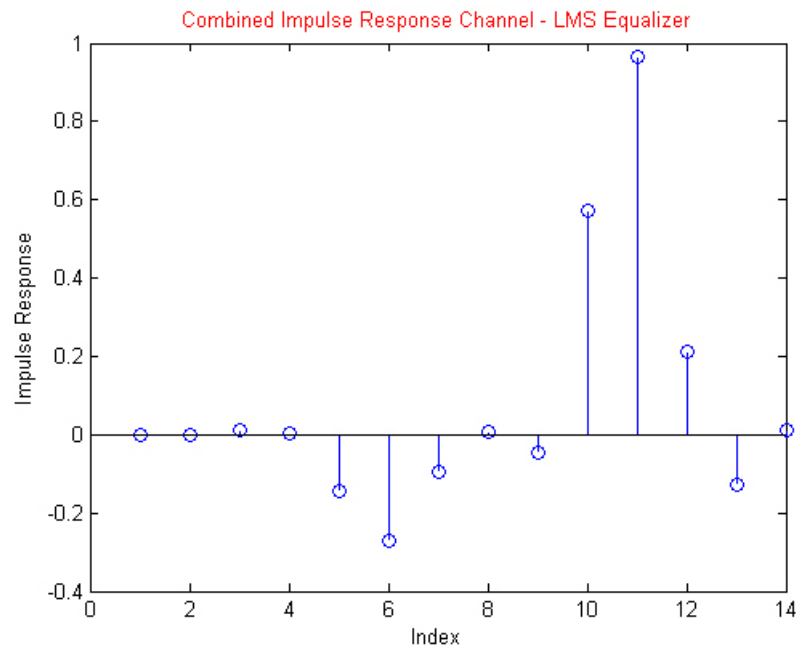




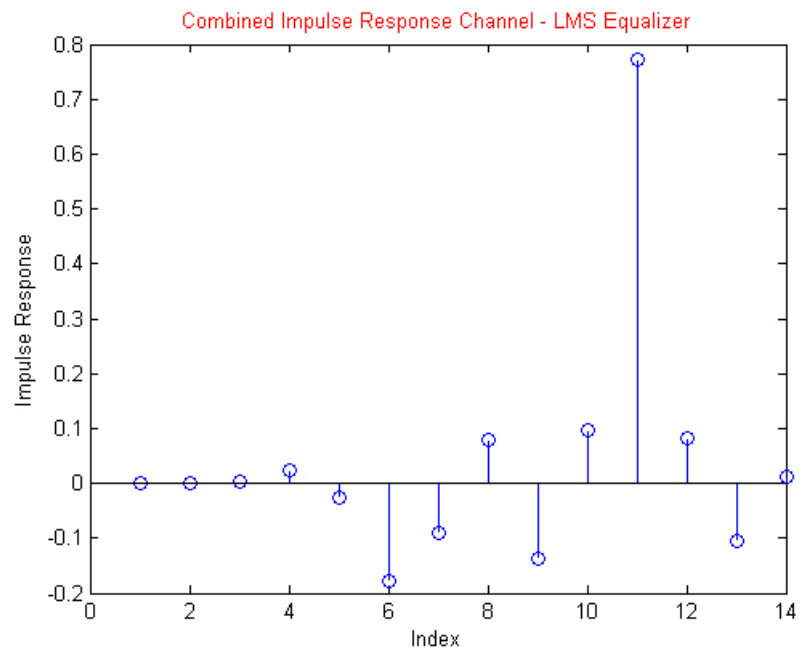
\* Προσαρμοστική Ισοστάθμιση - Αλγόριθμος LMS

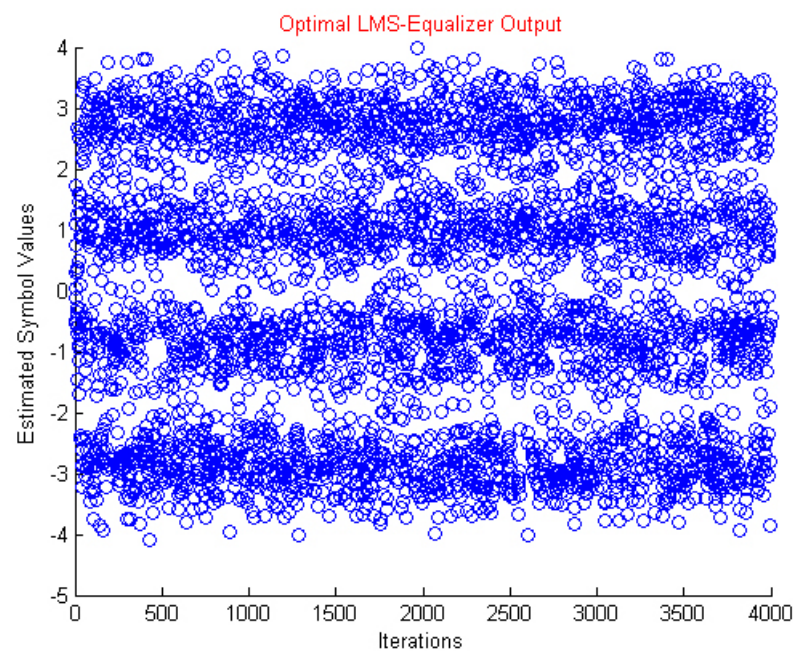


\* Για Εκτίμηση Καναλιού με την Μέθοδο της Ετεροσυσχέτισης



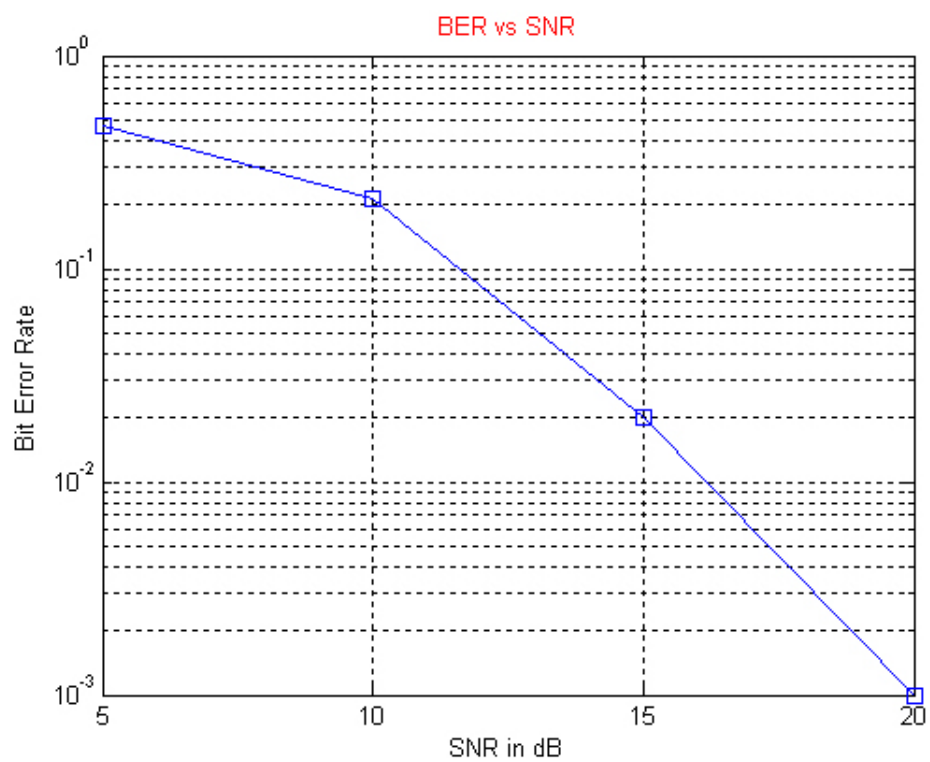
\* Για Εκτίμηση Καναλιού με Λύση Ελαχίστων Τετραγώνων





### Παρατηρήσεις - Σχόλια

Κατ'αρχήν στο **easy1.mat** για την ανάκτηση φορέα χρησιμοποιήθηκε απλός βρόχος Costas διότι δεν υπάρχει απόκλιση στη συχνότητα και όπως φαίνεται από το αντίστοιχο γράφημα, ο αλγόριθμος συγκλίνει γρήγορα στην εκτιμώμενη φάση. Αντιθέτως, τόσο στο **medium1.mat** όσο και στο **hard1.mat** έχει χρησιμοποιηθεί διπλός βρόχος Costas διότι υπάρχει απόκλιση στη συχνότητα του φέροντος. Στις δύο αυτές περιπτώσεις έχει προστεθεί ακόμη ένα γράφημα, το οποίο δείχνει μια ευθεία, η κλίση της οποίας δίνει τη διαφορά συχνότητας. Έτσι, ο αλγόριθμος χρησιμοποιώντας αυτή τη διαφορά συχνότητας συγκλίνει στην εκτιμώμενη φάση του φέροντος. Στη συνέχεια, για το πρώτο αρχείο, παρατηρούμε ότι οι δύο αλγόριθμοι που χρησιμοποιήθηκαν για την ανάκτηση χρονισμού συγκλίνουν στην ίδια τιμή ενώ για τα άλλα δύο αρχεία, δεν συμβαίνει αυτό. Αυτό οφείλεται στο γεγονός ότι στην περίπτωση του **easy1.mat** χρησιμοποιήθηκε απλό κανάλι που εισάγει απλά μια καθυστέρηση (μοναδιαίο κανάλι) και έχουμε σχεδόν ιδανικές συνθήκες, ενώ στις άλλες δύο περιπτώσεις έχουμε ενθόρυβα περιβάλλοντα με διασυμβολική παρεμβολή. Επιπλέον, ο αλγόριθμος μεγιστοποίησης της ισχύος εξόδου λειτουργεί καλύτερα σε σχέση με την μέθοδο ελαχιστοποίησης του cluster variance σε ενθόρυβο περιβάλλον, διότι η μεγιστοποίηση της ισχύος εξόδου προέρχεται από μεγιστοποίηση της ισχύος του χρήσιμου σήματος εξαιτίας του γεγονότος ότι ο θόρυβος έχει σταθερή ισχύ στο χρόνο. Στη συνέχεια παρατίθενται τα γραφήματα που παρουσιάζουν μια εκτίμηση του καναλιού με τη μέθοδο της ετεροσυσχέτισης και της λύσης των ελαχίστων τετραγώνων, όπου η τελευταία είναι η βέλτιστη εκτίμηση. Κατόπιν, παρουσιάζονται τα γραφήματα των ισοσταθμιστών που χρησιμοποιήθηκαν. Στην περίπτωση του ισοσταθμιστή LMS, χρησιμοποιήσαμε τα taps μετά τη σύγκλιση του προσαρμοστικού αλγορίθμου, για να φιλτράρουμε ολόκληρο το σήμα. Γενικότερα ο ισοσταθμιστής LS λειτουργεί καλύτερα από τον LMS. Παρατηρούμε ότι και στους δύο ισοσταθμιστές τα νέφη γύρω από τις τιμές των συμβόλων διαχέονται περισσότερο όσο η επεξεργασία του σήματος γίνεται σε περισσότερο ενθόρυβο περιβάλλον, με περισσότερες αποκλίσεις και διασυμβολική παρεμβολή. Τέλος, να αναφέρουμε ότι υλοποιήθηκε ο αλγόριθμος Viterbi για την ισοστάθμιση του καναλιού. Για να δείξουμε τη σωστή λειτουργία του αλγορίθμου, χρησιμοποιήσαμε τον αλγόριθμο σε τυχαίο κανάλι μήκους  $L = 5$ , με τυχαία σύμβολα 4-PAM αστερισμού  $(-3, -1, +1, +3)$ , πλήθους  $N = 1000$ , για τιμές του  $SNR = 5, 10, 15, 20dB$ .



Σχήμα 6.1 BER για Ισοστάθμιση Καναλιού με Αλγόριθμο Viterbi

# Παράρτημα Α

## Αποκωδικοποίηση

Στην παρούσα διπλωματική εργασία, η κωδικοποίηση του καναλιού έγινε χρησιμοποιώντας *Δυαδικό Γραμμικό Μπλοκ Κώδικα (5,2)*. Ο γεννήτορας πίνακας είναι [1]

$$G = \begin{bmatrix} 1 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 1 & 1 \end{bmatrix}$$

και ο πίνακας ελέγχου ισοτιμίας [1]

$$H^T = \begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Αυτός ο κώδικας, ομαδοποιεί τα bits σε ζεύγη και οι 4 κωδικές λέξεις είναι

$$x_1 = 00 \leftrightarrow c_1 = x_1 G = 00000$$

$$x_2 = 01 \leftrightarrow c_2 = x_2 G = 01011$$

$$x_3 = 10 \leftrightarrow c_3 = x_3 G = 10101$$

$$x_4 = 11 \leftrightarrow c_4 = x_4 G = 11110$$

Για τον υπολογισμό των κωδικών λέξεων χρησιμοποιήθηκε δυαδική αριθμητική και τελικά τα αποτελέσματα είναι σε δυαδική μορφή. Μετά την μετάδοση το λαμβανόμενο σήμα  $y$  πολλαπλασιάζεται με τον πίνακα  $H^T$ . Αν δεν έχουν συμβεί λάθη κατά τη μετάδοση, τότε το  $y$  είναι ίσο με μία από τις 4 κωδικές λέξεις  $c_i$ . Χρησιμοποιώντας τον πίνακα  $H^T$  όπως έχει οριστεί, έχουμε  $c_1 H^T = c_2 H^T = c_3 H^T = c_4 H^T = 0$ . Έτσι,  $y H^T = 0$ . Στην περίπτωση που έχουμε λάθος,  $y H^T \neq 0$  και αυτή η τιμή μπορεί να χρησιμοποιηθεί για να καθοριστεί το πιο πιθανό λάθος [1]. Για να γίνει πιο κατανοητό αυτό, ξαναγράφουμε

$$y = c + (y - c) \equiv c + e$$



όπου  $e$  συμβολίζει το λάθος που έχει συμβεί στην μετάδοση. Να σημειωθεί ότι

$$yH^T = (c + e)H^T = cH^T + eH^T = eH^T$$

καθώς  $cH^T = 0$  [1]. Η τιμή του  $eH^T$  μπορεί να χρησιμοποιηθεί εξετάζοντας τον πίνακα συνδρόμου, όπως στο σχήμα Α.1.

Για παράδειγμα, ας υποθέσουμε ότι το σύμβολο που μεταδόθηκε είναι το  $x_2 = 01$ , τότε η αντίστοιχη κωδική λέξη είναι η  $c_2 = 01011$ . Επίσης, υποθέτουμε ότι κατά την μετάδοση συνέβησαν λάθη. Έτσι, λάβαμε το  $y = 11011$ . Πολλαπλασιασμός του  $y$  με τον πίνακα ελέγχου ισοτιμίας δίνει  $yH^T = eH^T = 101$ . Κοιτάζοντας τον πίνακα συνδρόμου φαίνεται ότι το πιο πιθανό λάθος είναι το 10000. Σύμφωνα με αυτό, η κωδική λέξη που είναι πιθανότερο να έχει μεταδοθεί, είναι η  $y - e = 11011 - 10000 = 01011$ , που όντως μας δίνει τη σωστή κωδική λέξη  $c_2$ .

Από την άλλη πλευρά, αν περισσότερα από ένα λάθη συμβούν κατά τη μετάδοση, τότε ο *Γραμμικός Κώδικας (5,2)* δεν βρίσκει απαραίτητα τη σωστή κωδική λέξη. Για παράδειγμα, υποθέτουμε ότι το σύμβολο που μεταδόθηκε είναι το  $x_2 = 01$  που αντιστοιχεί στην κωδική λέξη  $c_2 = 01011$ , αλλά 2 λάθη συνέβησαν κατά την μετάδοση. Έτσι, η λαμβανόμενη λέξη είναι η  $y = 00111$ . Ο πολλαπλασιασμός με τον πίνακα ελέγχου ισοτιμίας δίνει  $yH^T = eH^T = 111$ . Κοιτάζοντας ξανά στον πίνακα συνδρόμου, φαίνεται το πιθανότερο λάθος να είναι το 10010. Σύμφωνα με αυτό, η κωδική λέξη που είναι πιθανότερο να έχει μεταδοθεί, είναι η  $y - e = 00111 - 10010 = 10101$ , το οποίο αντιστοιχεί στην κωδική λέξη  $c_3$ , που αντιστοιχεί στο σύμβολο  $x_3$  και όχι στο  $x_2$ . [1]

Ο πίνακας συνδρόμου μπορεί να δημιουργηθεί ως ακολούθως: πρώτα, παίρνουμε κάθε πιθανό λάθος  $e$  (ένας άσσος σε κάθε μία από 5 θέσεις, 00001, 00010 κλπ), και υπολογίζουμε το  $eH^T$  για το καθένα. Από τη στιγμή που οι στήλες του  $H$  είναι μη μηδενικές και διακριτές, τότε κάθε λάθος αντιστοιχεί σε διαφορετικό σύνδρομο. Για να συμπληρωθεί ο πίνακας, παίρνουμε όλα τα πιθανά διπλά λάθη και υπολογίζουμε το  $eH^T$ . Διαλέγουμε τα 2, που αντιστοιχούν στα 2 εναπομείνοντα σύνδρομα. [1]

Syndrome $\mathbf{e}H^T$	Most likely error $\mathbf{e}$
000	00000
001	00001
010	00010
011	01000
100	00100
101	10000
110	11000
111	10010

Σχήμα Α.1 Πίνακας Συνδρόμου

## Παράρτημα Β

# Πλήρης Υλοποίηση του Συστήματος Προσομοίωσης σε περιβάλλον Matlab

```
% Technical University of Crete, Fall 2005 - Spring 2006
% Telecommunications Division
% Digital Communication Systems
% Ioannis P. Lazaridis - Sotirios A. Vlachodimitropoulos
% SID: 2000030071 - 2000030055
% Directed by Dr. Athanasios P. Liavas
```

```
% Carrier Recovery Function
function dem_r = carrierRecovery(r,k,crAlgorithm,crAlgParameter,freqOffset)
```

```
% dem_r = carrierRecovery(matFile,crAlgorithm,crAlgParameter)
%      'r' is the received signal to process
%      'k' is the length of r (number of samples)
%      'crAlgorithm' is the choice of the Carrier
%              Recovery Algorithm shall use
%      'crAlgParameter' is the choice of the order of Algorithm Loop
%              (1->Single Loop, 2->Dual Loop)
%      'freqOffset' is the assumed clock frequency at the receiver
```

```
Ts = 1/(850*10^3); % Sampling period
time = Ts*k;      % Total time
t = 0:Ts:time-Ts; % Time vector
N_F = 2000;       % Number of samples for Fourier Transform
ssf = (-N_F/2:N_F/2-1)/(Ts*N_F); % Frequency Axis(-Fs/2,Fs/2)
```

```

% Case of Dual Costas Loop
if(crAlgorithm == 1 && crAlgParameter == 2)
    fl = 70; fbe = [0 102 112 425]/425; damp = [1 1 0 0];
    lpfc = remez(fl,fbe,damp); % Low-pass filter Design
    muc1 = 0.01; % Algorithm stepsize  $\mu_1$  for the top Costas loop
    muc2 = 0.00001; % Algorithm stepsize  $\mu_2$  for the bottom Costas loop
    f0 = 300*103; % Clock frequency at the transmitter
    fc = freqOffset; % Assumed clock frequency at the receiver

    % Initialize estimate vectors and buffers...
    theta1 = zeros(1,k); theta1(1) = 0;
    theta2 = zeros(1,k); theta2(1) = 0;
    zs1 = zeros(1,fl+1); zc1 = zeros(1,fl+1);
    zs2 = zeros(1,fl+1); zc2 = zeros(1,fl+1);

    % Adaptive algorithm to estimate the phase and frequency offsets
    for i=1:k-1
        % Top Costas Loop

        % z1's contain past fl+1 inputs
        zs1 = [zs1(2:fl+1), 2*r(i)*sin(2*pi*fc*t(i) + theta1(i))];
        zc1 = [zc1(2:fl+1), 2*r(i)*cos(2*pi*fc*t(i) + theta1(i))];
        % New output of filters
        lpfcs1 = fliplr(lpfc)*zs1'; lpfcc1=fliplr(lpfc)*zc1';
        % Algorithm update
        theta1(i+1) = theta1(i) - muc1*lpfcs1*lpfcc1;

        % Bottom Costas Loop

        % z2's contain past fl+1 inputs
        zs2=[zs2(2:fl+1), 2*r(i)*sin(2*pi*fc*t(i)+theta1(i)+theta2(i))];
        zc2=[zc2(2:fl+1), 2*r(i)*cos(2*pi*fc*t(i)+theta1(i)+theta2(i))];
        % New output of filters
        lpfcs2 = fliplr(lpfc)*zs2'; lpfcc2 = fliplr(lpfc)*zc2';
        % Algorithm update
        theta2(i+1) = theta2(i) - muc2*lpfcs2*lpfcc2;
    end
    % The estimated sinusoid to demodulate
    carest = cos(2*pi*fc*t + theta1 + theta2(end));

```

```

% Case of Single Costas Loop
elseif(crAlgorithm == 1 && crAlgParameter == 1)
    fl = 70; fbe=[0 102 112 425]/425; damp = [1 1 0 0];
    lpfc = remez(fl,fbe,damp); % Low-pass filter Design
    muc = 0.0001; % Algorithm stepsize  $\mu$ 
    fc = 300*103; % Clock frequency at the transmitter is the same
                    % with the clock frequency at the receiver
                    % (there is no frequency-offset)

    % Initialize estimate vectors and buffers...
    theta = zeros(1,k); theta(1) = 0;
    zs = zeros(1,fl+1); zc = zeros(1,fl+1);

    % Adaptive algorithm to estimate the phase offset
    for i=1:k-1
        % z's contain past fl+1 inputs
        zs = [zs(2:fl+1),2*r(i)*sin(2*pi*fc*t(i) + theta(i))];
        zc = [zc(2:fl+1),2*r(i)*cos(2*pi*fc*t(i) + theta(i))];
        % New output of filters
        lpfcs = fliplr(lpfc)*zs'; lpfcc = fliplr(lpfc)*zc';
        % Algorithm update
        theta(i+1) = theta(i) - muc*lpfcs*lpfcc;
    end
    % The estimated sinusoid to demodulate
    carest = cos(2*pi*fc*t + theta(end));
end

% Demodulation (Downconversion)
f_r = abs(fft(r)); % Fourier Transform of received signal r

figure();
plot(ssf,abs(fftshift(fft(r,N_F))));
xlabel('Frequency (Hz)'); ylabel('Magnitude');
title('Fourier Transform of r[kTs]','Color','r');

dem_r = r.*carest'; % Demodulated r
f_dem_r = abs((fft(dem_r))); % Fourier Transform of demodulated
                             % received signal r

```

```
figure();
plot(ssf,abs(fftshift(fft(dem_r,N_F))));
xlabel('Frequency (Hz)'); ylabel('Magnitude');
title('Fourier Transform of demodulated r[kTs]','Color','r');

% Plot phase-offset
if(crAlgorithm == 1 && crAlgParameter == 1) % Single Costas Loop
    figure();
    plot(t,theta);
    xlabel('Time (sec)'); ylabel('\theta');
    title('Phase Estimation via Single Costas Loop','Color','r');

elseif(crAlgorithm == 1 && crAlgParameter == 2) % Dual Costas Loop
    figure();
    plot(t,theta1);
    xlabel('Time (sec)'); ylabel('\theta_1');
    title('Phase Estimation via Dual Costas Loop','Color','r');

    figure();
    plot(t,theta2);
    xlabel('Time (sec)'); ylabel('\theta_2');
end
```

```
% Technical University of Crete, Fall 2005 - Spring 2006
% Telecommunications Division
% Digital Communication Systems
% Ioannis P. Lazaridis - Sotirios A. Vlachodimitropoulos
% SID: 2000030071 - 2000030055
% Directed by Dr. Athanasios P. Liavas
```

```
% Timing Recovery Function
```

```
function xs = timingRecovery(dem_r, trAlgorithm, rolloffFactor, k)
```

```
% xs = timingRecovery(dem_r, trAlgorithm, rolloffFactor, k)
```

```
% 'dem_r' is the demodulated received signal
```

```
% (the output of the Carrier Recovery stage)
```

```
% 'trAlgorithm' is the choice of the
```

```
% Timing Recovery Algorithm shall use
```

```
% 'rolloffFactor' is the roll-off factor on SRRC and Interpolator
```

```
% 'k' is the number of samples to process
```

```
% Matched Filtering
```

```
T = 6.4*10^-6; % Nominal symbol period
```

```
Ts = 1/(850*10^3); % Sampling period
```

```
N_F = 2000; % Number of samples for Fourier Transform
```

```
ssf = (-N_F/2:N_F/2-1)/(Ts*N_F); % Frequency Axis (-Fs/2, Fs/2)
```

```
m = T/Ts; % Evaluate at m different points
```

```
% (Oversampling factor)
```

```
l = round(4*(T/Ts)); % One sided length of data to interpolate
```

```
% (in symbols)
```

```
% Make Square Root Raised Cosine pulse shape for matched filtering
```

```
matchfilt = srrc(l, rolloffFactor, m);
```

```
% Matched filtering via convolution
```

```
r_match = conv(dem_r, matchfilt);
```

```
% Disregard the tail of convolution
```

```
r_match(end-length(matchfilt)+2:end)=[];
```

```

% Clock Recovery minimizing Cluster Variance
if(trAlgorithm == 1)
    N = round(k*Ts/T)-2*1;          % Number of data points
    tnow = 1*m+1; tau=0; xs=zeros(1,N); % Initialize vectors and
                                     % variables
    tausave = zeros(1,N); tausave(1) = tau; i=0;
    mu = 0.3;          % Algorithm stepsize
    delta = 0.01;      % Time for derivative
    while tnow<k-l*m    % Run iteration
        i=i+1;
        % Interpolated value at tnow+tau
        xs(i) = interpolation(r_match,tnow+tau,l);
        % Get value to the right
        x_deltap = interpolation(r_match,tnow+tau+delta,l);
        % Get value to the left
        x_deltam = interpolation(r_match,tnow+tau-delta,l);
        % Calculate numerical derivative
        dx = x_deltap - x_deltam;
        % Quantize xs to nearest 4-PAM symbol
        qx = quantalph(xs(i),[-3, -1, 1, 3]);
        % Algorithm update (Decision Directed)
        tau = tau + mu*dx*(qx-xs(i));
        % Save for plotting
        tnow = tnow+m; tausave(i) = tau;
    end

    % Optimize the output...
    i=0; tau=tausave(end);
    while tnow<k-l*m    % Run iteration
        i=i+1;
        % Interpolated value at tnow+tau
        xs(i) = interpolation(r_match,tnow+tau,l);
        % Update time tnow
        tnow = tnow+m;
    end

    % Disregard the first four symbols
    xs = xs(round(m):end);

    % Plotting
    figure();
    plot(tausave);

```



```

xlabel('Iterations'); ylabel('Time Offset Estimates');
title('Clock Recovery minimizing Cluster Variance','Color','r');

figure();
scatter(1:4000,xs(1:4000));
xlabel('Iterations'); ylabel('Estimated Symbol Values');
title('Constellation History before Equalization','Color','r');
end

% Clock Recovery maximizing Output Power
if(trAlgorithm == 2)
    N = round(k*Ts/T)-2*1; % Number of data points
    tnow = 1*m+1; tau=0; xs=zeros(1,N); % Initialize vectors and
    % variables
    tausave = zeros(1,N); tausave(1) = tau; i=0;
    mu = 0.3; % Algorithm stepsize
    delta = 0.009; % Time for derivative
    while tnow<k-l*m % Run iteration
        i=i+1;
        % Interpolated value at tnow+tau
        xs(i) = interpolation(r_match,tnow+tau,l);
        % Get value to the right
        x_deltap = interpolation(r_match,tnow+tau+delta,l);
        % Get value to the left
        x_deltam = interpolation(r_match,tnow+tau-delta,l);
        % Calculate numerical derivative
        dx = x_deltap - x_deltam;
        % Algorithm update (energy)
        tau = tau + mu*dx*xs(i);
        % Save for plotting
        tnow = tnow+m; tausave(i) = tau;
    end

    % Optimize the output...
    i=0; tau=tausave(end);
    while tnow<k-l*m % Run iteration
        i=i+1;
        % Interpolated value at tnow+tau
        xs(i) = interpolation(r_match,tnow+tau,l);

```

```

        % Update time tnow
        tnow = tnow+m;
    end
    % Disregard the first four symbols
    xs = xs(round(m):end);

    % Plotting
    figure();
    plot(tausave);
    xlabel('Iterations'); ylabel('Time Offset Estimates');
    title('Clock Recovery maximizing Output Power','Color','r');

    figure();
    scatter(1:4000,xs(1:4000));
    xlabel('Iterations'); ylabel('Estimated Symbol Values');
    title('Constellation History before Equalization','Color','r');
end

% Plotting for Matched Filtering (FDM user slot allotment: 204 KHz)
figure();
semilogy(ssf,abs(fftshift(fft(matchfilt,N_F))));
grid on;
xlabel('Frequency (Hz)'); ylabel('Magnitude (dB)');
title('Frequency Response of Matched Filter','Color','r');

figure();
semilogy(ssf,abs(fftshift(fft(r_match,N_F))));
grid on;
xlabel('Frequency (Hz)'); ylabel('Magnitude (dB)');
title('Input Signal after Matched Filtering','Color','r');

```

```

% Technical University of Crete, Fall 2005 - Spring 2006
% Telecommunications Division
% Digital Communication Systems
% Ioannis P. Lazaridis - Sotirios A. Vlachodimitropoulos
% SID: 2000030071 - 2000030055
% Directed by Dr. Athanasios P. Liavas

```

```

% Equalizer and Channel Estimation Function

```

```

function rq = ...

```

```

    equalizer(xs,typeOfEstimation,typeOfEqualizer,headPam,taps,l_frame)

```

```

% rq = equalizer(xs,typeOfEqualizer,headPam,taps,eqDelay,l_frame)

```

```

%     'xs' is the output of the Timing Recovery stage

```

```

%     (The input at the equalizer)

```

```

%     'typeOfEstimation' is the parameter which

```

```

%     correponds to channel estimation method

```

```

%     'typeOfEqualizer' is the choice of the Equalizer we shall use

```

```

%     'headPam' is the header/training sequence that starts each frame

```

```

%     'taps' is the number of Equalizer taps (Equalizer coefficients)

```

```

%     'l_frame' is the frame marker sequence period (in symbols)

```

```

% Correlation can locate the header with the data

```

```

% Header is a predefined string (head_pam)

```

```

% Do correlation...

```

```

for i=0:30

```

```

    corr(i+1)=headPam*xs(1+(50*l_frame)+i:length(headPam)+(50*l_frame)+i)';

```

```

    corr(i+1)=corr(i+1)/(length(headPam)*5);

```

```

end

```

```

% Using correlation, find a window (5 taps)

```

```

% with maximum energy

```

```

for i=1:26

```

```

    energy(i) = sum(corr(i:i+4).^2);

```

```

end

```

```

[zz,delay] = max(energy);

```

```

% Location of largest energy,

```

```

% gives the channel delay

```

```

xs = xs(delay+1:end); % Disregard the first symbols,
% according to specific channel
% delay

% Channel Estimation
if(typeOfEstimation == 1) % Using Cross-Correlation
    h = corr(delay:delay+4)'; % Estimated channel impulse response
else
    for i=1:241 % Using LS-solution
        A(i,:) = headPam(4+i:-1:i);
    end
    % Estimated channel impulse response
    h = inv(A'*A)*A'*xs(4:length(headPam)-1)';
end

% Plotting
figure();
stem(energy);
xlabel('Index');ylabel('Energy Over a Window Of 5 Samples');
title('Maximum Energy Corresponds To The Start Of Frame','Color','r');

figure();
stem(h);
xlabel('Taps');ylabel('Impulse Response');
title('Estimated Channel','Color','r');

% Case of LS-Equalizer
if(typeOfEqualizer == 1)
    p = length(headPam)-delay; % Data windows
    R = toeplitz(xs(taps:p),xs(taps:-1:1)); % Build matrix R
    S = headPam(taps-delay:p-delay)'; % and vector S
    f = inv(R'*R)*R'*S; % Calculate equalizer taps
    y = filter(f,1,xs); % Equalizer is a filter
    y = y(delay+1:end); % Synchronize data for
    % specific channel delay

    rq = quantalph(y,[-3 -1 1 3]); % Quantize to 4-PAM
    % alphabet
end

```

```

comb = conv(f,h);                                % Combined Impulse
                                                % Response
                                                % Channel with
                                                % LS-Equalizer

% Plotting
figure();
stem(f);
xlabel('Taps'); ylabel('Value');
title('Taps of LS-Equalizer','color','r');

figure();
stem(comb);
xlabel('Index'); ylabel('Impulse Response');
title('Combined Impulse Response Channel -
      LS Equalizer','color','r');

figure();
stem(corr);
xlabel('Index');
title('Correlation of Header with Data','Color','r');

figure();
scatter(1:4000,y(1:4000));
xlabel('Iterations'); ylabel('Estimated Symbol Values');
title('Optimal LS-Equalizer Output','Color','r');
end

% Case of LMS-Equalizer
if(typeOfEqualizer == 2)
    f=zeros(taps,length(headPam)-taps-1);    % Initialize equalizer at 0
    mu=.01;                                  % Algorithm stepsize
    j=1;
    for i=1+taps:length(headPam)              % Iterate
        rr = xs(i:-1:i-taps+1)';              % Window of Data
        error = headPam(i-delay)-f(:,j)'*rr;% Calculate error
        f(:,j+1) = f(:,j) + mu*error*rr;      % Update equalizer
                                                % coefficients

        j = j+1;
    end
end

```

```

y = filter(f(:,end),1,xs);           % Equalizer is a filter
y = y(delay+1:end);                 % Synchronize data
                                     % for specific channel
                                     % delay

rq = quantalph(y,[-3,-1,1,3]);       % Quantize to 4-PAM
                                     % alphabet

comb = conv(f(:,end),h);             % Combined Impulse Response
                                     % Channel with
                                     % LMS -Equalizer

% Plotting
figure();
xlabel('Iterations'); ylabel('Value');
title('Taps of LMS-Equalizer in time','color','r');
hold on;
for i=1:taps
    grid on;
    plot(f(i,:));
end

figure();
stem(comb);
xlabel('Index'); ylabel('Impulse Response');
title('Combined Impulse Response Channel -
      LMS Equalizer','color','r');

figure();
stem(corr);
xlabel('Index');
title('Correlation of Header with Data','Color','r');

figure();
scatter(1:4000,y(1:4000));
xlabel('Iterations'); ylabel('Estimated Symbol Values');
title('Optimal LMS-Equalizer Output','Color','r');
end

```

```

% Case of Viterbi-Equalizer
if(typeOfEqualizer == 3)
    pamSymbol = [-3 -1 1 3];          % Used known alphabet for 4-PAM
                                      % symbols

    xs = xs(1:2000);

    N = length(xs);                  % Length of data (input)
    M = length(pamSymbol);           % Length of alphabet
    L = length(h)-1;                 % L is the length of channel - 1

    input=[xs' ; (ones(L,1)*-1)]; % Pad input signal with L -1 symbols

    % Some initial conditions...
    cost = zeros(N+L+1, M^L);
    cost(:, :) = 10^10;
    cost(1,1) = 0;

    path = zeros(N+L+1, M^L);
    path(1, :) = -3;

    for n=1:N+L                      % Do for all time indices n
        for ii=1:M^L                 % Update all costs and surviving sequences
            for j=1:4                % For previous states
                realState(j, :) = [state2sym(ii-1) pamSymbol(j)];
                localCost(j) = (input(n) - sum(h'.*realState(j, :)))^2;
                stSym = state2sym(ii-1);
                oldState(j, :) = ...
                    bit2int(sym2bit([stSym(2:L) pamSymbol(j)]))+1;
                symCost(j) = cost(n, oldState(j, :)) + localCost(j);
            end
            % Choose lower cost
            [minCost, ind] = min(symCost);
            % Update cost and path on trellis diagram
            cost(n+1, ii) = minCost;
            path(n+1, ii) = pamSymbol(ind);
        end
    end

    currentState=1;

```

```
% The path on trellis diagram with the lower total
% cost gives the estimation of the input at the channel
for n = N+L:-1:1
    % A buffer with 4 symbols where found at specific state
    currentSymbols = state2sym(currentState-1);

    % Estimate the input sequence
    if n<=N
        rq(n) = currentSymbols(1);
    end

    % Update state to find the correct symbol at the input
    currentState = bit2int(sym2bit(( ...
        [currentSymbols(2:L) path(n+1, currentState) ] )))+1;
end
end
```



```

% Technical University of Crete, Fall 2005 - Spring 2006
% Telecommunications Division
% Digital Communication Systems
% Ioannis P. Lazaridis - Sotirios A. Vlachodimitropoulos
% SID: 2000030071 - 2000030055
% Directed by Dr. Athanasios P. Liavas

% Decoding Function
function ztext = decoding(rq,l_frame,headPam)

% ztext = decoding(rq,l_frame,headPam)
%      'rq' is the output of the Equalization stage
%      (The input for decoding)
%      'l_frame' is the frame marker sequence period (in symbols)
%      'headPam' is the header/training sequence
%      that starts each frame

j=1;
for i=1:l_frame:length(rq)-l_frame
    r_hat(j:(l_frame-length(headPam)-1+j))=...
        rq(i+length(headPam):i+l_frame-1);
    j=j+l_frame-length(headPam);
end

% (5,2) Binary Linear Block Code Part 1: Definitions
% The generator and parity check matrices
g = [1 0 1 0 1;
     0 1 0 1 1];

h = [1 0 1 0 0;
     0 1 0 1 0;
     1 1 0 0 1];

% "Inverse" ginv such that cw*ginv=x (mod 2)
ginv = [1 1; 1 0 ; 0 0; 1 0; 0 1];

```

```

% The syndrome table
syn = [0 0 0 0 0;
       0 0 0 0 1;
       0 0 0 1 0;
       0 1 0 0 0;
       0 0 1 0 0;
       1 0 0 0 0;
       1 1 0 0 0;
       1 0 0 1 0];

% (5,2) Binary Linear Block Code Part 2: Decoding data
k=1;
for i=1:round(length(rq)/10)           % Translate back into 0-1 binary
    if r_hat(i) == +3, yy(k:k+1) = [1,1]; end
    if r_hat(i) == +1, yy(k:k+1) = [1,0]; end
    if r_hat(i) == -1, yy(k:k+1) = [0,1]; end
    if r_hat(i) == -3, yy(k:k+1) = [0,0]; end
    k = k+2;
end

% Initialize variables
k=1; z = zeros(1,floor((2/5)*length(yy)));
for i=1:5:length(yy)-4
    % Decode binary using
    % (5,2) block code
    eh = mod(yy(i:i+4)*h',2);           % Multiply by parity check h'
    ehind = eh(1)*4+eh(2)*2+eh(3)+1;    % Turn syndrome into index
    e = syn(ehind,:);                   % Error from syndrome table
    yy(i:i+4) = mod(yy(i:i+4)+e,2);     % Add e to correct errors
    z(k:k+1) = mod(yy(i:i+4)*ginv,2);   % Decode corrected codeword
    k = k+2;
end
% Translate from 0-1 binary to text to receive the secret signal
ztext=bin2text(z);

```

```

% Technical University of Crete, Fall 2005 - Spring 2006
% Telecommunications Division
% Digital Communication Systems
% Ioannis P. Lazaridis - Sotirios A. Vlachodimitropoulos
% SID: 2000030071 - 2000030055
% Directed by Dr. Athanasios P. Liavas

% Viterbi Algorithm for ML sequence estimation
% Data are considered -3, -1, +1, +3
% Channel has length L+1, h(1),...,h(L+1)
close all; clear all; clc;

N = 1000;                                % Number of samples to process

h = randn(5,1);                          % Random channel with 5 elements
h = h/norm(h);                          % Normalized channel
s = randn(N,1);                          % Random process for input
r = randn(N,1);                          % Random process for noise

x = quantalph(s,[-3 -1 1 3]);            % 4-PAM input
                                         % (Quantize to 4-PAM alphabet)

pamSymbol = [-3 -1 1 3];                 % 4-PAM alphabet we shall use
M = length(pamSymbol);                   % Length of alphabet
L = length(h)-1;                         % Length of channel -1

k=1;
for SNR = 5:5:20;                         % For different SNR's in dB
    xpower = 5*sum(abs(h).^2);            % Power of signal at
                                         % the output of the channel
    npower = xpower/(10^(SNR/10));        % Noise variance for different SNR's

    noise = sqrt(npower)*r;              % Additional noise

    y = conv(x,h);                        % Convolve data with channel
    y = y(1:N);                          % Disregard the tail of convolution
    xs = y + noise;                      % Output of the channel

```

```

input=[xs ; (ones(L,1).*(-1))]; % Pad input signal
                                   % with L -1 symbols

% Some initial conditions...
% Initialize cost
cost = zeros(N+L+1, M^L);
cost(1,:) = inf;
cost(1,1) = 0;

% Initialize path on trellis diagram
path = zeros(N+L+1, M^L);
path(1,:) = -3;

for n=1:N+L % Do for all time indices n
    for ii=1:M^L % Update all costs and surviving sequences
        for j=1:4 % For previous states
            realState(j,:) = [state2sym(ii-1) pamSymbol(j)];
            localCost(j) = (input(n) - sum(h'.*realState(j,:)))^2;
            stSym = state2sym(ii-1);
            oldState(j,:) = ...
                bit2int(sym2bit([stSym(2:L) pamSymbol(j)]))+1;
            symCost(j) = cost(n,oldState(j,:)) + localCost(j);
        end
        % Choose lower cost
        [minCost,ind] = min(symCost);
        % Update cost and path on trellis diagram
        cost(n+1,ii) = minCost;
        path(n+1,ii) = pamSymbol(ind);
    end
end

currentState=1;

```

```

% The path on trellis diagram with the lower total
% cost gives the estimation of the input at the channel
for n = N+L:-1:1
    % A buffer with 4 symbols where found at specific state
    currentSymbols = state2sym(currentState-1);

    % Estimate the input sequence
    if n<=N
        out(n) = currentSymbols(1);
    end

    % Update state to find the correct symbol at the input
    currentState = bit2int(sym2bit(...
        [currentSymbols(2:L) path(n+1, currentState) ] ))+1;
end

% Calculate errors for each SNR
errors(k) = (sum(out' ~= x))/N;
k=k+1;
end

% Plotting BER
semilogy(5:5:20,errors,'-s');
grid on;
xlabel('SNR in dB');
ylabel('Bit Error Rate');
title('BER vs SNR','Color','r');

```

```
% Technical University of Crete, Fall 2005 - Spring 2006
% Telecommunications Division
% Digital Communication Systems
% Ioannis P. Lazaridis - Sotirios A. Vlachodimitropoulos
% SID: 2000030071 - 2000030055
% Directed by Dr. Athanasios P. Liavas

% C. R. Johnson, Jr - William A Sethares

% Quantizer Function (Quantize to nearest 4-PAM symbol)
function y = quantalph(x,alphabet)

% y = quantalph(x,alphabet)
%   Quantize the input signal x to the alphabet
%   using nearest neighbor method
%   'x' is the input vector to be quantized
%   'alphabet' is the vector of discrete values that y can take on
%           sorted in ascending order
%   'y' is the output - the quantized vector

[r c] = size(alphabet);
if c>r
    alphabet = alphabet';
end

[r c] = size(x);
if c>r
    x = x';
end

alpha = alphabet(:,ones(size(x)))';
dist = (x(:,ones(size(alphabet))) - alpha).^2;
[v,i] = min(dist,[],2);
y = alphabet(i);
```

```
% Technical University of Crete, Fall 2005 - Spring 2006
% Telecommunications Division
% Digital Communication Systems
% Ioannis P. Lazaridis - Sotirios A. Vlachodimitropoulos
% SID: 2000030071 - 2000030055
% Directed by Dr. Athanasios P. Liavas
```

```
% C. R. Johnson, Jr - William A Sethares
```

```
% Square Root Raised Cosine Pulse Function
function s = srrc(syms, beta, P, t_off);
```

```
% s = srrc(syms, beta, P, t_off);
%   Generate a Square-Root Raised Cosine Pulse
%   'syms' is the one sided length of srrc pulse in symbol durations
%   'beta' is the rolloff factor: beta=0 gives the sinc function
%   'P' is the oversampling factor
%   't_off' is the phase (or timing) offset
```

```
% If unspecified, offset is 0
if nargin==3, t_off=0; end;
```

```
% Sampling indices as a multiple of T/P
k = -syms*P + 1e-8 + t_off:syms*P + 1e-8 + t_off;
```

```
% Numerical problems if beta=0
if (beta==0), beta = 1e-8; end;
```

```
% Calculation of srrc pulse
s = 4*beta/sqrt(P)*(cos((1+beta)*pi*k/P)+...
    sin((1-beta)*pi*k/P)./(4*beta*k/P))./...
    (pi*(1-16*(beta*k/P).^2));
```

```
% Technical University of Crete, Fall 2005 - Spring 2006
% Telecommunications Division
% Digital Communication Systems
% Ioannis P. Lazaridis - Sotirios A. Vlachodimitropoulos
% SID: 2000030071 - 2000030055
% Directed by Dr. Athanasios P. Liavas
```

```
% Interpolator Function
```

```
function y = interpolation(x, t, l, beta)
```

```
% y = interpolation(x, t, l, beta)
```

```
% 'x' is the sampled data
```

```
% 't' is the place at which value desired
```

```
% 'l' is the one sided length of data to interpolate
```

```
% 'beta' is the rolloff factor for SRRC function
```

```
% if beta = 0 is a sinc
```

```
if nargin == 3, beta = 0; end; % If unspecified, beta is 0
tnow = round(t); % Create indices tnow = integer part
tau = t-round(t); % plus tau = fractional part
s_tau = srrc(l,beta,1,tau); % Interpolating sinc at offset tau
for i=1:2*l+1
    x_tau(i) = x(tnow-i)*s_tau(i); % Interpolate the signal
end
y = sum(x_tau); % y is the new sample
```



```
% Technical University of Crete, Fall 2005 - Spring 2006
% Telecommunications Division
% Digital Communication Systems
% Ioannis P. Lazaridis - Sotirios A. Vlachodimitropoulos
% SID: 2000030071 - 2000030055
% Directed by Dr. Athanasios P. Liavas

% C. R. Johnson, Jr - William A Sethares

% Function which converts 7-bit representation to a text string
function ztext = bin2text(z)

rp = floor(length(z)/7); rez='';
for i=1:7*rp
    rez(i) = num2str(z(i));
end
recon = reshape(rez,7,rp)';
ztext = char(bin2dec(recon))';
```

```
% Technical University of Crete, Fall 2005 - Spring 2006
% Telecommunications Division
% Digital Communication Systems
% Ioannis P. Lazaridis - Sotirios A. Vlachodimitropoulos
% SID: 2000030071 - 2000030055
% Directed by Dr. Athanasios P. Liavas
```

```
% Converts each 8 consecutive bits (MSbit @ bottom, LSbit @ top)
% of the bs
% column vector of bits (1/0), to the corresponding integer in
% {0,...,255}
function intrep = bit2int(bs)

intrep = [2^7 2^6 2^5 2^4 2^3 2^2 2 1]*bs;
```

```
% Technical University of Crete, Fall 2005 - Spring 2006
% Telecommunications Division
% Digital Communication Systems
% Ioannis P. Lazaridis - Sotirios A. Vlachodimitropoulos
% SID: 2000030071 - 2000030055
% Directed by Dr. Athanasios P. Liavas
```

```
% Function which converts an integer to a 8-bit string
function bitstring = int2bitstring(n)
```

```
% bitstring = int2bitstring(n)
%      'n' is the input to convert
%      'bitstring' is the output of the function
%      The column of this matrix will store the 8-bit binary
%      representation of the integers contained in vector n.
%      Most Significant bit is at the top of each
%      column and Least Significant bit is at the bottom
```

```
bitstring = zeros(8,1);
for k=8:-1:1
    bit = mod(n,2);
    bitstring(k,:) = bit';
    n = n-bit;
    n = n/2;
end
```

```
% Technical University of Crete, Fall 2005 - Spring 2006
% Telecommunications Division
% Digital Communication Systems
% Ioannis P. Lazaridis - Sotirios A. Vlachodimitropoulos
% SID: 2000030071 - 2000030055
% Directed by Dr. Athanasios P. Liavas
```

```
% Function which translates a decimal state (0 to 255)
% to 4 PAM symbols, using known alphabet [-3 -1 1 3]
function symbols = state2sym(state)
```

```
binState = int2bitstring(state); % Converts decimal state to binary
j=1;
for pointer=1:2:8
    if binState(pointer)==0 && binState(pointer+1)==0
        symbols(1,j) = -3;
    elseif binState(pointer)==0 && binState(pointer+1)==1
        symbols(1,j) = -1;
    elseif binState(pointer)==1 && binState(pointer+1)==0
        symbols(1,j) = 1;
    else
        symbols(1,j) = 3;
    end
    j=j+1;
end
```

```
% Technical University of Crete, Fall 2005 - Spring 2006
% Telecommunications Division
% Digital Communication Systems
% Ioannis P. Lazaridis - Sotirios A. Vlachodimitropoulos
% SID: 2000030071 - 2000030055
% Directed by Dr. Athanasios P. Liavas
```

```
% Function which translates 4-PAM symbols to a binary
% representation, using known alphabet [-3 -1 1 3]
function bitstring = sym2bit(symbols)
```

```
j=1;
bitstring = zeros(8,1);
for i=1:min(length(symbols),4)
    if(symbols(i)==-3)
        bitstring(j:j+1)=[0 ; 0];
    elseif(symbols(i)==-1)
        bitstring(j:j+1)=[0 ; 1];
    elseif(symbols(i)==1)
        bitstring(j:j+1)=[1 ; 0];
    else
        bitstring(j:j+1)=[1 ; 1];
    end
    j=j+2;
end
```

# Βιβλιογραφία

- [1] Johnson, C. Richard, Jr. & Sethares, William A. [2004], *Telecommunication Breakdown: “Concepts of Communication Transmitted via Software-Defined Radio”*
- [2] Proakis, John G. & Salehi, Masoud. [2002], *Communication Systems Engineering*
- [3] Widrow, Bernard & Stearns, Samuel D. [1985], *Adaptive signal processing*
- [4] Lee, Edward A. & Messerschmitt, David G. [1994], *Digital communication*
- [5] Vaseghi, Saeed V. [2000], *Advanced digital signal processing and noise reduction*
- [6] Tretter, Steven A. [2003], *Communication system design using DSP algorithms*
- [7] Chassaing, Rulph. [2005], *Digital signal processing and applications with the C67*
- [8] Σιδηρόπουλος, Νικόλαος. Πανεπιστημιακές Σημειώσεις Μαθήματος “Στατιστική Επεξεργασία Σήματος για τις Τηλεπικοινωνίες”
- [9] Σιδηρόπουλος, Νικόλαος. Πανεπιστημιακές Σημειώσεις Μαθήματος “Ψηφιακά Τηλεπικοινωνιακά Συστήματα Ι”
- [10] Λιάβας, Αθανάσιος. Πανεπιστημιακές Σημειώσεις Μαθήματος “Ψηφιακά Τηλεπικοινωνιακά Συστήματα ΙΙ”
- [11] Λιάβας, Αθανάσιος. Πανεπιστημιακές Σημειώσεις Μαθήματος “Σήματα & Συστήματα”