

Εκτίμηση Φάσης σε Δορυφορικές Επικοινωνίες

Ιωάννης Α. Παπαδόπουλος
Τμήμα Μηχανικών Η/Υ, Τηλεπικοινωνιών και Δικτύων
Πανεπιστήμιο Θεσσαλίας

Επιβλέπων καθηγητής: Γεώργιος Μουστακίδης
Επιτροπή αξιολόγησης: Γεώργιος Ευθυβουλίδης
Ιορδάνης Κουτσόπουλος

1η Οκτωβρίου 2005



ΠΑΝΕΠΙΣΤΗΜΙΟ ΘΕΣΣΑΛΙΑΣ
ΥΠΗΡΕΣΙΑ ΒΙΒΛΙΟΘΗΚΗΣ & ΠΛΗΡΟΦΟΡΗΣΗΣ
ΕΙΔΙΚΗ ΣΥΛΛΟΓΗ «ΓΚΡΙΖΑ ΒΙΒΛΙΟΓΡΑΦΙΑ»

Αριθ. Εισ.: 3463/1
Ημερ. Εισ.: 10-05-2006
Δωρεά: Συγγραφέα
Ταξιθετικός Κωδικός: ΠΤ- ΜΗΥΤΔ
2005
ΠΑΠ

Περίληψη

Στην παρούσα εργασία παρουσιάζονται σύντομα τα κυριότερα χαρακτηριστικά των δορυφορικών συστημάτων και τα προβλήματα που αντιμετωπίζονται κατά τη διάρκεια της μεταφοράς δεδομένων. Μελετάται η επίδραση του θορύβου του καναλιού και του θορύβου φάσης σε σύμβολα QPSK, την πιο συνηθισμένη διαμόρφωση στα σύγχρονα δορυφορικά συστήματα.

Παράλληλα, γίνεται μια προσέγγιση στην οικογένεια των εκτιμητών NDA Viterbi & Viterbi και βαρυκέντρου για την εκτίμηση του θορύβου φάσης. Παρουσιάζεται μια βέλτιστη μορφή της πρώτης και μία υλοποίηση με πολλαπλά περάσματα.

Η εργασία ολοκληρώνεται με την εκτέλεση προσομοιώσεων για την απόδοση των εκτιμητών σε διαφορετικές συνθήκες και σύγκριση μεταξύ των.

Περιεχόμενα

1	ΕΙΣΑΓΩΓΗ	1
1.1	Εισαγωγή στις δορυφορικές επικοινωνίες	1
1.1.1	On-board processing	2
1.1.2	Το μέλλον	2
1.2	Χαρακτηριστικά και προδιαγραφές λειτουργίας	2
1.2.1	Τροχιές	3
1.2.2	Περιοχές συχνοτήτων	4
1.2.3	Διαμόρφωση	4
1.2.4	Μοντέλο πρόσβασης	4
1.2.5	Καθολικός συγχρονισμός του συστήματος	5
1.3	Ατέλειες κατά την διαμόρφωση και την αποδιαμόρφωση	6
1.4	Αρχιτεκτονική αποδιαμορφωτή	7
1.4.1	Λειτουργίες	7
1.4.2	Συγχρονισμός	7
1.5	Υποθέσεις εργασίας	9
1.6	Οργάνωση του κειμένου	10
2	QPSK	11
2.1	Εισαγωγή	11
2.1.1	Διαμορφωτής QPSK	12
2.1.2	Αποδιαμορφωτής QPSK	12
2.2	Ιδανική περίπτωση	14
2.2.1	Ιδανικό σύμβολο	14
2.2.2	Ιδανικό σήμα	14
2.2.3	Συναρτήσεις ορθοκανονικής βάσης	14

2.2.4	Ορισμός διανύσματος παρατήρησης	15
2.2.5	Διάνυσμα παρατήρησης ιδανικού σήματος	16
2.3	Περίπτωση με AWGN θόρυβο	16
2.3.1	Σήμα με AWGN θόρυβο	16
2.3.2	Διάνυσμα παρατήρησης σήματος με AWGN θόρυβο	17
2.4	Περίπτωση με AWGN θόρυβο και θόρυβο φάσης στον πομπό	18
2.4.1	Θόρυβος φάσης	18
2.4.2	Σήμα με θόρυβο φάσης	20
2.4.3	Λαμβανόμενο σήμα	20
2.4.4	Διάνυσμα παρατήρησης σήματος με θόρυβο φάσης στον πομπό, AWGN θόρυβο και απώλειες	20
2.5	Περίπτωση με AWGN θόρυβο και θόρυβο φάσης σε πομπό και δέκτη	23
2.5.1	Συναρτήσεις ορθοκανονικής βάσης με θόρυβο φάσης	23
2.5.2	Διάνυσμα παρατήρησης σήματος με AWGN θόρυβο και θόρυβο φάσης σε πομπό και δέκτη	24
3	ΕΚΤΙΜΗΤΗΣ VITERBI & VITERBI	29
3.1	Μη γραμμικός εκτιμητής V&V	29
3.2	Βήματα του αλγορίθμου του εκτιμητή V&V	33
3.3	Μοντελοποίηση του προβλήματος	33
3.4	Εναλλακτική μορφή του εκτιμητή V&V	34
3.5	Αμεροληψία εκτιμητή	34
3.5.1	Εκτιμητής φάσης για BPSK και $F(\bullet) = (\bullet)^2$	34
3.5.2	Εκτιμητής φάσης για QPSK και $F(\bullet) = (\bullet)^4$	36
3.6	Βέλτιστος εκτιμητής V&V	38
3.6.1	Βέλτιστος εκτιμητής V&V για $\Delta f = 0$	39
3.6.2	Βέλτιστος εκτιμητής V&V για $\Delta f \neq 0$	40
3.7	Εκτιμητής V&V πολλαπλών περασμάτων	40
4	ΕΚΤΙΜΗΤΕΣ ΒΑΡΥΚΕΝΤΡΟΥ	45
4.1	Εισαγωγή στους εκτιμητές βαρυκέντρου	45
4.2	Εύρεση μέσου όρου υπολοίπου φάσης από πλήθος n συμβόλων	45
4.3	Αφαίρεση διαμόρφωσης – συνάρτηση <i>FOLD</i>	46
4.4	Συναρτήσεις εύρεσης μέσου όρου υπολοίπου φάσης	47

4.4.1	Η αρχική συνάρτηση βαρυκέντρου	47
4.4.2	Άλλες συναρτήσεις βαρυκέντρου	48
4.5	Κυριότερες διαφορές εκτιμητών βαρυκέντρου και V&V	49
5	ΠΡΟΣΟΜΟΙΩΣΕΙΣ	51
5.1	Σχέση μεταξύ SNR και $\frac{E_b}{N_0}$ για QPSK	51
5.2	Παράμετροι προσομοίωσης	52
5.3	Προσομοιώσεις των εκτιμητών V&V	53
5.3.1	Μελέτη εκτιμητή V&V για διαφορετικό $F(\bullet)$	53
5.3.2	Μελέτη βέλτιστου εκτιμητή V&V και εκτιμητή V&V πολλαπλών περασμάτων	62
5.4	Προσομοιώσεις των εκτιμητών βαρυκέντρου	76
5.4.1	Μελέτη εκτιμητών βαρυκέντρου σε σχέση με τον V&V	76
5.4.2	Σύγκριση εκτιμητών LBAR και QBAR	79
A'	ΧΡΗΣΙΜΕΣ ΣΧΕΣΕΙΣ	83
A'.1	Σχέση ενέργειας συμβόλου με ενέργεια bit	83
A'.2	Χρήσιμες τριγωνομετρικές σχέσεις	83
A'.3	Χρήσιμες μιγαδικές σχέσεις	84
B'	ΚΩΔΙΚΑΣ MatLab	85
B'.1	Κώδικας Κεφαλαίου 2	85
B'.1.1	Κώδικας δημιουργίας του Σχήματος(2.5)	85
B'.2	Κώδικας Κεφαλαίου 4	85
B'.2.1	Κώδικας δημιουργίας του Σχήματος(4.2)	85
B'.2.2	Κώδικας δημιουργίας του Σχήματος(4.3)	86
B'.3	Συναρτήσεις προσομοιώσεων	87
B'.3.1	Κώδικας μετατροπής ακτινίων σε μοίρες	87
B'.3.2	Κώδικας εύρεσης φάσης συμβόλου στο διάστημα $[-\frac{\pi}{4}, \frac{3\pi}{2} + \frac{\pi}{4}]$	87
B'.3.3	Κώδικας ανίχνευσης συμβόλων	87
B'.3.4	Κώδικας εκτιμητή V&V	88
B'.3.5	Κώδικας βέλτιστου εκτιμητή V&V	88
B'.3.6	Κώδικας συνάρτησης αφαίρεσης διαμόρφωσης <i>FOLD</i>	89
B'.3.7	Κώδικας συνάρτησης βαρυκέντρου <i>BAR</i>	90
B'.3.8	Κώδικας συνάρτησης βαρυκέντρου <i>EBAR</i>	90

B'.3.9	Κώδικας συνάρτησης βαρυκέντρου <i>LBAR</i>	91
B'.3.10	Κώδικας συνάρτησης βαρυκέντρου <i>QLBAR</i>	91
B'.3.11	Κώδικας εκτιμητή <i>BAR</i>	92
B'.3.12	Κώδικας εκτιμητή <i>EBAR</i>	92
B'.3.13	Κώδικας εκτιμητή <i>LBAR</i>	93
B'.3.14	Κώδικας εκτιμητή <i>QLBAR</i>	93
B'.4	Κώδικας προσομοιώσεων εκτιμητών <i>V&V</i>	94
B'.4.1	Κώδικας Ενότητας 5.3.1	94
B'.4.2	Κώδικας Ενότητας 5.3.2	107
B'.5	Κώδικας προσομοιώσεων εκτιμητών βαρυκέντρου	123

Κατάλογος Σχημάτων

1.1	Τυπικό δορυφορικό σύστημα	3
1.2	bit synchronized σύστημα vs burst synchronized	6
1.3	Δομή αποδιαμορφωτή	7
1.4	Αλγόριθμοι feedback και feed-forward	8
1.5	Αλγόριθμοι Decision Directed και Non Data Aided	8
2.1	QPSK constellation diagram	11
2.2	Διαμορφωτής QPSK	12
2.3	Αποδιαμορφωτής QPSK	13
2.4	Διάταξη απόφασης	13
2.5	Επίπτωση του θορύβου φάσης στα σύμβολα QPSK – δεν εμφανίζεται η επίπτωση τυχόν AWGN	18
3.1	Μη γραμμικός εκτιμητής V&V	30
4.1	Στιγμιότυπο εύρεσης μέσου όρου 8 τιμών	46
4.2	Αποτέλεσμα συνάρτησης BAR	48
4.3	Συναρτήσεις αξιοπιστίας $LBAR$ και $QLBAR$	50
5.1	Προσομοίωση εκτιμητή V&V σε επίπεδο συμβόλων με $SNR = 10$ dB	53
5.2	Προσομοίωση εκτιμητή V&V σε επίπεδο συμβόλων με $SNR = 4$ dB	54
5.3	Προσομοίωση εκτιμητή V&V σε επίπεδο συμβόλων με $SNR = 0$ dB	54
5.4	Προσομοίωση εκτιμητή V&V σε επίπεδο συμβόλων με $SNR = 0$ dB και μικρό Δf	54
5.5	Προσομοίωση εκτιμητή V&V για μεταβλητό θόρυβο φάσης (Υπόλοιπο διαφοράς φάσης)	56
5.6	Προσομοίωση εκτιμητή V&V για μεταβλητό θόρυβο φάσης (Ποσοστό εσφαλμένων συμβόλων)	57
5.7	Προσομοίωση εκτιμητή V&V για διαφορετικά SNR	58

5.8	Προσομοίωση εκτιμητή V&V για μεταβλητό μήκος παραθύρου	59
5.9	Προσομοίωση εκτιμητή V&V σε συνθήκες υψηλού θορύβου για μεγάλα παράθυρα	60
5.10	Προσομοίωση εκτιμητή V&V, βέλτιστου και πολλαπλών περασμάτων σε επίπεδο συμβόλων με $SNR = 10$ dB	63
5.11	Προσομοίωση εκτιμητή V&V, βέλτιστου και πολλαπλών περασμάτων σε επίπεδο συμβόλων με $SNR = 4$ dB	63
5.12	Προσομοίωση εκτιμητή V&V, βέλτιστου και πολλαπλών περασμάτων σε επίπεδο συμβόλων με $SNR = 0$ dB	64
5.13	Προσομοίωση εκτιμητή V&V, βέλτιστου και πολλαπλών περασμάτων σε επίπεδο συμβόλων με $SNR = 0$ dB και μικρό Δf	64
5.14	Προσομοίωση εκτιμητή V&V, βέλτιστου V&V και V&V πολλαπλών περασμάτων για μεταβλητό θόρυβο φάσης (Υπόλοιπο διαφοράς φάσης)	66
5.15	Προσομοίωση εκτιμητή V&V, βέλτιστου V&V και V&V πολλαπλών περασμάτων για μεταβλητό θόρυβο φάσης (Ποσοστό εσφαλμένων σύμβολων)	67
5.16	Προσομοίωση εκτιμητή V&V, βέλτιστου V&V και V&V πολλαπλών περασμάτων για διαφορετικά SNR	68
5.17	Προσομοίωση εκτιμητή V&V, βέλτιστου V&V και V&V πολλαπλών περασμάτων για μήκος παραθύρου 64 σύμβολα	69
5.18	Προσομοίωση εκτιμητή V&V, βέλτιστου V&V και V&V πολλαπλών περασμάτων για μήκος παραθύρου 128 σύμβολα	69
5.19	Προσομοίωση εκτιμητή V&V, βέλτιστου V&V και V&V πολλαπλών περασμάτων για μήκος παραθύρου 256 σύμβολα	70
5.20	Προσομοίωση εκτιμητή V&V, βέλτιστου V&V και V&V πολλαπλών περασμάτων για μήκος παραθύρου 256 σύμβολα και μεγάλο Δf	70
5.21	Προσομοίωση εκτιμητή V&V πολλαπλών περασμάτων για διαφορετικό $F(\bullet)$	71
5.22	Προσομοίωση εκτιμητή V&V πολλαπλών περασμάτων για 1 και 2 περάσματα και σύγκριση με βέλτιστο και απλό V&V	72
5.23	Προσομοίωση εκτιμητή V&V πολλαπλών περασμάτων για 1 και 2 περάσματα και σύγκριση με βέλτιστο και απλό V&V με υψηλό θόρυβο και παράθυρα των 64 και 128 συμβόλων	73
5.24	Προσομοίωση εκτιμητών βαρυκέντρου για μεταβλητό θόρυβο φάσης	76
5.25	Προσομοίωση εκτιμητών βαρυκέντρου για διαφορετικά SNR	77
5.26	Προσομοίωση εκτιμητών βαρυκέντρου για μεταβλητό μήκος παραθύρου	78
5.27	Προσομοίωση εκτιμητών LBAR και QLBAR για μεταβλητό θόρυβο φάσης	80
5.28	Προσομοίωση εκτιμητών LBAR και QLBAR για διαφορετικά SNR	80

Κατάλογος Πινάκων

1.1	Σύγκριση απαιτήσεων ισχύος-εύρους ζώνης μεταξύ mPSK και BPSK. Πιθανότητα σφάλματος συμβόλων = 10^{-4}	4
2.1	QPSK σύμβολα και dibits	12
5.1	Τυπικές τιμές SNR και $\frac{E_b}{N_0}$	52
5.2	Παράμετροι προσομοίωσης	52

Κεφάλαιο 1

ΕΙΣΑΓΩΓΗ

1.1 Εισαγωγή στις δορυφορικές επικοινωνίες

Πάντα η ανάγκη για αξιόπιστη, γρήγορη και απροβλημάτιστη επικοινωνία ωθούσε τον άνθρωπο να ξεπερνάει τα φυσικά του όρια, να σχεδιάζει και να κατασκευάζει μια πλειάδα συστημάτων. Από τους αρχαίους χρόνους χρησιμοποιούσε ανθρώπους και ζώα για να μεταφέρει μηνύματα. Συνέχισε με τις πρώτες ασύρματες μεταδόσεις, χρησιμοποιώντας φωτιές και καπνό και στη σύγχρονη εποχή δημιούργησε εκτενή δίκτυα ταχυδρομείων, τηλεγράφου και τηλεφώνου.

Αυτή η ίδια ανάγκη για επικοινωνία ώθησε στην χρήση και των ραδιοκυμάτων. Όταν πλέον είχαν απλωθεί χιλιόμετρα καλωδίων, οι διάφορες κεραίες γέμισαν τη γη, μεταδίδοντας ήχο, ομιλία και εικόνα με συγκριτικά χαμηλότερο κόστος. Όμως ούτε αυτό επαρκούσε. Για κοντινές αποστάσεις, οι επίγειες επικοινωνίες ήταν πολύ καλές. Το κόστος όμως της επίγειας κάλυψης αυξάνεται δυσανάλογα πολύ όταν πρόκειται για την κάλυψη αποστάσεων χιλιάδων χιλιομέτρων. Η προφανέστερη λύση ήταν οι δορυφόροι.

Ο πρώτος δορυφόρος εκτοξεύτηκε στις 4 Οκτωβρίου 1957 και ήταν ο πασίγνωστος πλέον Sputnik I της ΕΣΣΔ. Από εκείνη την ημερομηνία-σταθμό για τις επικοινωνίες ξεκίνησε και η κούρσα του διαστήματος, εν μέσω του Ψυχρού Πολέμου. Ευρώπη (Γαλλία 1965 Astérix, Αγγλία 1971 Prospero X-3) αλλά κυρίως ΗΠΑ (1958, Explorer 1) και ΕΣΣΔ ξεκίνησαν μια μακρόχρονη κούρσα για την επικράτηση στο τελευταίο σύνορο. Οι αναλογικοί δορυφόροι γέμισαν τον ουρανό. Αντί πλέον να χρησιμοποιούνται επίγειες μόνον ζεύξεις, γινόταν χρήση ραδιοσυχνοτήτων που έστελναν τα δεδομένα στον δορυφόρο, ο οποίος απλώς τις αναμετέδιδε, ενισχύοντας τις, προς κάποιο άλλο σημείο, άλλη χώρα, άλλη ήπειρο. Όταν κυριαρχούσαν οι μικρές αποστάσεις και η αμφίδρομη επικοινωνία, οι επίγειες επικοινωνίες ήταν παρούσες. Στις μεγάλες αποστάσεις όμως οι δορυφόροι θριάμβευαν. Τα δορυφορικά δίκτυα προσέφεραν τη δυνατότητα παγκόσμιας κάλυψης σε εξαιρετικά σύντομο χρόνο και με ανταγωνιστικό κόστος.

Όμως, η πραγματική επανάσταση στις δορυφορικές επικοινωνίες συντελέστηκε με την χρήση των ψηφιακών δεδομένων σε πακέτα. Οι δορυφόροι έπαψαν να είναι απλοί αναμεταδότες και έγιναν κέντρα επεξεργασίας δεδομένων. Όλα τα πακέτα πλέον φτάνουν στον δορυφόρο, ο οποίος τα αποδιαμορφώνει, τα αποκωδικοποιεί, τα διορθώνει και τα αναμεταδίδει, συνήθως σε διαφορετική μορφή από αυτήν που του ήρθαν. Έχει αποκτήσει πια έναν ενεργητικό ρόλο στην μεταγωγή δεδομένων.

Σήμερα πολλά συστήματα και υπηρεσίες βασίζονται σε ψηφιακούς δορυφόρους. Για παράδειγμα το GPS, με τους 28 δορυφόρους του, δεκάδες δορυφόροι της δορυφορικής τηλεόρασης και του δο-

ρυφορικού ραδιοφώνου, αλλά και γενικής χρήσης δορυφόροι – τηλεοπτικό σήμα, δεδομένα, Internet – όπως οι HellasSat 1 και HellasSat 2, βρίσκονται σε τροχιά γύρω από την γη.

1.1.1 On-board processing

Οι σύγχρονες ανακαλύψεις στους τομείς της αναλογικής και ψηφιακής σχεδίασης επέτρεψαν στους σχεδιαστές δορυφορικών συστημάτων να αυξήσουν τα κυκλώματα που χρησιμοποιούν στους δορυφόρους. Οι κατασκευαστές έχοντας πλέον διαθέσιμα σε κάθε δορυφόρο συστήματα αποδιαμόρφωσης, διόρθωσης λαθών, αποκοπής θορύβου και φιλτραρίσματος, μπορούν να μειώσουν την πολυπλοκότητα των επίγειων σταθμών που αποστέλλουν τα δεδομένα στους δορυφόρους και ταυτόχρονα να αυξήσουν την ταχύτητα διαμεταγωγής. Οι δορυφόροι είναι ενεργά μέλη των δικτύων που επεξεργάζονται και διορθώνουν το σήμα και όχι απλοί αναμεταδότες. Αυτό επιτρέπει ακόμα μεγαλύτερη ευελιξία, ιδίως τώρα που τα δορυφορικά συστήματα είναι ζωτικής σημασίας για τις σύγχρονες πολυμεσικές εφαρμογές.

Με τον όρο *on-board processing* (OBP) περιγράφεται αυτή ακριβώς αυτή η ενσωμάτωση ενός αριθμού κυκλωμάτων που αναλαμβάνουν αποδιαμόρφωση, ανάκτηση, μεταγωγή πακέτων, αποκωδικοποίηση και διαμόρφωση πάνω στον δορυφόρο. Έτσι, οι κατασκευαστές πέτυχαν

- να μειώσουν τη συνολική επίδραση του θορύβου, αφού πλέον οι δορυφόροι δεν ήταν απλοί ενισχυτές που ενίσχυαν και τον θόρυβο,
- να χρησιμοποιούν διαφορετικά συστήματα κωδικοποίησης και πρόσβασης στην ζεύξη Γης-δορυφόρου και δορυφόρου-Γης και
- να μειώσουν την πολυπλοκότητα των συστημάτων στη Γη, αφού μέρος της επεξεργασίας γινόταν πλέον στο δορυφόρο.

1.1.2 Το μέλλον

Το τι θα επακολουθήσει είναι ακόμα άγνωστο. Πάντως βραχυπρόθεσμα θα πρέπει να περιμένουμε ακόμα μεγαλύτερη χρήση της επεξεργασίας δεδομένων πάνω στον ίδιο το δορυφόρο. Η ανάγκη για ακόμα μεγαλύτερη διαμεταγωγή δεδομένων αλλά και η εμφάνιση νέων εφαρμογών και η αύξηση του θορύβου από τις ολοένα και περισσότερες εκπομπές, θα οδηγήσουν σε μια νέα γενιά ολοκληρωμένων και κυκλωμάτων για αφαίρεση του θορύβου, μεγιστοποίησης του σηματοθορυβικού λόγου και ταχύτερη ανάκτηση των δεδομένων.

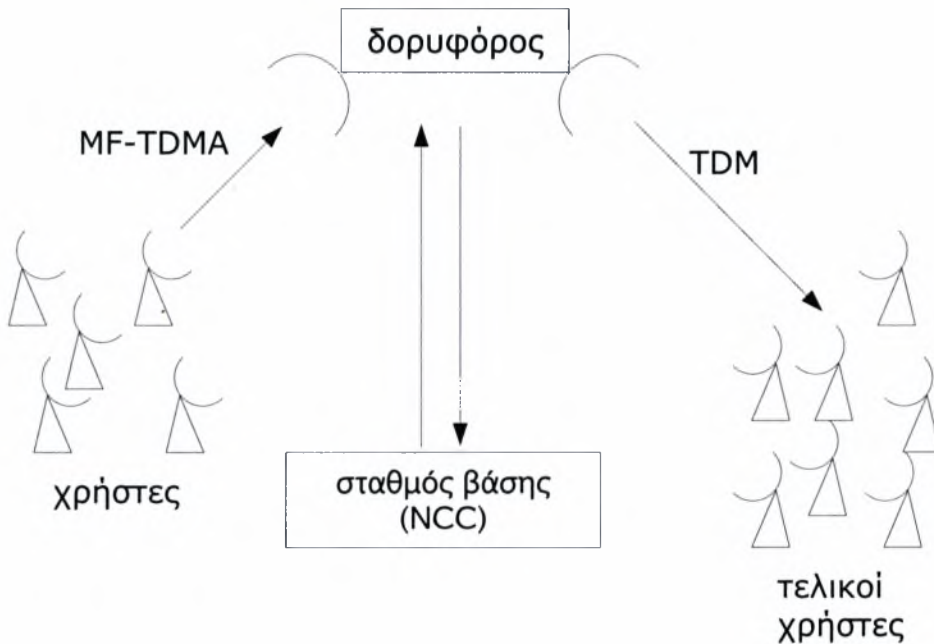
Επίσης, ήδη συστήματα διασύνδεσης των δορυφόρων μεταξύ τους έχουν εμφανιστεί και σύντομα θα γίνουν ο πιο συνηθισμένος τρόπος μεταφοράς δεδομένων ανά την υφήλιο. Θα ελαχιστοποιηθεί έτσι η ανάγκη για σταθμούς βάσης, ελαχιστοποιώντας παράλληλα και το κόστος ανάπτυξης και αναβάθμισης κάθε δικτύου.

1.2 Χαρακτηριστικά και προδιαγραφές λειτουργίας

Τα σύγχρονα δορυφορικά συστήματα, αν και προορίζονται για διαφορετικές χρήσεις, από πολυμεσικές εφαρμογές – κυρίως διανομή τηλεοπτικού σήματος και ραδιοφώνου – μέχρι και ψηφιακά δεδομένα

από το Internet, παρουσιάζουν σημαντικές ομοιότητες μεταξύ τους όσον αφορά τα χαρακτηριστικά τους και τις προδιαγραφές λειτουργίας¹.

Αν και το κάθε δορυφορικό σύστημα έχει τα δικά του χαρακτηριστικά, μια αφηρημένη εικόνα για το πως δουλεύουν βρίσκεται στο Σχήμα(1.1). Ο *επίγειος σταθμός βάσης (Network Control Centre ή NCC)* αναλαμβάνει να οργανώσει την επικοινωνία ανάμεσα στους χρήστες που στέλνουν τα δεδομένα και τους τελικούς χρήστες που τα λαμβάνουν. Το σύστημα ελέγχου της μεταγωγής της πληροφορίας και των πακέτων, δηλαδή η κατανομή των πόρων, ο καθορισμός της διαμεταγωγής για τον κάθε χρήστη κ.ά, είναι κατανεμημένο ανάμεσα στον δορυφόρο και το NCC.



Σχήμα 1.1: Τυπικό δορυφορικό σύστημα

Περισσότερες λεπτομέρειες για τα δορυφορικά συστήματα και τις κυριότερες παραμέτρους λειτουργίας τους ακολουθούν σε αυτό το κεφάλαιο.

1.2.1 Τροχιές

Ανά καιρούς έχουν προταθεί και χρησιμοποιούνται διάφορες τροχιές, από τις οποίες οι πιο σημαντικές είναι οι:

- **Low Earth Orbit (LEO: 200 – 1200 km** από την επιφάνεια της Γης), στην οποία ο κάθε δορυφόρος πραγματοποιεί έναν πλήρη κύκλο γύρω από την γη σε περίπου 90 λεπτά. Συνήθως σε αυτή τη τροχιά χρησιμοποιούνται *σμήνη δορυφόρων (satellite constellations)*. Την χρησιμοποιούν τα συστήματα Iridium και Globalstar.
- **Medium Earth Orbit (MEO ή Intermediate circular orbit (ICO): 1200 – 35790 km).**
- **Geosynchronous Orbit (GEO: 35790 km),** που επιτρέπει δορυφόρους που είναι σχεδόν ακίνητοι σε σχέση με το γήινο έδαφος.

¹Οι περισσότερες πληροφορίες έχουν ληφθεί από την Wikipedia [10] και το [1]

- **Geostationary Orbit (GSO)**, που είναι μια ειδική κατηγορία της GSO, στην οποία οι δορυφόροι βρίσκονται στην ζώνη Clark (Clarke belt) πάνω από τον ισημερινό.
- **High Earth Orbit (HEO: > 35790 km)**.

Για εφαρμογές τηλεπικοινωνιών και πολυμέσων συνήθως προτιμάται η GEO, επειδή οι δορυφόροι παραμένουν σταθεροί πάνω από κάποιο επίγειο σημείο και μπορούν εύκολοτερα να τους σταλλούν δεδομένα από έναν συγκεκριμένο σταθμό βάσης.

1.2.2 Περιοχές συχνοτήτων

Αν και το διαθέσιμο φάσμα είναι πολύ μεγάλο, οι σύγχρονοι δορυφόροι δουλεύουν κυρίως σε 2 διαφορετικές περιοχές συχνοτήτων, την

- **μπάντα Ka** (18 GHz – 40 GHz), που είναι στα 30 GHz για το uplink και στα 20 GHz για το downlink, και είναι η συνηθέστερη, και την
- **μπάντα V** (50 GHz – 75 GHz), που είναι στα 50 GHz για το uplink και στα 40 GHz για το downlink.

Μία και αρχίζει πλέον να διαφαίνεται ο κορεσμός της μπάντας Ka, ολόένα και περισσότερα συστήματα επιλέγουν την 'άδεια' ακόμα μπάντα V.

1.2.3 Διαμόρφωση

Για την διαμόρφωση του σήματος επιλέγεται κάποια mPSK. Συνήθως, για περιπτώσεις υψηλού θορύβου επιλέγεται η BPSK ($m = 2$) και για κανονικές περιπτώσεις η QPSK ($m = 4$), αφού σύμφωνα με το [9] προσφέρει την καλύτερη σχέση μεταξύ απαιτήσεων ισχύος και εύρους ζώνης. Σχετικός είναι ο Πίνακας(1.1).

Τιμή του m	$\frac{(\text{Εύρος ζώνης})_{\text{mPSK}}}{(\text{Εύρος ζώνης})_{\text{BPSK}}}$	$\frac{(\text{Μέση ισχύς})_{\text{mPSK}}}{(\text{Μέση ισχύς})_{\text{BPSK}}}$
4	0.5	0.34 dB
8	0.333	3.91 dB
16	0.25	8.52 dB

Πίνακας 1.1: Σύγκριση απαιτήσεων ισχύος-εύρους ζώνης μεταξύ mPSK και BPSK. Πιθανότητα σφάλματος συμβόλων = 10^{-4}

1.2.4 Μοντέλο πρόσβασης

Όσον αφορά το μοντέλο πρόσβασης (access mode) στο uplink, κυριαρχούν δύο τεχνικές: η *MF-TDMA* (*Multi-Frequency Time Division Multiple Access*) και η *MF-CDMA* (*Multi-Frequency Code Division Multiple Access*). Κύριο χαρακτηριστικό τους είναι πως χρησιμοποιούν πολλαπλά συχνότητες, εξ ου και ο χαρακτηρισμός MF (multifrequency).

- Η MF-TDMA στηρίζεται στην λογική ότι το κάθε πλαίσιο (frame) του TDMA αποτελείται από χρονοθυρίδες (slots), οι οποίες είναι στην ουσία FDMA πλαίσια. Το κάθε FDMA πλαίσιο αποτελείται από θυρίδες που κάθε μία αντιστοιχεί σε έναν χρήστη².

Ο κάθε χρήστης έχει το δικαίωμα να εκπέμπει σε οποιαδήποτε από τις n διαθέσιμες συχνότητες και σε οποιοδήποτε από τις p θυρίδες.

Παρόλα αυτά για πρακτικούς λόγους, κυρίως για την υπερβολική ισχύ που απαιτείται και το κόστος του επιπλέον εξοπλισμού, ο κάθε χρήστης μπορεί να εκπέμπει σε μία συχνότητα σε κάθε θυρίδα. Έτσι επιτυγχάνεται πολυπλεξία και πολλαπλή πρόσβαση για όλους τους χρήστες.

- Για την MF-CDMA χρησιμοποιείται η λογική της χρήσης κωδικών (chipping codes) που προσπαθούν να πετύχουν ορθογωνιότητα των σημάτων, όπως και στην κλασσική CDMA.

Υπάρχει η δυνατότητα χρήσης διαφορετικών συχνοτήτων κάθε φορά και ο κάθε χρήστης προσδιορίζεται από τον διαφορετικό κωδικό του. Έτσι, αν χρειάζεται (για παράδειγμα υπάρχει υπερβολικός θόρυβος ή το κανάλι έχει πολλούς χρήστες) μπορεί να αλλάζει συχνότητα, συγχρονισμένος πάντα με τον δέκτη στο δορυφόρο.

Αν και προσφέρει κάποια σοβαρά πλεονεκτήματα έναντι της MF-TDMA, όπως αποδοτικότερη χρήση του φάσματος και καλύτερο έλεγχο του multi-path fading, χρειάζεται καλύτερο συγχρονισμό μεταξύ πομπού και δέκτη.

Δεδομένου ότι σε GEO δορυφόρους το κανάλι έχει γκαουσιανό θόρυβο και το multi-path fading είναι σχεδόν ανύπαρκτο, το συνηθέστερο σύστημα είναι το MF-TDMA. Αυτό κυρίως συμβαίνει αφού από την φύση του το MF-TDMA επιτρέπει πιο απλούς πομποδέκτες, όπως και το γεγονός ότι οι chipping codes για δορυφόρους που καλύπτουν μεγάλες περιοχές, δεν θα επαρκούσαν για να καλύψουν τον μεγάλο αριθμό των χρηστών χωρίς μείωση της ταχύτητας.

Για το downlink, αν και επαφίεται στις ανάγκες και στις επιλογές του σχεδιαστή, η ανάγκη για υποστήριξη πολλών τελικών χρηστών³ υπαγορεύει συνήθως την χρήση της TDM (*Time Division Multiplexing*).

1.2.5 Καθολικός συγχρονισμός του συστήματος

Όσον αφορά τον καθολικό συγχρονισμό ρυθμού (*global synchronization*) του συστήματος δορυφόρου χρήστες υπάρχουν δύο μέθοδοι που χρησιμοποιούνται – σχετικό είναι το Σχήμα (1.2):

- **bit synchronized** σύστημα, στο οποίο κάθε πλαίσιο πληροφορίας έχει σταθερό μήκος T_f και απαρτίζεται από πακέτα μεταβλητού μήκους που ανήκουν σε διαφορετικούς χρήστες.

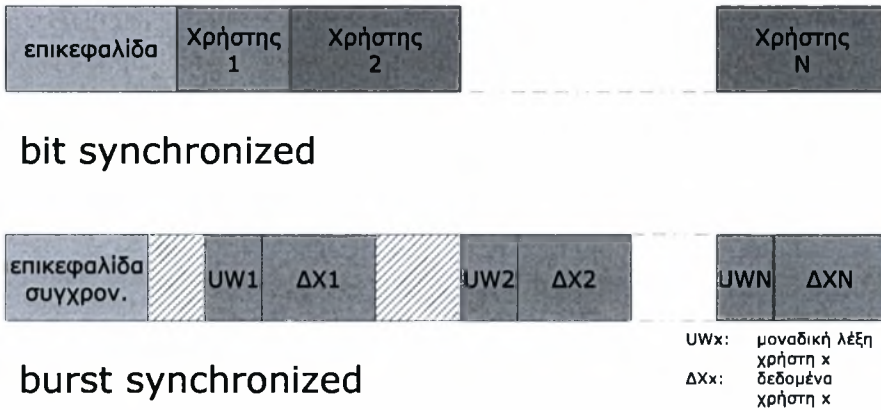
Επιβάλλει απόλυτο συγχρονισμό όλων των εμπλεκόμενων συστημάτων – δορυφόρου και χρηστών – και το πλαίσιο έχει μια επικεφαλίδα που καθορίζει από ποιον προέρχεται κάθε πακέτο.

- **burst synchronized** σύστημα, που κάθε σταθερό frame μήκους T_f απαρτίζεται από πλαίσια των χρηστών. Αν και υπάρχει η επικεφαλίδα, είναι αποκλειστικά και μόνον για συγχρονισμό μεταξύ δορυφόρου-χρήστη.

Το πλαίσιο του κάθε χρήστη μπορεί να είναι μεταβλητού μήκους και έχει σαν επικεφαλίδα κάποια μοναδική λέξη (unique word), που ξεχωρίζει τους χρήστες.

²σαν χρήστης θα αναφέρεται από εδώ και στο εξής οποιαδήποτε υποδομή χρησιμοποιείται για την αποστολή δεδομένων στον δορυφόρο

³σαν τελικός χρήστης ορίζεται κάθε υποδομή η οποία δέχεται το σήμα από τον δορυφόρο



Σχήμα 1.2: bit synchronized σύστημα vs burst synchronized

Είναι πιο απλό σύστημα, αφού δεν επιβάλλει τον συγχρονισμό όλων των χρηστών και του δορυφόρου. Σε ένα burst synchronized σύστημα, το ρολόι βρίσκεται στο δορυφόρο και ο κάθε χρήστης συγχρονίζεται με αυτό.

Μια και ήδη υπάρχει κάποιες μορφές πολυπλεξία λόγω MF-TDMA, συνήθως προτιμώνται τα burst synchronized συστήματα.

1.3 Ατέλειες κατά την διαμόρφωση και την αποδιαμόρφωση

Όπως σε κάθε σύστημα που δουλεύει σε πραγματικές συνθήκες, έτσι και στα δορυφορικά υπάρχει ένας αριθμός από παραμέτρους που επηρεάζουν την αποδοτικότητα του συστήματος. Οι περισσότερες από αυτές είναι τυχαίες και επηρεάζονται από κατασκευαστικές ατέλειες και τις συνθήκες του περιβάλλοντος.

Οι κυριότερες παράμετροι που επηρεάζουν την λειτουργία του συστήματος είναι

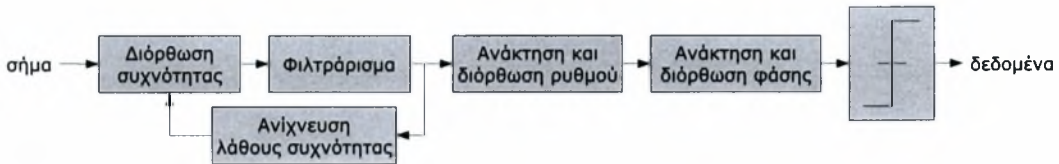
- ο θόρυβος από το κανάλι,
- ο θόρυβος φάσης που προκαλείται από τους ταλαντωτές στους διαμορφωτές και τους αποδιαμορφωτές,
- κάποιο υπόλοιπο συχνότητας, λόγω μη σωστής ανάκτησης της συχνότητας του φέροντος στους αποδιαμορφωτές,
- ο ελλειπής συγχρονισμός μεταξύ πομπού και δέκτη,
- το φαινόμενο Doppler και
- το multi-path fading που υποβαθμίζει την συνολική ποιότητα και την ένταση του χρήσιμου σήματος.

Η παρούσα εργασία ασχολείται κυρίως με την εξάλειψη της επίδρασης του θορύβου φάσης και του υπολοίπου συχνότητας από το σήμα.

1.4 Αρχιτεκτονική αποδιαμορφωτή

Εφόσον ακολουθείται η λογική του OBP, πάνω στον δορυφόρο έχει αναπτυχθεί μια πολύπλοκη αρχιτεκτονική που αναλαμβάνει την συνολική επεξεργασία του σήματος.

Από τα πιο σημαντικά κομμάτια της αρχιτεκτονικής είναι ο δέκτης/αποδιαμορφωτής, αφού είναι και η νέα προσθήκη στον δορυφόρο – αναλαμβάνει να ξεχωρίσει τα δεδομένα κάθε χρήστη, να διορθώσει τα λαμβανόμενα σύμβολα και να επιλύσει τα προβλήματα συγχρονισμού. Μία απλοποιημένη δομή του υπάρχει στο Σχήμα(1.3).



Σχήμα 1.3: Δομή αποδιαμορφωτή

1.4.1 Λειτουργίες

Οι λειτουργίες του αποδιαμορφωτή είναι:

- το φιλτράρισμα του σήματος (signal filtering) για την μείωση του θορύβου,
- ο συγχρονισμός του ρυθμού (timing synchronization),
- η ανάκτηση της συχνότητας (frequency synchronization) και της διαφοράς φάσης της φέρουσας (phase synchronization),
- η ανάκτηση του σήματος από το κωδικοποιημένο σε περιπτώσεις όπου χρησιμοποιείται CDMA (decoding) και
- ο συγχρονισμός στην μοναδική λέξη σε συστήματα burst synchronized.

Δεδομένου ότι η παρούσα εργασία ασχολείται κυρίως με το συγχρονισμό της συχνότητας και δη της φάσης, θα θεωρείται ότι οι δομές που κάνουν οποιοδήποτε άλλου τύπου συγχρονισμού έχουν τέλειο αποτέλεσμα.

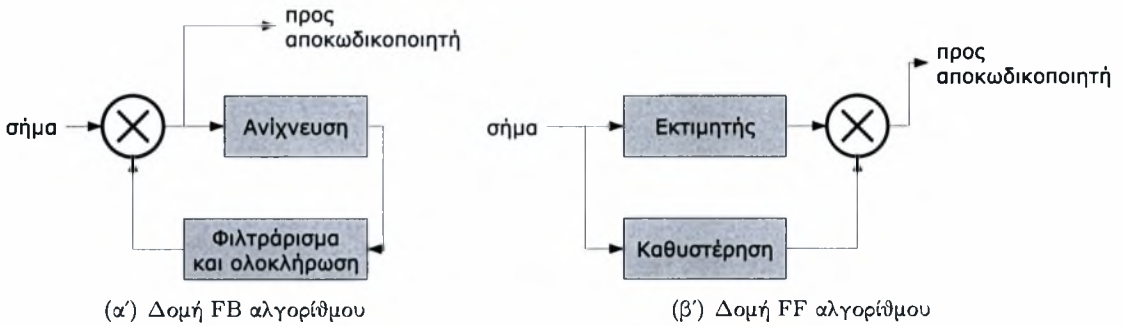
1.4.2 Συγχρονισμός

Οι αλγόριθμοι συγχρονισμού μπορούν να χωριστούν σε κατηγορίες ανάλογα με το αν είναι με ανατροφοδότηση ή όχι και με το αν χρησιμοποιούν την πληροφορία που φέρουν τα σύμβολα ή όχι.

Κατηγοριοποίηση αλγορίθμων συγχρονισμού

FB-FF Η πρώτη κατηγοριοποίηση – βλ. Σχήμα(1.4) – γίνεται σε

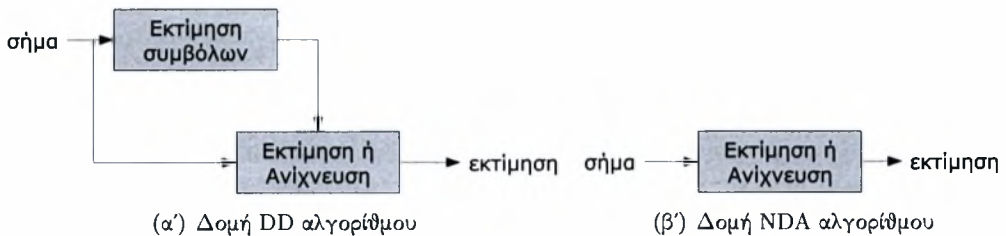
- αλγόριθμους με ανατροφοδότηση ή **feedback (FB)**, όπου ο αλγόριθμος υποβοηθείται από προηγούμενη πληροφορία της παραμέτρου για την οποία πρέπει να γίνει συγχρονισμός και σε



Σχήμα 1.4: Αλγόριθμοι feedback και feed-forward

- **αλγορίθμους χωρίς ανατροφοδότηση ή feed-forward (FF)**, στους οποίους γίνεται εκτίμηση της παραμέτρου μόνον από την υπάρχουσα κάθε φορά πληροφορία. Εναλλακτικά, απαντάται σαν αλγόριθμοι εκτίμησης ή εκτιμητές.

Είναι προφανές πως οι αλγόριθμοι FF υστερούν σε ακρίβεια υπολογισμού, αλλά υπερτερούν κατά πολύ σε ταχύτητα. Για τις δορυφορικές επικοινωνίες, όπου κυριαρχούν οι υψηλές ταχύτητες επικοινωνίας συνήθως χρησιμοποιούνται συστήματα συγχρονισμού βασισμένα στους FF αλγορίθμους.



Σχήμα 1.5: Αλγόριθμοι Decision Directed και Non Data Aided

DD-NDA Η δεύτερη κατηγοριοποίηση γίνεται βάση του αν χρησιμοποιούν ή όχι την πληροφορία που μεταφέρουν τα σήματα. Η βασική τους δομή εμφανίζεται στο Σχήμα(1.5). Έτσι, χωρίζονται σε

- **κατευθυνόμενους από αποφάσεις αλγορίθμους ή Decision Directed (DD)**, που υπαγορεύουν την εύρεση της πληροφορίας που μεταφέρουν τα σύμβολα και τη χρήση αυτής για να υπολογίσουν πιο αποτελεσματικά την παράμετρο – επομένως, χρειάζεται μια μονάδα ανίχνευσης συμβόλων που θα υποβοηθεί την μονάδα που υλοποιεί τον αλγόριθμο – και σε
- **μη υποβοηθούμενους από δεδομένα αλγορίθμους ή Non Data Aided (NDA)**, που χρησιμοποιούν μόνον κάποιο μέσο όρο στα σύμβολα που δέχονται, χωρίς να κάνουν οποιαδήποτε χρήση της πληροφορίας που δέχονται. Παρόλα αυτά, πρέπει να υπάρχει κάποιος τρόπος αφαίρεσης της πληροφορίας, ώστε να μην επηρεάζει τους υπολογισμούς.

Συγχρονισμός ρυθμού

Για να υπάρχει σωστή παράδοση της πληροφορίας, πρέπει ο πομπός και ο δέκτης να είναι συγχρονισμένοι, όσον αφορά που ξεκινά και που τελειώνει ένα σύμβολο. Συνεπώς, κατά το *συγχρονισμό*

ρυθμού, το σύστημα πρέπει να ξεχωρίσει τα σύμβολα μεταξύ τους και να απαλείψει τυχόν επικαλύψεις ή διασυμβολικές παρεμβολές (ISI^4).

Ο συγχρονισμός ρυθμού επιτυγχάνεται χρησιμοποιώντας εκτιμητές Oerder-Meyr για συστήματα τύπου burst-synchronized και συνήθως ανιχνευτή Gardner για bit-synchronized, όπως αναφέρεται στο [1]. Στην παρούσα εργασία δεν θα αναλυθούν τέτοιου τύπου εκτιμητές.

Συγχρονισμός συχνότητας φέρουσας

Για να μπορέσει να επιτευχθεί η επικοινωνία ανάμεσα σε χρήστη και δορυφόρο, είναι αναγκαία η ανάκτηση και η διόρθωση της συχνότητας φέρουσας. Τα αντίστοιχα κυκλώματα κάνουν ανάκτηση της συχνότητας και σε επόμενα βήματα γίνεται η αναγκαία διόρθωση.

Η διόρθωση του θορύβου φάσης αναγκαστικά εμπίπτει στην κατηγορία του συγχρονισμού στη συχνότητα. Ταλαντωτές (oscillators) σε πομπό και δέκτη, που αναλαμβάνουν να φέρουν το σήμα από τη ζώνη βάσης σε κάποια συχνότητα και το αντίστροφο, εισάγουν λόγω κατασκευαστικών ατελειών και ανοχών κάποια μετατόπιση στην φάση του σήματος. Για να υπάρξει σωστή επικοινωνία, αυτή η διαφορά στην φάση πρέπει να βρεθεί και να αφαιρεθεί.

Συνήθως, ο τρόπος εύρεσης του θορύβου φάσης είναι με κάποιον αλγόριθμο χωρίς ανατροφοδότηση (αλγόριθμος εκτίμησης), που προσφέρει καλύτερη ταχύτητα από τους αλγορίθμους με ανατροφοδότηση, αλλά χαμηλότερη ακρίβεια. Ως εκ τούτου, στην παρούσα εργασία μελετώνται κυρίως οι αλγόριθμοι εκτίμησης φάσης.

1.5 Υποθέσεις εργασίας

Στην παρούσα εργασία μελετάται κάποιο θεωρητικό σύστημα, για το οποίο έχουν γίνει οι παρακάτω παραδοχές:

- βρίσκεται σε GEO τροχιά,
- δουλεύει στην μπάντα Ka,
- έχει μόλις έναν χρήστη, ο οποίος χρησιμοποιεί MF-TDMA,
- τα δεδομένα διαμορφώνονται χρησιμοποιώντας QPSK,
- δεν χρησιμοποιείται κωδικοποίηση (coding),
- έχει γίνει τέλειος συγχρονισμός ρυθμού (timing synchronization),
- έχει γίνει σχεδόν τέλεια ανάκτηση φέρουσας (frequency synchronization) – παρόλα αυτά έχει παραμείνει κάποιο υπόλοιπο συχνότητας,
- ο θόρυβος του καναλιού είναι τυχαίος και
- ο θόρυβος φάσης είναι μία σταθερή αλλά άγνωστη τιμή.

Οι κυριότερες παράμετροι μείωσης του σηματοθορυβικού λόγου είναι ο θόρυβος του καναλιού, ο θόρυβος φάσης και το υπόλοιπο της συχνότητας.

⁴Intersymbol Interference

1.6 Οργάνωση του κειμένου

Η εργασία, εκτός από αυτό το πρώτο κεφάλαιο της εισαγωγής, αποτελείται από τα

- **Κεφάλαιο 2**, που παρουσιάζει το πιο σύνθητες σύστημα διαμόρφωσης στις δορυφορικές επικοινωνίες, το απλό QPSK, καθώς και την επίδραση του θορύβου φάσης κατά την αποδιαμόρφωση των συμβόλων,
- **Κεφάλαιο 3**, που αναλύει τον NDA εκτιμητή Viterbi & Viterbi για την εκτίμηση φάσης, την βέλτιστη μορφή του και μια επαναληπτική μορφή του,
- **Κεφάλαιο 4**, που μελετά την οικογένεια των εκτιμητών βαρυκέντρου και
- **Κεφάλαιο 5**, που αναλύει τις επιλεγμένες παραμέτρους των προσομοιώσεων και παρουσιάζει τις προσομοιώσεις για τους εκτιμητές Viterbi & Viterbi και βαρυκέντρου, με παρατηρήσεις και συμπεράσματα.

Στο τέλος της εργασίας ο αναγνώστης μπορεί να βρει τα

- **Παράρτημα Α'**, που έχει ένα σύνολο από χρήσιμες τριγωνομετρικές και μιγαδικές σχέσεις που χρησιμοποιήθηκαν σε αυτήν την εργασία και
- **Παράρτημα Β'**, που έχει τον κώδικα MatLab που χρησιμοποιήθηκε

και την

- **Βιβλιογραφία**, με αναφορές σε εργασίες τρίτων και ιστοσελίδες του Διαδικτύου, από όπου αντλήθηκαν οι πληροφορίες.

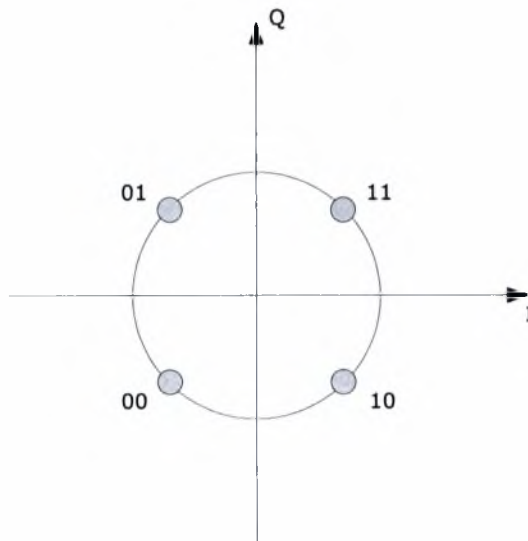
Κεφάλαιο 2

QPSK

2.1 Εισαγωγή

Η *Quadrature Phase Shift Keying* ή *QPSK* (ορθογωνική μεταλλαγή μετατόπισης φάσης) είναι μια ψηφιακή τεχνική διαμόρφωσης, στην οποία τα δεδομένα κωδικοποιούνται στην φάση του φέροντος κύματος. Υπάρχουν δύο κύριοι τρόποι αναπαράστασης της πληροφορίας

- κωδικοποιώντας την πληροφορία στην φάση του σήματος, με τον αποκωδικοποιητή να απαιτείται να έχει κάποιο σήμα αναφοράς ώστε να το συγκρίνει το λαμβανόμενο και να αποφασίσει για το πιο σύμβολο έλαβε και
- κωδικοποιώντας την πληροφορία στις αλλαγές φάσης (differential schemes), που ως ένα βαθμό δεν απαιτεί σήμα αναφοράς στον αποκωδικοποιητή, εκτός από κάποια σύμβολα που είναι σύμβολα αναφοράς. Αυτή συνήθως αναφέρεται σαν DQPSK (Differential QPSK).



Σχήμα 2.1: QPSK constellation diagram

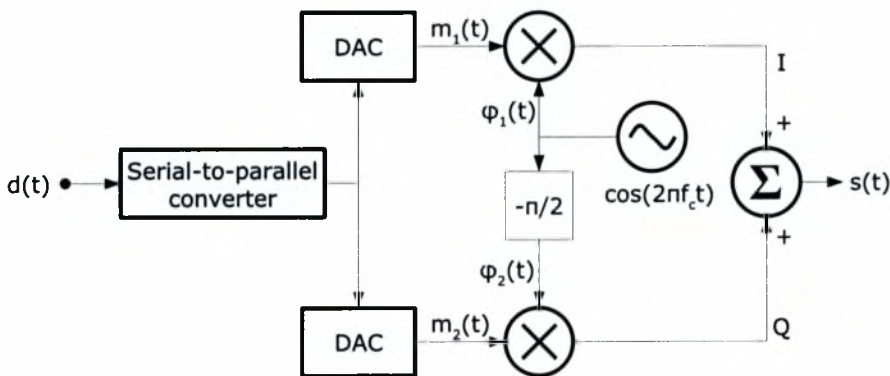
Στην QPSK η φάση του φέροντος μπορεί να πάρει 4 διαφορετικές τιμές, δηλαδή μπορεί να κωδικοποιήσει 4 διαφορετικά σύμβολα, που διαφέρουν κατά $\frac{\pi}{2}$. Πρόκειται για μια mPSK με $m = 4$ ή αλλιώς 4-PSK. Η διαμόρφωση κατά QPSK μπορεί να θεωρηθεί και σαν δύο ανεξάρτητες και διαμορφωμένες κατά BPSK φέρουσες, I και Q , που διαφέρουν κατά $\frac{\pi}{2}$. Στο Σχήμα(2.1) εμφανίζονται τα σύμβολα πάνω στον κύκλο.

Εφόσον υπάρχουν 4 διαφορετικά σύμβολα, τότε το κάθε σύμβολο μπορεί κωδικοποιεί μέχρι 2 bits πληροφορίας. Αυτό το σύνθετο σύμβολο ονομάζεται *dibit*. Στον Πίνακα(2.1) αναφέρονται οι αντιστοιχίες μεταξύ dibits και συμβόλων.

Σύμβολο	φάση	dibit	Διάνυσμα τιμών
s_1	$\frac{\pi}{4}$	10	$+\sqrt{\frac{E}{2}}, -\sqrt{\frac{E}{2}}$
s_2	$\frac{3\pi}{4}$	00	$-\sqrt{\frac{E}{2}}, -\sqrt{\frac{E}{2}}$
s_3	$\frac{5\pi}{4}$	01	$-\sqrt{\frac{E}{2}}, +\sqrt{\frac{E}{2}}$
s_4	$\frac{7\pi}{4}$	11	$+\sqrt{\frac{E}{2}}, +\sqrt{\frac{E}{2}}$

Πίνακας 2.1: QPSK σύμβολα και dibits

2.1.1 Διαμορφωτής QPSK

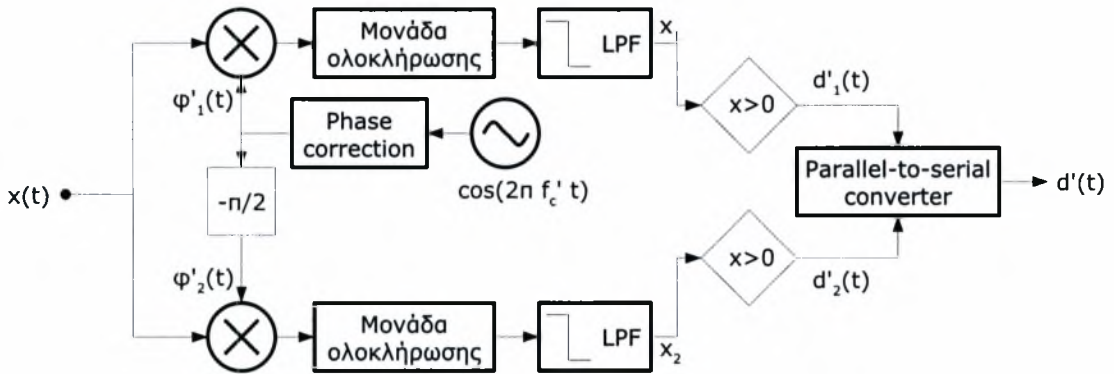


Σχήμα 2.2: Διαμορφωτής QPSK

Στο Σχήμα(2.2) εικονίζεται μια απλοποιημένη εκδοχή μίας μονάδας διαμόρφωσης κατά QPSK, όπου $d(t)$ είναι μία ροή bit, $m_1(t)$ τα άρτια bits και $m_2(t)$ τα περιττά με τιμές $+\sqrt{E_b}$ για το 1 και $-\sqrt{E_b}$ για το 0. Η έξοδος του διαμορφωτή είναι το σήμα $s(t) = m_1(t) \cdot \phi_1(t) + m_2(t) \cdot \phi_2(t)$ το οποίο και αποστέλλεται. Τα $\phi_1(t)$ και $\phi_2(t)$ είναι οι συναρτήσεις ορθοκανονικής βάσης, από την διαδικασία ορθογωνοποίησης Gram-Schmidt, που πολλαπλασιαζόμενες με τα $m_1(t)$ και $m_2(t)$ δίνουν τις δύο BPSK ροές I και Q .

2.1.2 Αποδιαμορφωτής QPSK

Στον δέκτη που χρησιμοποιεί τον αποδιαμορφωτή του Σχήματος(2.3), το σύστημα δέχεται το σήμα $x(t)$, το οποίο είναι το σήμα $s(t)$ που αποστάληκε από κάποιον πομπό με τυχόν απώλειες και θόρυβο από το κανάλι. Στο υπόλοιπο του κειμένου υποθέτουμε ότι έχουμε *αθροιστικό λευκό γκαουσιανό*



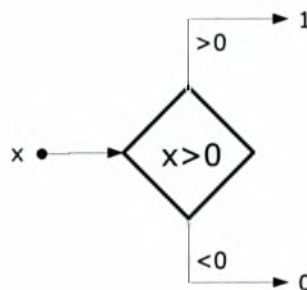
Σχήμα 2.3: Αποδιαμορφωτής QPSK

θόρυβο ή *AWGN*¹.

Είναι καθήκον του αποδιαμορφωτή να κάνει *ανάκτηση χρονισμού* (*timing synchronization*), ώστε να διαχωρίσει τα σύμβολα, και *ανάκτηση συχνότητας* (*frequency synchronization*) του φέροντος κύματος συχνότητας f_c όσο το δυνατόν καλύτερα.

Σκοπός είναι, αν f'_c η ανακτημένη συχνότητα, να ελαχιστοποιηθεί η διαφορά $|f'_c - f_c|$. Το υπόλοιπο που απομένει μετά την ανάκτηση, δηλαδή η διαφορά, είναι το $\Delta f = f_c - f'_c$. Στο [2] αναφέρεται πως $(\Delta f)T \ll 1$, αν θεωρηθεί πως οι πηγές συχνοτήτων είναι αρκετά σταθερές και οι διακυμάνσεις στην συχνότητα ανάμεσα σε διαδοχικά σύμβολα παρακολουθούνται και διορθώνονται. Παρόλα αυτά, είναι μία από τις πηγές που προκαλούν προβλήματα στην σωστή αποδιαμόρφωση του σήματος.

Άξια αναφοράς επίσης είναι η μονάδα διόρθωσης φάσης, με την οποία ασχολείται η παρούσα εργασία και μπορεί να είναι είτε κάποια μονάδα NDA ή decision directed, ανάλογα με την σχεδιαστική προσέγγιση του αποκωδικοποιητή.



Σχήμα 2.4: Διάταξη απόφασης

Εφόσον γίνουν οι απαραίτητες διορθώσεις, το σήμα αποδιαμορφώνεται και ολοκληρώνεται. Στο Σχήμα(2.4) εμφανίζεται μία μονάδα απόφασης για την τιμή του bit που θα δώσει για την κάθε αναλογική είσοδο το σύστημα, μετά το στάδιο της ολοκλήρωσης.

¹AWGN: additive white gaussian noise

2.2 Ιδανική περίπτωση

2.2.1 Ιδανικό σύμβολο

Ελλείψει θορύβου, τα QPSK σύμβολα μπορούν να εκφραστούν σαν

$$s_i(t) = \begin{cases} \sqrt{\frac{2E_s}{T}} \cos [2\pi f_c t + (2i - 1)\frac{\pi}{4}] & , 0 \leq t \leq T, i \in [1, 2, 3, 4] \\ 0 & , \text{αλλιώς} \end{cases} \quad (2.1)$$

με E_s ενέργεια συμβόλου, T περίοδο συμβόλου και f_c συχνότητα φέροντος.

Η συχνότητα f_c του φέροντος συνδέεται με την περίοδο T ενός συμβόλου σύμφωνα με την σχέση

$$f_c = n_c \frac{1}{T}, n_c \in \mathbb{Z} \quad (2.2)$$

Μια ισοδύναμη έκφραση της (2.1) είναι η

$$s_i(t) = \begin{cases} \sqrt{\frac{2E_s}{T}} \cos [(2i - 1)\frac{\pi}{4}] \cos(2\pi f_c t) \\ -\sqrt{\frac{2E_s}{T}} \sin [(2i - 1)\frac{\pi}{4}] \sin(2\pi f_c t) & , 0 \leq t \leq T, i \in [1, 2, 3, 4] \\ 0 & , \text{αλλιώς} \end{cases} \quad (2.3)$$

2.2.2 Ιδανικό σήμα

Ένας συρμός συμβόλων μπορεί να περιγραφεί με την συνάρτηση *συνεχούς χρόνου, διακριτού πλάτους* $S(t)$. Το πεδίο τιμών της $S(t)$ είναι το $[1, 2, 3, 4]$ και έχει την ίδια τιμή σε κάθε χρονικό διάστημα $[iT, (i + 1)T]$, $i \in \mathbb{Z}$. Έτσι, η συνάρτηση που δίνει την φάση του σήματος μπορεί να οριστεί ως

$$Z(t) = [2S(t) - 1] \frac{\pi}{4} \quad (2.4)$$

Από τις (2.3) και (2.4)

$$s(t) = \sqrt{\frac{2E_s}{T}} \cos(Z(t)) \cos(2\pi f_c t) - \sqrt{\frac{2E_s}{T}} \sin(Z(t)) \sin(2\pi f_c t) \quad (2.5)$$

2.2.3 Συναρτήσεις ορθοκανονικής βάσης

Είναι γνωστό πως από την διαδικασία ορθογωνοποίησης Gram-Schmidt προκύπτουν οι συναρτήσεις ορθοκανονικής βάσης

$$\phi_1(t) = \sqrt{\frac{2}{T}} \cos(2\pi f_c t) \quad (2.6)$$

$$\phi_2(t) = \sqrt{\frac{2}{T}} \sin(2\pi f_c t) \quad (2.7)$$

Επίσης, ισχύουν οι παρακάτω σχέσεις

$$\begin{aligned}
 \int_0^T \phi_1(t)\phi_1(t) dt &= \int_0^T \frac{2}{T} \cos^2(2\pi f_c t) dt \\
 &= \int_0^T \frac{2}{T} \frac{1}{2} (1 + \cos(2 \cdot 2\pi f_c t)) dt \\
 &= 1 + \int_0^T \frac{1}{T} \cos(2\pi(2f_c)t) dt
 \end{aligned} \tag{2.8}$$

$$\begin{aligned}
 \int_0^T \phi_2(t)\phi_2(t) dt &= \int_0^T \frac{2}{T} \sin^2(2\pi f_c t) dt \\
 &= \int_0^T \frac{2}{T} \frac{1}{2} (1 - \cos(2 \cdot 2\pi f_c t)) dt \\
 &= 1 - \int_0^T \frac{1}{T} \cos(2\pi(2f_c)t) dt
 \end{aligned} \tag{2.9}$$

$$\begin{aligned}
 \int_0^T \phi_1(t)\phi_2(t) dt &= \int_0^T \frac{2}{T} \cos(2\pi f_c t) \sin(2\pi f_c t) dt \\
 &= \int_0^T \frac{2}{T} \frac{\sin(2 \cdot 2\pi f_c t) + \sin(0)}{2} dt \\
 &= \int_0^T \frac{1}{T} \sin(2\pi(2f_c)t) dt
 \end{aligned} \tag{2.10}$$

Εφόσον $T \cdot f_c = n_c$ και $n_c \in \mathbb{Z}$, τότε

$$\int_0^T \phi_1(t)\phi_1(t) dt = 1 \tag{2.11}$$

$$\int_0^T \phi_2(t)\phi_2(t) dt = 1 \tag{2.12}$$

$$\int_0^T \phi_1(t)\phi_2(t) dt = 0 \tag{2.13}$$

2.2.4 Ορισμός διανύσματος παρατήρησης

Το διάνυσμα παρατηρούμενων τιμών για μία περίοδο T , ή αλλιώς για ένα σύμβολο, ορίζεται ως

$$\mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \tag{2.14}$$

με

$$x_1 = \int_0^T x(t)\phi_1(t) dt \tag{2.15}$$

$$x_2 = \int_0^T x(t)\phi_2(t) dt \tag{2.16}$$

2.2.5 Διάνυσμα παρατήρησης ιδανικού σήματος

Έστω ότι ιδανικό σήμα $s(t)$ που λαμβάνεται και αποδιαμορφώνεται από ιδανικό δέκτη². Επίσης, είναι σκόπιμο να αναφερθεί πως η $Z(t)$ παραμένει σταθερή στο χρονικό διάστημα $[iT, (i+1)T]$, $i \in \mathbb{Z}$.

Αποδιαμορφώνοντας

$$\begin{aligned}
 x_1 &= \int_0^T s(t)\phi_1(t) dt \\
 &= \int_0^T \left[\sqrt{\frac{2E_s}{T}} \cos(Z(t)) \cos(2\pi f_c t) - \sqrt{\frac{2E_s}{T}} \sin(Z(t)) \sin(2\pi f_c t) \right] \sqrt{\frac{2}{T}} \cos(2\pi f_c t) dt \\
 &= \sqrt{E_s} \left[\int_0^T \frac{2}{T} \cos(Z(t)) \cos^2(2\pi f_c t) dt - \int_0^T \frac{2}{T} \sin(Z(t)) \sin(2\pi f_c t) \cos(2\pi f_c t) dt \right] \\
 &= \sqrt{E_s} \left[\cos(Z(t)) \int_0^T \frac{2}{T} \cos^2(2\pi f_c t) dt - \sin(Z(t)) \int_0^T \frac{2}{T} \sin(2\pi f_c t) \cos(2\pi f_c t) dt \right] \\
 &= \sqrt{E_s} [\cos(Z(t)) \cdot 1 - \sin(Z(t)) \cdot 0] \Rightarrow \\
 x_1 &= \sqrt{E_s} \cos(Z(t)) \tag{2.17}
 \end{aligned}$$

Ομοίως,

$$\begin{aligned}
 x_2 &= \int_0^T s(t)\phi_2(t) dt \\
 &= \int_0^T \left[\sqrt{\frac{2E_s}{T}} \cos(Z(t)) \cos(2\pi f_c t) - \sqrt{\frac{2E_s}{T}} \sin(Z(t)) \sin(2\pi f_c t) \right] \sqrt{\frac{2}{T}} \sin(2\pi f_c t) dt \\
 &= \sqrt{E_s} \left[\int_0^T \frac{2}{T} \cos(Z(t)) \cos(2\pi f_c t) \sin(2\pi f_c t) dt - \int_0^T \frac{2}{T} \sin(Z(t)) \sin^2(2\pi f_c t) dt \right] \\
 &= \sqrt{E_s} \left[\cos(Z(t)) \int_0^T \frac{2}{T} \cos(2\pi f_c t) \sin(2\pi f_c t) dt - \sin(Z(t)) \int_0^T \frac{2}{T} \sin^2(2\pi f_c t) dt \right] \\
 &= \sqrt{E_s} [\cos(Z(t)) \cdot 0 - \sin(Z(t)) \cdot 1] \Rightarrow \\
 x_2 &= -\sqrt{E_s} \sin(Z(t)) \tag{2.18}
 \end{aligned}$$

Το διάνυσμα παρατήρησης για την ιδανική περίπτωση είναι

$$\mathbf{x} = \begin{bmatrix} \sqrt{E_s} \cos(Z(t)) \\ -\sqrt{E_s} \sin(Z(t)) \end{bmatrix} \tag{2.19}$$

2.3 Περίπτωση με AWGN θόρυβο

2.3.1 Σήμα με AWGN θόρυβο

Μέχρι τον δέκτη, το σήμα υφίσταται απώλειες που μπορούν να μοντελοποιηθούν σαν πολλαπλασιασμός του σήματος με έναν παράγοντα $G(t)$. Αυτός ο παράγοντας μπορεί να θεωρηθεί σταθερός

²δηλαδή ο δέκτης δεν προσθέτει οποιούδηποτε είδους θόρυβο

για κάθε σύμβολο ώστε να μην εισαχθεί αχρείαστη πολυπλοκότητα στους υπολογισμούς, χωρίς να υπάρξει βλάβη της γενικότητας. Επιπλέον υπάρχει AWGN $w(t)$.

Το νέο σήμα με τον θόρυβο και τις απώλειες μπορεί να εκφραστεί σαν

$$\begin{aligned} s_w(t) &= G(t)s(t) + w(t) \\ &= G(t) \left[\sqrt{\frac{2E_s}{T}} \cos(Z(t)) \cos[2\pi f_c t] - \sqrt{\frac{2E_s}{T}} \sin(Z(t)) \sin[2\pi f_c t] \right] + w(t) \Rightarrow \\ s_w(t) &= A(t) \sqrt{\frac{2}{T}} \cos(Z(t)) \cos[2\pi f_c t] - A(t) \sqrt{\frac{2}{T}} \sin(Z(t)) \sin[2\pi f_c t] + w(t) \end{aligned} \quad (2.20)$$

με

$$A(t) = G(t) \sqrt{E_s} \quad (2.21)$$

2.3.2 Διάνυσμα παρατήρησης σήματος με AWGN θόρυβο

Όταν υπάρχει θόρυβος στο κανάλι οι εξισώσεις είναι παρόμοιες με την ιδανική περίπτωση. Έτσι,

$$\begin{aligned} x_1 &= \int_0^T s_w(t) \phi_1(t) dt \\ &= \int_0^T [G(t)s(t) + w(t)] \phi_1(t) dt \\ &= G(t) \int_0^T s(t) \phi_1(t) dt + \int_0^T w(t) \phi_1(t) dt \\ &= \left(G(t) \sqrt{E_s} \right) \cos(Z(t)) + w_1 \Rightarrow \\ x_1 &= A(t) \cos(Z(t)) + w_1 \end{aligned} \quad (2.22)$$

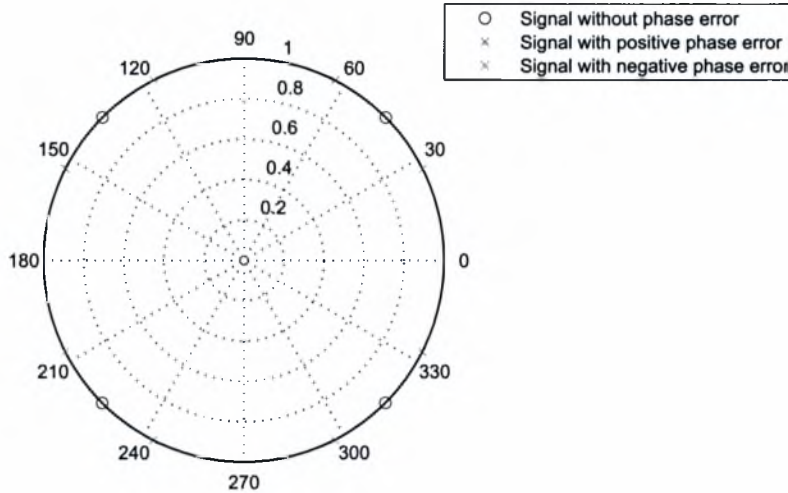
και

$$\begin{aligned} x_2 &= \int_0^T s_w(t) \phi_2(t) dt \\ &= \int_0^T [G(t)s(t) + w(t)] \phi_2(t) dt \Rightarrow \\ x_2 &= -A(t) \sin(Z(t)) + w_2 \end{aligned} \quad (2.23)$$

με $w_1 = \int_0^T w(t) \phi_1(t) dt$ και $w_2 = \int_0^T w(t) \phi_2(t) dt$.

Άρα, το διάνυσμα παρατήρησης για την περίπτωση με AWGN θόρυβο είναι

$$\mathbf{x} = \begin{bmatrix} A(t) \cos(Z(t)) + w_1 \\ -A(t) \sin(Z(t)) + w_2 \end{bmatrix} \quad (2.24)$$



Σχήμα 2.5: Επίπτωση του θορύβου φάσης στα σύμβολα QPSK – δεν εμφανίζεται η επίπτωση τυχόν AWGN

2.4 Περίπτωση με AWGN θόρυβο και θόρυβο φάσης στον πομπό

2.4.1 Θόρυβος φάσης

Ο θόρυβος φάσης, όπως προαναφέρθηκε, είναι ένα από τα κυριότερα προβλήματα στα δορυφορικά συστήματα. Δημιουργείται λόγω των κατασκευαστικών ατελειών των συστημάτων διαμόρφωσης/αποδιαμόρφωσης και των ανοχών των αναλογικών κυκλωμάτων και ειδικότερα των ταλαντωτών, που αναλαμβάνουν την μεταφορά του σήματος από τη ζώνη βάσης και το αντίθετο.

Υπερβολικός θόρυβος φάσης, σε συνδυασμό με τον πανταχού παρόντα θόρυβο του καναλιού, ισχυρό ή μη, έχει σαν συνέπεια την καταστροφή της πληροφορίας, αφού μπορεί να μετατοπίσει αρκετά την φάση ώστε να αλλάξει η πληροφορία του συμβόλου. Τα αποτελέσματα του θορύβου φάσης οπτικοποιούνται στο Σχήμα(2.5). Είναι εμφανές πως αν ο θόρυβος φάσης ξεπεράσει τα $\pm \frac{\pi}{4}$, ο αποκωδικοποιητής τελικά αναγνωρίζει άλλο σύμβολο από αυτό που στάλθηκε.

Έκφραση θορύβου φάσης

Έστω $\Psi(t)$ ο θόρυβος φάσης. Σύμφωνα με το [1], θα μπορούσε να μοντελοποιηθεί σαν ένα ημίτονο συχνότητας f_m το οποίο μεταβάλλεται πολύ αργά στο χρόνο και σε σχέση με τη φέρουσα, δηλαδή $f_m \ll f_c$. Η αντίστοιχη έκφραση είναι η

$$\Psi(t) = \frac{\Delta f}{f_m} \sin(2\pi f_m t) \quad (2.25)$$

Επιπλέον, αν υποθεθεί ότι $|\Psi(t)| \ll 1$, τότε

$$\cos(\Psi(t)) \approx 1 \quad (2.26)$$

$$\sin(\Psi(t)) \approx \Psi(t) \quad (2.27)$$

Άθροισμα 2 θορύβων φάσης

Έστω 2 εκφράσεις $\Psi_1(t)$ και $\Psi_2(t)$ που περιγράφουν θόρυβο φάσης με τύπους

$$\Psi_1(t) = \frac{\Delta f_1}{f_{m_1}} \sin(2\pi f_{m_1} t)$$

$$\Psi_2(t) = \frac{\Delta f_2}{f_{m_2}} \sin(2\pi f_{m_2} t)$$

Έχοντας υπόψη ότι

$$\sin(x) \approx x \text{ όταν } |x| \ll 1$$

τότε το άθροισμά τους ισούται με

$$\begin{aligned} \Psi_1(t) + \Psi_2(t) &= \frac{\Delta f_1}{f_{m_1}} \sin(2\pi f_{m_1} t) + \frac{\Delta f_2}{f_{m_2}} \sin(2\pi f_{m_2} t) \\ &= \frac{\Delta f_1}{f_{m_1}} (2\pi f_{m_1} t) + \frac{\Delta f_2}{f_{m_2}} (2\pi f_{m_2} t) \\ &= 2\pi t (\Delta f_1 + \Delta f_2) \\ &= \frac{f_{m_1} + f_{m_2}}{f_{m_1} + f_{m_2}} 2\pi t (\Delta f_1 + \Delta f_2) \\ &= \frac{\Delta f_1 + \Delta f_2}{f_{m_1} + f_{m_2}} \sin[2\pi(f_{m_1} + f_{m_2})t] \end{aligned}$$

οπότε

$$\Psi_s(t) = \Psi_1(t) + \Psi_2(t) \quad (2.28)$$

Συνεπώς, το άθροισμα δύο εκφράσεων θορύβου φάσης είναι ουσιαστικά πάλι μία έκφραση θορύβου φάσης.

Γινόμενο 2 θορύβων φάσης

Έστω εκφράσεις $\Psi_1(t)$ και $\Psi_2(t)$ που περιγράφουν 2 θορύβους φάσης. Το γινόμενό τους ισούται με

$$\begin{aligned} \Psi_1(t) \cdot \Psi_2(t) &= \frac{\Delta f_1}{f_{m_1}} \sin(2\pi f_{m_1} t) \cdot \frac{\Delta f_2}{f_{m_2}} \sin(2\pi f_{m_2} t) \\ &= \frac{\Delta f_1}{f_{m_1}} \frac{\Delta f_2}{f_{m_2}} \cdot \sin(2\pi f_{m_1} t) \sin(2\pi f_{m_2} t) \\ &\approx 0 \end{aligned}$$

οπότε

$$\Psi_1(t) \cdot \Psi_2(t) \approx 0 \quad (2.29)$$

Ουσιαστικά, το γινόμενο δύο μικρών θορύβων είναι αμελητέο.

2.4.2 Σήμα με θόρυβο φάσης

Έστω $\Psi(t)$ ο θόρυβος φάσης στο σήμα λόγω των ατελειών των συστημάτων διαμόρφωσης του πομπού. Η (2.5) γίνεται

$$s_n(t) = \sqrt{\frac{2E_s}{T}} \cos(Z(t)) \cos[2\pi f_c t + \Psi(t)] - \sqrt{\frac{2E_s}{T}} \sin(Z(t)) \sin[2\pi f_c t + \Psi(t)] \quad (2.30)$$

2.4.3 Λαμβανόμενο σήμα

Συνδυαζόμενες οι (2.30) και (2.20) δίνουν την εξίσωση του λαμβανόμενου σήματος $x(t)$, που περιέχει το θόρυβο του καναλιού αλλά και τον θόρυβο φάσης που επιβάλλεται από τα στοιχεία του πομπού.

Η εξίσωση που περιγράφει το λαμβανόμενο σήμα είναι

$$\begin{aligned} x(t) &= G(t)s_n(t) + w(t) \\ &= G(t) \left[\sqrt{\frac{2E_s}{T}} \cos(Z(t)) \cos[2\pi f_c t + \Psi(t)] - \sqrt{\frac{2E_s}{T}} \sin(Z(t)) \sin[2\pi f_c t + \Psi(t)] \right] \Rightarrow \\ x(t) &= A(t) \sqrt{\frac{2}{T}} \cos(Z(t)) \cos[2\pi f_c t + \Psi(t)] \\ &\quad - A(t) \sqrt{\frac{2}{T}} \sin(Z(t)) \sin[2\pi f_c t + \Psi(t)] + w(t) \end{aligned} \quad (2.31)$$

2.4.4 Διάνυσμα παρατήρησης σήματος με θόρυβο φάσης στον πομπό, AWGN θόρυβο και απώλειες

Έστω ότι το λαμβανόμενο σήμα έχει θόρυβο φάσης που προκαλείται από τα στοιχεία του πομπού, AWGN θόρυβο και απώλειες, δηλαδή είναι το σήμα $x(t)$ που παρουσιάστηκε στην προηγούμενη ενότητα. Στο στάδιο της αποδιαμόρφωσης δεν υπάρχει επιπλέον θόρυβος φάσης από τον δέκτη.

Η παρατηρούμενη τιμή x_1 , ομοίως με πριν, είναι

$$\begin{aligned} x_1 &= \int_0^T x(t) \phi_1(t) dt \\ &= \int_0^T [G(t)s_n(t) + w(t)] \phi_1(t) dt \\ &= G(t) \int_0^T s_n(t) \phi_1(t) dt + \int_0^T w(t) \phi_1(t) dt \\ &= G(t) \int_0^T \left[\sqrt{\frac{2E_s}{T}} \cos(Z(t)) \cos[2\pi f_c t + \Psi(t)] - \sqrt{\frac{2E_s}{T}} \sin(Z(t)) \sin[2\pi f_c t + \Psi(t)] \right] \\ &\quad \cdot \sqrt{\frac{2}{T}} \cos(2\pi f_c t) dt + w_1 \\ &\stackrel{(2.21)}{=} A(t) \cos(Z(t)) \int_0^T \frac{2}{T} \cos[2\pi f_c t + \Psi(t)] \cos(2\pi f_c t) dt \\ &\quad - A(t) \sin(Z(t)) \int_0^T \frac{2}{T} \sin[2\pi f_c t + \Psi(t)] \cos(2\pi f_c t) dt + w_1 \Rightarrow \end{aligned}$$

$$x_1 = A(t) \cos(Z(t)) \cdot \alpha_1 - A(t) \sin(Z(t)) \cdot \beta_1 + w_1 \quad (2.32)$$

Ο παράγοντας α_1 είναι

$$\begin{aligned} \alpha_1 &= \int_0^T \frac{2}{T} \cos[2\pi f_c t + \Psi(t)] \cos(2\pi f_c t) dt \\ &= \int_0^T \frac{2}{T} [\cos(2\pi f_c t) \cos(\Psi(t)) - \sin(2\pi f_c t) \sin(\Psi(t))] \cos(2\pi f_c t) dt \\ &= \int_0^T \frac{2}{T} [\cos(2\pi f_c t) \cdot 1 - \sin(2\pi f_c t) \cdot \Psi(t)] \cos(2\pi f_c t) dt \\ &= \int_0^T \frac{2}{T} \cos^2(2\pi f_c t) dt - \int_0^T \frac{2}{T} \Psi(t) \cos(2\pi f_c t) \sin(2\pi f_c t) dt \\ &= 1 - \int_0^T \frac{2}{T} \Psi(t) \cos(2\pi f_c t) \sin(2\pi f_c t) dt \end{aligned}$$

εφόσον υπάρχουν φίλτρα που απορρίπτουν όρους που βρίσκονται στο $2f_c$, σύμφωνα με τις (2.11), (2.12) και (2.13), ενώ ο β_1

$$\begin{aligned} \beta_1 &= \int_0^T \frac{2}{T} \sin[2\pi f_c t + \Psi(t)] \cos(2\pi f_c t) dt \\ &= \int_0^T \frac{2}{T} [\sin(2\pi f_c t) \cos(\Psi(t)) + \cos(2\pi f_c t) \sin(\Psi(t))] \cos(2\pi f_c t) dt \\ &= \int_0^T \frac{2}{T} [\sin(2\pi f_c t) \cdot 1 + \cos(2\pi f_c t) \cdot \Psi(t)] \cos(2\pi f_c t) dt \\ &= \int_0^T \frac{2}{T} \sin(2\pi f_c t) \cos(2\pi f_c t) dt + \int_0^T \frac{2}{T} \Psi(t) \cos^2(2\pi f_c t) dt \\ &= 0 + \int_0^T \frac{2}{T} \Psi(t) \left[\frac{1}{2} + \frac{1}{2} \cos(2 \cdot 2\pi f_c t) \right] dt \\ &= \int_0^T \frac{1}{2T} \Psi(t) dt + \int_0^T \frac{1}{2T} \Psi(t) \cos(2 \cdot 2\pi f_c t) dt \\ &= \int_0^T \frac{1}{T} \Psi(t) dt \end{aligned}$$

Έτσι, η (2.32) γίνεται

$$\begin{aligned} x_1 &= A(t) \cos(Z(t)) \left[1 - \int_0^T \frac{2}{T} \Psi(t) \cos(2\pi f_c t) \sin(2\pi f_c t) dt \right] \\ &\quad - A(t) \sin(Z(t)) \left[\int_0^T \frac{1}{T} \Psi(t) dt \right] + w_1 \end{aligned} \quad (2.33)$$

Αν επιπλέον θεωρηθεί ότι η $\Psi(t)$ μεταβάλλεται ιδιαίτερα αργά σε σχέση με την περίοδο T , τότε

$$\begin{aligned} x_1 &= A(t) \cos(Z(t)) \left[1 - \Psi(t) \int_0^T \frac{2}{T} \cos(2\pi f_c t) \sin(2\pi f_c t) dt \right] \\ &\quad - A(t) \sin(Z(t)) \left[\Psi(t) \int_0^T \frac{1}{T} \cdot 1 dt \right] + w_1 \\ &= A(t) \cos(Z(t)) [1 - \Psi(t) \cdot 0] - A(t) \sin(Z(t)) \Psi(t) + w_1 \Rightarrow \end{aligned}$$

$$x_1 = A(t) \cos(Z(t)) - A(t) \sin(Z(t)) \Psi(t) + w_1 \quad (2.34)$$

Ομοίως,

$$\begin{aligned} x_2 &= \int_0^T x(t) \phi_2(t) dt \\ &= \int_0^T [G(t) s_n(t) + w(t)] \phi_2(t) dt \\ &= G(t) \int_0^T s_n(t) \phi_2(t) dt + \int_0^T w(t) \phi_2(t) dt \\ &= G(t) \int_0^T \left[\sqrt{\frac{2E_s}{T}} \cos(Z(t)) \cos[2\pi f_c t + \Psi(t)] - \sqrt{\frac{2E_s}{T}} \sin(Z(t)) \sin[2\pi f_c t + \Psi(t)] \right] \\ &\quad \cdot \sqrt{\frac{2}{T}} \sin(2\pi f_c t) dt + w_2 \\ &\stackrel{(2.21)}{=} A(t) \cos(Z(t)) \int_0^T \frac{2}{T} \cos[2\pi f_c t + \Psi(t)] \sin(2\pi f_c t) dt \\ &\quad - A(t) \sin(Z(t)) \int_0^T \frac{2}{T} \sin[2\pi f_c t + \Psi(t)] \sin(2\pi f_c t) dt + w_2 \Rightarrow \\ x_2 &= A(t) \cos(Z(t)) \cdot \alpha_2 - A(t) \sin(Z(t)) \cdot \beta_2 + w_2 \end{aligned} \quad (2.35)$$

Ο παράγοντας α_2 είναι

$$\begin{aligned} \alpha_2 &= \int_0^T \frac{2}{T} \cos[2\pi f_c t + \Psi(t)] \sin(2\pi f_c t) dt \\ &= \int_0^T \frac{2}{T} [\cos(2\pi f_c t) \cos(\Psi(t)) - \sin(2\pi f_c t) \sin(\Psi(t))] \sin(2\pi f_c t) dt \\ &= \int_0^T \frac{2}{T} [\cos(2\pi f_c t) \cdot 1 - \sin(2\pi f_c t) \cdot \Psi(t)] \sin(2\pi f_c t) dt \\ &= \int_0^T \frac{2}{T} \cos(2\pi f_c t) \sin(2\pi f_c t) dt - \int_0^T \frac{2}{T} \Psi(t) \sin^2(2\pi f_c t) dt \\ &= 0 - \int_0^T \frac{2}{T} \Psi(t) \left[\frac{1}{2} - \frac{1}{2} \cos(2 \cdot 2\pi f_c t) \right] dt \\ &= - \int_0^T \frac{1}{2T} \Psi(t) dt + \int_0^T \frac{1}{2T} \Psi(t) \cos(2 \cdot 2\pi f_c t) dt \\ &= - \int_0^T \frac{1}{2T} \Psi(t) dt \end{aligned}$$

ενώ ο β_2

$$\begin{aligned}
 \beta_2 &= \int_0^T \frac{2}{T} \sin [2\pi f_c t + \Psi(t)] \sin(2\pi f_c t) dt \\
 &= \int_0^T \frac{2}{T} [\sin(2\pi f_c t) \cos(\Psi(t)) + \cos(2\pi f_c t) \sin(\Psi(t))] \sin(2\pi f_c t) dt \\
 &= \int_0^T \frac{2}{T} [\sin(2\pi f_c t) \cdot 1 + \cos(2\pi f_c t) \cdot \Psi(t)] \sin(2\pi f_c t) dt \\
 &= \int_0^T \frac{2}{T} \sin^2(2\pi f_c t) dt + \int_0^T \frac{2}{T} \Psi(t) \cos(2\pi f_c t) \sin(2\pi f_c t) dt \\
 &= 1 + \int_0^T \frac{2}{T} \Psi(t) \cos(2\pi f_c t) \sin(2\pi f_c t) dt
 \end{aligned}$$

Έτσι, η (2.35) γίνεται

$$\begin{aligned}
 x_2 &= A(t) \cos(Z(t)) \left[-\int_0^T \frac{1}{T} \Psi(t) dt \right] \\
 &\quad - A(t) \sin(Z(t)) \left[1 + \int_0^T \frac{2}{T} \Psi(t) \cos(2\pi f_c t) \sin(2\pi f_c t) dt \right] + w_2
 \end{aligned} \tag{2.36}$$

Και αντίστοιχα με την θεώρηση που οδήγησε στην (2.34)

$$\begin{aligned}
 x_2 &= A(t) \cos(Z(t)) \left[-\Psi(t) \int_0^T \frac{1}{T} dt \right] \\
 &\quad - A(t) \sin(Z(t)) \left[1 + \Psi(t) \int_0^T \frac{2}{T} \Psi(t) \cos(2\pi f_c t) \sin(2\pi f_c t) dt \right] \\
 &\quad + w_2 \Rightarrow \\
 x_2 &= -A(t) \cos(Z(t)) \Psi(t) - A(t) \sin(Z(t)) + w_2
 \end{aligned} \tag{2.37}$$

Το διάνυσμα παρατήρησης για την περίπτωση με AWGN θόρυβο και θόρυβο φάσης στον πομπό είναι

$$\mathbf{x} = \begin{bmatrix} A(t) \cos(Z(t)) - A(t) \sin(Z(t)) \Psi(t) + w_1 \\ -A(t) \cos(Z(t)) \Psi(t) - A(t) \sin(Z(t)) + w_2 \end{bmatrix} \tag{2.38}$$

2.5 Περίπτωση με AWGN θόρυβο και θόρυβο φάσης σε πομπό και δέκτη

2.5.1 Συναρτήσεις ορθοκανονικής βάσης με θόρυβο φάσης

Αν υποθεθεί ότι υπάρχει και θόρυβος φάσης λόγω των στοιχείων του δέκτη, το πρόβλημα θα μπορούσε να μοντελοποιηθεί προσθέτοντας κάποια φάση $\Psi_\phi(t)$ στην συχνότητα που χρησιμοποιείται στις συναρτήσεις βάσης για την αποδιαμόρφωση. Οι αντίστοιχες συναρτήσεις βάσης γίνονται

$$\phi_{n_1}(t) = \sqrt{\frac{2}{T}} \cos [2\pi f_c t + \Psi_\phi(t)] \tag{2.39}$$

$$\phi_{n_2}(t) = \sqrt{\frac{2}{T}} \sin [2\pi f_c t + \Psi_\phi(t)] \tag{2.40}$$

2.5.2 Διάνυσμα παρατήρησης σήματος με AWGN θόρυβο και θόρυβο φάσης σε πομπό και δέκτη

Έστω ότι το λαμβανόμενο σήμα έχει θόρυβο φάσης, AWGN θόρυβο και απώλειες. Πλέον και στο στάδιο της αποδιαμόρφωσης υπάρχει θόρυβος φάσης.

Η τιμή x_1 είναι

$$\begin{aligned}
 x_1 &= \int_0^T x(t) \phi_{n_1}(t) dt \\
 &\stackrel{(2.21)}{=} A(t) \cos(Z(t)) \int_0^T \frac{2}{T} \cos[2\pi f_c t + \Psi(t)] \cos[2\pi f_c t + \Psi_\phi(t)] dt \\
 &\quad - A(t) \sin(Z(t)) \int_0^T \frac{2}{T} \sin[2\pi f_c t + \Psi(t)] \cos[2\pi f_c t + \Psi_\phi(t)] dt + w_1 \Rightarrow \\
 x_1 &= A(t) \cos(Z(t)) \cdot a_1 - A(t) \sin(Z(t)) \cdot b_1 + w_1 \tag{2.41}
 \end{aligned}$$

Ο παράγοντας a_1 είναι

$$\begin{aligned}
 a_1 &= \int_0^T \frac{2}{T} \cos[2\pi f_c t + \Psi(t)] \cos[2\pi f_c t + \Psi_\phi(t)] dt \\
 &= \int_0^T \frac{2}{T} [\cos(2\pi f_c t) \cos(\Psi(t)) - \sin(2\pi f_c t) \sin(\Psi(t))] \\
 &\quad \cdot [\cos(2\pi f_c t) \cos(\Psi_\phi(t)) - \sin(2\pi f_c t) \sin(\Psi_\phi(t))] dt \\
 &= \int_0^T \frac{2}{T} [\cos(2\pi f_c t) \cdot 1 - \sin(2\pi f_c t) \cdot \Psi(t)] [\cos(2\pi f_c t) \cdot 1 - \sin(2\pi f_c t) \cdot \Psi_\phi(t)] dt \\
 &= \int_0^T \frac{2}{T} \cos^2(2\pi f_c t) dt - \int_0^T \frac{2}{T} \Psi_\phi(t) \cos(2\pi f_c t) \sin(2\pi f_c t) dt \\
 &\quad - \int_0^T \frac{2}{T} \Psi(t) \cos(2\pi f_c t) \sin(2\pi f_c t) dt + \int_0^T \frac{2}{T} \Psi(t) \Psi_\phi(t) \sin^2(2\pi f_c t) dt \\
 &= 1 - \int_0^T \frac{2}{T} (\Psi_\phi(t) + \Psi(t)) \cos(2\pi f_c t) \sin(2\pi f_c t) dt \\
 &\quad + \int_0^T \frac{2}{T} \Psi(t) \Psi_\phi(t) \left(\frac{1}{2} - \frac{1}{2} \cos(2 \cdot 2\pi f_c t) \right) dt \\
 &= 1 - \int_0^T \frac{2}{T} (\Psi_\phi(t) + \Psi(t)) \cos(2\pi f_c t) \sin(2\pi f_c t) dt \\
 &\quad + \int_0^T \frac{1}{2} \frac{2}{T} \Psi(t) \Psi_\phi(t) dt - \int_0^T \frac{2}{T} \Psi(t) \Psi_\phi(t) \frac{1}{2} \cos(2 \cdot 2\pi f_c t) dt \\
 &= 1 - \int_0^T \frac{2}{T} (\Psi_\phi(t) + \Psi(t)) \cos(2\pi f_c t) \sin(2\pi f_c t) dt + \int_0^T \frac{1}{T} \Psi(t) \Psi_\phi(t) dt
 \end{aligned}$$

ενώ ο b_1

$$\begin{aligned}
b_1 &= \int_0^T \frac{2}{T} \sin[2\pi f_c t + \Psi(t)] \cos[2\pi f_c t + \Psi_\phi(t)] dt \\
&= \int_0^T \frac{2}{T} [\sin(2\pi f_c t) \cos(\Psi(t)) + \cos(2\pi f_c t) \sin(\Psi(t))] \\
&\quad \cdot [\cos(2\pi f_c t) \cos(\Psi_\phi(t)) - \sin(2\pi f_c t) \sin(\Psi_\phi(t))] dt \\
&= \int_0^T \frac{2}{T} [\sin(2\pi f_c t) \cdot 1 + \cos(2\pi f_c t) \cdot \Psi(t)] [\cos(2\pi f_c t) \cdot 1 - \sin(2\pi f_c t) \cdot \Psi_\phi(t)] dt \\
&= \int_0^T \frac{2}{T} \sin(2\pi f_c t) \cos(2\pi f_c t) dt - \int_0^T \frac{2}{T} \Psi_\phi(t) \sin^2(2\pi f_c t) dt \\
&\quad + \int_0^T \frac{2}{T} \Psi(t) \cos^2(2\pi f_c t) dt - \int_0^T \frac{2}{T} \Psi(t) \Psi_\phi(t) \cos(2\pi f_c t) \sin(2\pi f_c t) dt \\
&= 0 - \int_0^T \frac{2}{T} \Psi_\phi(t) \left(\frac{1}{2} - \frac{1}{2} \cos(2 \cdot 2\pi f_c t) \right) dt \\
&\quad + \int_0^T \frac{2}{T} \Psi(t) \left(\frac{1}{2} + \frac{1}{2} \cos(2 \cdot 2\pi f_c t) \right) dt - \int_0^T \frac{2}{T} \Psi(t) \Psi_\phi(t) \cos(2\pi f_c t) \sin(2\pi f_c t) dt \\
&= - \int_0^T \frac{1}{2T} \Psi_\phi(t) dt + \int_0^T \frac{1}{2T} \Psi_\phi(t) \cos(2 \cdot 2\pi f_c t) dt \\
&\quad + \int_0^T \frac{1}{2T} \Psi(t) dt + \int_0^T \frac{1}{2T} \Psi(t) \cos(2 \cdot 2\pi f_c t) dt \\
&\quad - \int_0^T \frac{2}{T} \Psi(t) \Psi_\phi(t) \cos(2\pi f_c t) \sin(2\pi f_c t) dt \\
&= \int_0^T \frac{1}{T} (\Psi(t) - \Psi_\phi(t)) dt - \int_0^T \frac{2}{T} \Psi(t) \Psi_\phi(t) \cos(2\pi f_c t) \sin(2\pi f_c t) dt
\end{aligned}$$

Αν

$$\Psi(t) + \Psi_\phi(t) = \Psi_{\text{sum}}(t) \quad (2.42)$$

$$\Psi(t) - \Psi_\phi(t) = \Psi_{\text{diff}}(t) \quad (2.43)$$

$$\Psi(t) \Psi_\phi(t) = \Psi_{\text{prod}}(t) \quad (2.44)$$

η (2.41) γίνεται

$$\begin{aligned}
x_1 &= A(t) \cos(Z(t)) \left[1 - \int_0^T \frac{2}{T} \Psi_{\text{sum}}(t) \cos(2\pi f_c t) \sin(2\pi f_c t) dt + \int_0^T \frac{1}{T} \Psi_{\text{prod}}(t) dt \right] \\
&\quad - A(t) \sin(Z(t)) \left[\int_0^T \frac{1}{T} \Psi_{\text{diff}}(t) dt - \int_0^T \frac{2}{T} \Psi_{\text{prod}}(t) \cos(2\pi f_c t) \sin(2\pi f_c t) dt \right] \\
&\quad + w_1
\end{aligned} \quad (2.45)$$

Χρησιμοποιώντας την ίδια λογική από την οποία προέκυψε η (2.34) και τις σχέσεις (2.28) και (2.29) τότε

$$x_1 = A(t) [\cos(Z(t)) (1 - 0 + 0) - \sin(Z(t)) \Psi_{\text{diff}}(t)] + w_1 \Rightarrow$$

$$x_1 = A(t) [\cos(Z(t)) - \sin(Z(t)) (\Psi(t) - \Psi_\phi(t))] + w_1 \quad (2.46)$$

που είναι μια εξίσωση παρόμοια με την (2.34).

Ομοίως, η τιμή x_2 είναι

$$\begin{aligned} x_2 &= \int_0^T x(t)\phi_{n_2}(t) dt \\ &\stackrel{(2.21)}{=} A(t) \cos(Z(t)) \int_0^T \frac{2}{T} \cos[2\pi f_c t + \Psi(t)] \sin[2\pi f_c t + \Psi_\phi(t)] dt \\ &\quad - A(t) \sin(Z(t)) \int_0^T \frac{2}{T} \sin[2\pi f_c t + \Psi(t)] \sin[2\pi f_c t + \Psi_\phi(t)] dt + w_2 \end{aligned}$$

$$x_2 = A(t) \cos(Z(t)) \cdot a_2 - A(t) \sin(Z(t)) \cdot b_2 + w_2 \quad (2.47)$$

Ο παράγοντας a_2 είναι

$$\begin{aligned} a_2 &= \int_0^T \frac{2}{T} \cos[2\pi f_c t + \Psi(t)] \sin[2\pi f_c t + \Psi_\phi(t)] dt \\ &= \int_0^T \frac{2}{T} [\cos(2\pi f_c t) \cos(\Psi(t)) - \sin(2\pi f_c t) \sin(\Psi(t))] \\ &\quad \cdot [\sin(2\pi f_c t) \cos(\Psi_\phi(t)) + \cos(2\pi f_c t) \sin(\Psi_\phi(t))] dt \\ &= \int_0^T \frac{2}{T} [\cos(2\pi f_c t) \cdot 1 - \sin(2\pi f_c t) \cdot \Psi(t)] [\sin(2\pi f_c t) \cdot 1 + \cos(2\pi f_c t) \cdot \Psi_\phi(t)] dt \\ &= \int_0^T \frac{2}{T} \cos(2\pi f_c t) \sin(2\pi f_c t) dt + \int_0^T \frac{2}{T} \Psi_\phi(t) \cos^2(2\pi f_c t) dt \\ &\quad - \int_0^T \frac{2}{T} \Psi(t) \sin^2(2\pi f_c t) dt + \int_0^T \frac{2}{T} \Psi(t) \Psi_\phi(t) \cos(2\pi f_c t) \sin(2\pi f_c t) dt \\ &= 0 + \int_0^T \frac{2}{T} \Psi_\phi(t) \left(\frac{1}{2} + \frac{1}{2} \cos(2 \cdot 2\pi f_c t) \right) dt \\ &\quad - \int_0^T \frac{2}{T} \Psi(t) \left(\frac{1}{2} - \frac{1}{2} \cos(2 \cdot 2\pi f_c t) \right) dt + \int_0^T \frac{2}{T} \Psi(t) \Psi_\phi(t) \cos(2\pi f_c t) \sin(2\pi f_c t) dt \\ &= \int_0^T \frac{1}{T} \Psi_\phi(t) dt - \int_0^T \frac{1}{T} \Psi(t) dt + \int_0^T \frac{2}{T} \Psi(t) \Psi_\phi(t) \cos(2\pi f_c t) \sin(2\pi f_c t) dt \\ &= - \int_0^T \frac{1}{T} (\Psi(t) - \Psi_\phi(t)) dt + \int_0^T \frac{2}{T} \Psi(t) \Psi_\phi(t) \cos(2\pi f_c t) \sin(2\pi f_c t) dt \end{aligned}$$

και ο b_2

$$\begin{aligned}
b_2 &= \int_0^T \frac{2}{T} \sin [2\pi f_c t + \Psi(t)] \sin [2\pi f_c t + \Psi_\phi(t)] dt \\
&= \int_0^T \frac{2}{T} [\sin(2\pi f_c t) \cos(\Psi(t)) + \cos(2\pi f_c t) \sin(\Psi(t))] \\
&\quad \cdot [\sin(2\pi f_c t) \cos(\Psi_\phi(t)) + \cos(2\pi f_c t) \sin(\Psi_\phi(t))] dt \\
&= \int_0^T \frac{2}{T} [\sin(2\pi f_c t) \cdot 1 + \cos(2\pi f_c t) \cdot \Psi(t)] [\sin(2\pi f_c t) \cdot 1 + \cos(2\pi f_c t) \cdot \Psi_\phi(t)] dt \\
&= \int_0^T \frac{2}{T} \sin^2(2\pi f_c t) dt + \int_0^T \frac{2}{T} \Psi_\phi(t) \cos(2\pi f_c t) \sin(2\pi f_c t) dt \\
&\quad + \int_0^T \frac{2}{T} \Psi(t) \cos(2\pi f_c t) \sin(2\pi f_c t) dt + \int_0^T \frac{2}{T} \Psi(t) \Psi_\phi(t) \cos^2(2\pi f_c t) dt \\
&= 1 + \int_0^T \frac{2}{T} (\Psi(t) + \Psi_\phi(t)) \cos(2\pi f_c t) \sin(2\pi f_c t) dt \\
&\quad + \int_0^T \frac{2}{T} \Psi(t) \Psi_\phi(t) \left(\frac{1}{2} + \frac{1}{2} \cos(2 \cdot 2\pi f_c t) \right) dt \\
&= 1 + \int_0^T \frac{2}{T} (\Psi(t) + \Psi_\phi(t)) \cos(2\pi f_c t) \sin(2\pi f_c t) dt + \int_0^T \frac{1}{T} \Psi(t) \Psi_\phi(t) dt
\end{aligned}$$

Έτσι, η (2.47) γίνεται

$$\begin{aligned}
x_2 &= A(t) \cos(Z(t)) \left[- \int_0^T \frac{1}{T} \Psi_{\text{diff}}(t) dt + \int_0^T \frac{2}{T} \Psi_{\text{prod}}(t) \cos(2\pi f_c t) \sin(2\pi f_c t) dt \right] \\
&\quad - A(t) \sin(Z(t)) \left[1 + \int_0^T \frac{2}{T} \Psi_{\text{sum}}(t) \cos(2\pi f_c t) \sin(2\pi f_c t) dt + \int_0^T \frac{1}{T} \Psi_{\text{prod}}(t) dt \right] \\
&\quad + w_2
\end{aligned} \tag{2.48}$$

Όπως και στην (2.46)

$$\begin{aligned}
x_2 &= A(t) [-\cos(Z(t)) \Psi_{\text{diff}}(t) - \sin(Z(t)) (1 + 0 + 0)] + w_2 \Rightarrow \\
x_2 &= A(t) [-\cos(Z(t)) (\Psi(t) - \Psi_\phi(t)) - \sin(Z(t))] + w_2
\end{aligned} \tag{2.49}$$

που είναι μια εξίσωση παρόμοια με την (2.37).

Το διάνυσμα παρατήρησης για την περίπτωση με AWGN θόρυβο και θόρυβο φάσης σε πομπό και δέκτη είναι

$$\mathbf{x} = \begin{bmatrix} A(t) [\cos(Z(t)) - \sin(Z(t)) (\Psi(t) - \Psi_\phi(t))] + w_1 \\ A(t) [-\cos(Z(t)) (\Psi(t) - \Psi_\phi(t)) - \sin(Z(t))] + w_2 \end{bmatrix} \tag{2.50}$$

Από τα παραπάνω αποδεικνύεται πως αν

$$\Psi_\phi(t) = \Psi(t) \tag{2.51}$$

τότε διορθώνεται το πρόβλημα που προκαλείται από τον θόρυβο φάσης και εξαλείφονται οι όροι που υποβαθμίζουν το σήμα. Θα μπορούσε κανείς να πει πως όλες οι μέθοδοι διόρθωσης φάσης προσπαθούν να προσεγγίσουν την εξίσωση

$$|\Psi_\phi(t) - \Psi(t)| = 0 \tag{2.52}$$

Αν τελικά ελαχιστοποιόταν η παραπάνω εξίσωση, το διάνυσμα παρατήρησης θα ήταν

$$\mathbf{x} = \begin{bmatrix} A(t) \cos(Z(t)) + w_1 \\ -A(t) \sin(Z(t)) + w_2 \end{bmatrix} \quad (2.53)$$

δηλαδή το ίδιο με το διάνυσμα με AWGN θόρυβο (2.24).

Κεφάλαιο 3

ΕΚΤΙΜΗΤΗΣ VITERBI & VITERBI

3.1 Μη γραμμικός εκτιμητής V&V

Μια κλασική NDA μέθοδος εκτίμησης του θορύβου φάσης σε ένα κατά mPSK διαμορφωμένο σήμα είναι ο εκτιμητής των Viterbi & Viterbi (*V&V estimator*) [2].

Σύμφωνα με την εργασία των V&V, έστω ένα κατά mPSK διαμορφωμένο σήμα $y(t)$, τέτοιο ώστε

$$y(t) = \sqrt{\frac{2E_s}{T}} \sin(2\pi(f_c + \Delta f)t + \theta_n) + w(t) \quad \text{με } \Delta f \text{ κάποιο υπόλοιπο συχνότητας} \quad (3.1)$$
$$w(t) \sim \mathcal{N}(0, \sigma_w^2) \text{ (AWGN)}$$

Η θ_n περιγράφει την φάση του σήματος

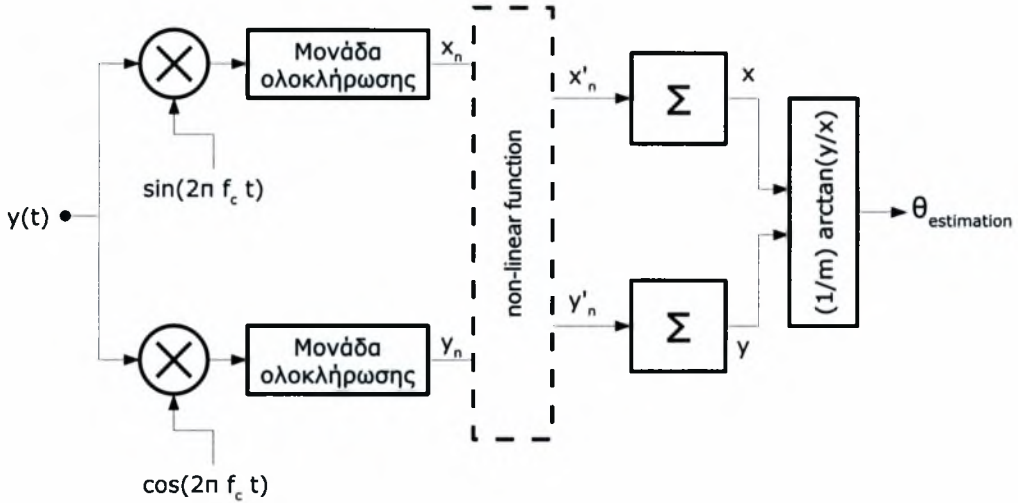
$$\theta_n = k_n \left(\frac{2\pi}{m} \right) + \theta \quad \text{με } 0 \leq k_n < m \text{ και } m \text{ μέγιστο αριθμό συμβόλων,} \quad (3.2)$$

θ θόρυβος φάσης

εκφράζοντας ουσιαστικά την πληροφορία (k_n) και τον θόρυβο φάσης (θ). Το Δf , το υπόλοιπο συχνότητας είναι αποτέλεσμα της ατελούς ανάκτησης συχνότητας της φέρουσας.

Η εκτίμηση βασίζεται στην εύρεση ενός μιγαδικού φασιθέτη ή φάσορα (phasor) για κάθε σύμβολο s_n , που ανήκει σε ένα παράθυρο μήκους $2N + 1$ συμβόλων, αφού η φάση κάθε συμβόλου πολλαπλασιαστεί με m ώστε να αφαιρεθεί η διαμόρφωση ή αλλιώς το σύμβολο να υψωθεί στην m . Έπειτα, οι φάσορες αθροίζονται και βρίσκειται η γωνία του αθροίσματός των. Τέλος, η γωνία διαιρείται με m για να αναιρεθεί ο προηγούμενος πολλαπλασιασμός με m . Έτσι προκύπτει η εκτίμηση του θορύβου φάσης του κεντρικού συμβόλου του παραθύρου, εξού και η ανάγκη για $2N + 1$ σύμβολα. Ένα απλοποιημένο διάγραμμα του εκτιμητή V&V εμφανίζεται στο Σχήμα(3.1), όπως περιγράφεται και αναλύεται στο [2].

Η εκτίμηση του θορύβου φάσης θ για κάποιο σύμβολο γίνεται χρησιμοποιώντας $2N + 1$ σύμβολα,



Σχήμα 3.1: Μη γραμμικός εκτιμητής V&V

ή αλλιώς δείγματα, και δίνεται από την εξίσωση¹

$$\hat{\theta} = \frac{1}{m} \arg \left\{ \left(\frac{1}{2N+1} \sum_{n=-N}^N x'_n \right) + j \cdot \left(\frac{1}{2N+1} \sum_{n=-N}^N y'_n \right) \right\} \quad (3.3)$$

με τα x'_n και y'_n να είναι αποτέλεσμα της

$$x'_n + j \cdot y'_n = F(\rho_n) e^{jm\phi_n} \quad (3.4)$$

με

$$\rho_n = \sqrt{x_n^2 + y_n^2} \quad (3.5)$$

$$\phi_n = \arg \{x_n + jy_n\} \quad (3.6)$$

$$F(\bullet) = (\bullet)^i, i = 0, 1, 2, \dots \quad (3.7)$$

και τα x_n και y_n είναι το αποτέλεσμα της αποδιαμόρφωσης του αρχικού σήματος $y(t)$. Είναι σκόπιμο να τονιστεί ότι το i δεν έχει καμία σχέση με το m .

Η ύψωση στην m -ιοστή είναι απαραίτητη για την αφαίρεση της διαμόρφωσης² όπως φαίνεται στο παρακάτω παράδειγμα

$$\begin{aligned} e^{jm\phi_n} &= e^{jm[k_n(\frac{2\pi}{m}) + \theta]} \\ &= e^{j[m \cdot k_n \frac{2\pi}{m} m + m \cdot \theta]} \\ &= e^{jm\theta} \end{aligned}$$

¹Ο εκτιμητής που παρουσιάζεται στο [2] έχει το μειονέκτημα ότι χρησιμοποιεί την συνάρτηση $\tan^{-1}()$, που έχει σαν πεδίο τιμών το $[-\frac{\pi}{2}, \frac{\pi}{2}]$. Εισάγει μια αβεβαιότητα ως προς τη φάση που μπορεί να λυθεί αν χρησιμοποιηθεί η $\arg()$, που έχει σαν πεδίο τιμών το $[-\pi, \pi]$. Συνεπώς, η μορφή του εκτιμητή με $\tan^{-1}()$ που παρουσιάζεται στο [2] δεν είναι απόλυτα σωστή.

²Στο Κεφάλαιο 2 για την QPSK, η φάση μπορεί να πάρει τις τιμές $+\frac{\pi}{4}, +\frac{3\pi}{4}, +\frac{5\pi}{4}, +\frac{7\pi}{4}$.

Στην (3.2) εμφανίζεται η φάση να παίρνει τιμές $0, +\frac{\pi}{2}, +\pi, +\frac{3\pi}{2}$. Αν ακολουθηθεί η λογική που διέπει το Κεφάλαιο 2 για την επιλογή τιμών φάσης, για την αφαίρεση της διαμόρφωσης θα έπρεπε να αφαιρεθεί $\frac{\pi}{4}$ από κάθε σύμβολο ή από την τελική εκτίμηση του θορόβου φάσης, αφού όλα τα σύμβολα έχουν μια μετατόπιση κατά $\frac{\pi}{4}$.

Μία γενικότερη συνάρτηση για την εκτίμηση του θορύβου φάσης ενός οποιοδήποτε συμβόλου n από έναν συρμό, βασισμένη στην (3.3), είναι η

$$\hat{\theta}_n = \frac{1}{m} \arg \left\{ \left(\frac{1}{2N+1} \sum_{l=-N}^N x'_{n+l} \right) + j \cdot \left(\frac{1}{2N+1} \sum_{l=-N}^N y'_{n+l} \right) \right\} \quad (3.8)$$

Ακολουθεί το στάδιο της αποδιαμόρφωσης των συμβόλων. Έτσι,

$$\begin{aligned} x_n &= \int_{nT-\frac{T}{2}}^{nT+\frac{T}{2}} \sqrt{\frac{2E_s}{T}} \sin(2\pi(f_c + \Delta f)t + \theta_n) \sqrt{2} \sin(2\pi f_c t) dt \\ &\quad + \int_{nT-\frac{T}{2}}^{nT+\frac{T}{2}} w(t) \sqrt{2} \sin(2\pi f_c t) dt \\ &= \int_{nT-\frac{T}{2}}^{nT+\frac{T}{2}} \sqrt{\frac{E_s}{T}} [\cos(\theta_n + 2\pi\Delta f t) - \cos(4\pi f_c t + 2\pi\Delta f t + \theta_n)] dt + w_1 \\ &= \sqrt{\frac{E_s}{T}} \int_{nT-\frac{T}{2}}^{nT+\frac{T}{2}} \cos(\theta_n + 2\pi\Delta f t) dt \\ &\quad - \sqrt{\frac{E_s}{T}} \int_{nT-\frac{T}{2}}^{nT+\frac{T}{2}} \cos(4\pi f_c t + 2\pi\Delta f t + \theta_n) dt + w_1 \\ &= \sqrt{\frac{E_s}{T}} \left[\frac{\sin(\theta_n + 2\pi\Delta f t)}{2\pi\Delta f} \right]_{nT-\frac{T}{2}}^{nT+\frac{T}{2}} \\ &\quad - \sqrt{\frac{E_s}{T}} \left[\frac{\sin(4\pi f_c t + 2\pi\Delta f t + \theta_n)}{4\pi f_c + 2\pi\Delta f} \right]_{nT-\frac{T}{2}}^{nT+\frac{T}{2}} + w_1 \\ &= \sqrt{\frac{E_s}{T}} \left[\frac{\sin(\theta_n + 2\pi\Delta f(nT + \frac{T}{2})) - \sin(\theta_n + 2\pi\Delta f(nT - \frac{T}{2}))}{2\pi\Delta f} \right] \\ &\quad - \sqrt{\frac{E_s}{T}} \left[\frac{\sin(\theta_n + 2\pi(2f_c + \Delta f)(nT + \frac{T}{2})) - \sin(\theta_n + 2\pi(2f_c + \Delta f)(nT - \frac{T}{2}))}{4\pi f_c + 2\pi\Delta f} \right] \\ &\quad + w_1 \\ &= \sqrt{\frac{E_s}{T}} \left[\frac{2 \cos(\theta_n + 2\pi\Delta f nT) \sin(\pi\Delta f T)}{2\pi\Delta f} \right] \\ &\quad - \sqrt{\frac{E_s}{T}} \left[\frac{2 \cos(\theta_n + 2\pi(2f_c + \Delta f)nT) \sin(\pi(2f_c + \Delta f)T)}{4\pi f_c + 2\pi\Delta f} \right] + w_1 \\ &= \sqrt{\frac{E_s}{T}} \left[\frac{\cos(\theta_n + 2\pi\Delta f nT) \sin(\pi\Delta f T) T}{\pi\Delta f T} \right] \\ &\quad - \sqrt{\frac{E_s}{T}} \left[\frac{\cos(\theta_n + 2\pi(2f_c + \Delta f)nT) \sin(\pi(2f_c + \Delta f)T) T}{(2\pi f_c + \pi\Delta f) T} \right] + w_1 \Rightarrow \\ x_n &= \sqrt{E_s T} \cos(\theta_n + 2\pi\Delta f nT) \text{sinc}(\Delta f T) \\ &\quad - \sqrt{E_s T} \cos(\theta_n + 2\pi(2f_c + \Delta f)nT) \text{sinc}((2f_c + \Delta f)T) + w_1 \end{aligned} \quad (3.9)$$

Ομοίως,

$$\begin{aligned}
y_n &= \int_{nT-\frac{T}{2}}^{nT+\frac{T}{2}} \sqrt{\frac{2E_s}{T}} \sin(2\pi(f_c + \Delta f)t + \theta_n) \sqrt{2} \cos(2\pi f_c t) dt \\
&\quad + \int_{nT-\frac{T}{2}}^{nT+\frac{T}{2}} w(t) \sqrt{2} \cos(2\pi f_c t) dt \\
&= \int_{nT-\frac{T}{2}}^{nT+\frac{T}{2}} \sqrt{\frac{E_s}{T}} [\sin(\theta_n + 2\pi\Delta f t) + \sin(4\pi f_c t + 2\pi\Delta f t + \theta_n)] dt + w_2 \\
&= \sqrt{\frac{E_s}{T}} \int_{nT-\frac{T}{2}}^{nT+\frac{T}{2}} \sin(\theta_n + 2\pi\Delta f t) dt \\
&\quad - \sqrt{\frac{E_s}{T}} \int_{nT-\frac{T}{2}}^{nT+\frac{T}{2}} \sin(4\pi f_c t + 2\pi\Delta f t + \theta_n) dt + w_2 \\
&= \sqrt{\frac{E_s}{T}} \left[\frac{-\cos(\theta_n + 2\pi\Delta f t)}{2\pi\Delta f} \right]_{nT-\frac{T}{2}}^{nT+\frac{T}{2}} \\
&\quad - \sqrt{\frac{E_s}{T}} \left[\frac{-\cos(4\pi f_c t + 2\pi\Delta f t + \theta_n)}{4\pi f_c + 2\pi\Delta f} \right]_{nT-\frac{T}{2}}^{nT+\frac{T}{2}} + w_2 \\
&= \sqrt{\frac{E_s}{T}} \left[\frac{-\cos(\theta_n + 2\pi\Delta f(nT + \frac{T}{2})) + \cos(\theta_n + 2\pi\Delta f(nT - \frac{T}{2}))}{2\pi\Delta f} \right] \\
&\quad - \sqrt{\frac{E_s}{T}} \left[\frac{-\cos(\theta_n + 2\pi(2f_c + \Delta f)(nT + \frac{T}{2})) + \cos(\theta_n + 2\pi(2f_c + \Delta f)(nT - \frac{T}{2}))}{4\pi f_c + 2\pi\Delta f} \right] \\
&\quad + w_2 \\
&= \sqrt{\frac{E_s}{T}} \left[\frac{2\sin(\theta_n + 2\pi\Delta f nT) \sin(\pi\Delta f T)}{2\pi\Delta f} \right] \\
&\quad - \sqrt{\frac{E_s}{T}} \left[\frac{2\sin(\theta_n + 2\pi(2f_c + \Delta f)nT) \sin(\pi(2f_c + \Delta f)T)}{4\pi f_c + 2\pi\Delta f} \right] + w_2 \\
&= \sqrt{\frac{E_s}{T}} \left[\frac{\sin(\theta_n + 2\pi\Delta f nT) \sin(\pi\Delta f T) T}{\pi\Delta f T} \right] \\
&\quad - \sqrt{\frac{E_s}{T}} \left[\frac{\sin(\theta_n + 2\pi(2f_c + \Delta f)nT) \sin(\pi(2f_c + \Delta f)T) T}{(2\pi f_c + \pi\Delta f) T} \right] + w_2 \Rightarrow \\
y_n &= \sqrt{E_s T} \sin(\theta_n + 2\pi\Delta f nT) \text{sinc}(\Delta f T) \\
&\quad - \sqrt{E_s T} \sin(\theta_n + 2\pi(2f_c + \Delta f)nT) \text{sinc}((2f_c + \Delta f)T) + w_2 \tag{3.10}
\end{aligned}$$

Αφού το f_c για δορυφορικά συστήματα συνήθως παίρνει τιμές μεταξύ 20GHz και 30GHz , τότε $\text{sinc}((2f_c + \Delta f)T) \rightarrow 0$. Με αυτήν την παραδοχή, οι τελικές εκφράσεις για τα x_n και y_n είναι

$$x_n = \sqrt{E_s T} \cos(\theta_n + 2\pi(\Delta f)nT) \text{sinc}(\Delta f T) + w_1 \tag{3.11}$$

$$y_n = \sqrt{E_s T} \sin(\theta_n + 2\pi(\Delta f)nT) \text{sinc}(\Delta f T) + w_2 \tag{3.12}$$

Από τα παραπάνω, εξάγεται ότι η (3.6) είναι

$$\phi_n = \theta + k_n \frac{2\pi}{m} + 2\pi n(\Delta f)T + \epsilon_n \quad \epsilon_n \text{ θόρυβος φάσης λόγω AWGN} \tag{3.13}$$

3.2 Βήματα του αλγορίθμου του εκτιμητή V&V

Από το προηγούμενο μέρος μπορούν να εξαχθούν 5 διακριτά βήματα για την λειτουργία του αλγορίθμου του εκτιμητή V&V:

1. μετατροπή από καρτεσιανές συντεταγμένες σε πολικές,
2. αφαίρεση των δεδομένων, υψώνοντας στην m το εκάστοτε σύμβολο και εφαρμόζοντας την συνάρτηση $F(\bullet)$ στην ενέργεια του συμβόλου,
3. μετατροπή των συντεταγμένων από πολικές σε καρτεσιανές,
4. μέσος όρος των πραγματικών και φανταστικών μερών των συμβόλων, μετά την αφαίρεση της διαμόρφωσης και
5. εύρεση της εκτίμησης $\hat{\theta}_n$, χρησιμοποιώντας μετατροπή από καρτεσιανές σε πολικές συντεταγμένες και διαιρώντας με m .

3.3 Μοντελοποίηση του προβλήματος

Το σήμα θα μπορούσε να περιγραφεί χρησιμοποιώντας μιγαδικούς αριθμούς ως

$$y_n = A \cdot e^{j(\theta + k_n \frac{2\pi}{m} + 2\pi n(\Delta f)T)} + w_n \quad (3.14)$$

με

$A = \sqrt{\frac{2E_s}{T}}$	ενέργεια συμβόλου
θ	θόρυβος φάσης
$k_n \in [0, \dots, m-1]$	πληροφορία
Δf	υπόλοιπο συχνότητας με $(\Delta f)T \ll 1$
T	περίοδο συμβόλου
$w_n \sim \mathcal{N}(0, \sigma_w^2)$	μιγαδικό θόρυβο

Αυτή είναι η μορφή ενός mPSK σήματος στη βασική ζώνη (baseband representation). Φυσικά, η (3.14) προϋποθέτει ότι έχει γίνει σωστό φιλτράρισμα και δειγματοληψία του σήματος σύμφωνα και με το όριο Nyquist.

Απώτερος σκοπός των εκτιμητών θορύβου φάσης είναι η προσέγγιση των παραμέτρων Δf και θ , χρησιμοποιώντας μόνον κάποια δείγματα y_n . Αν υποθεθεί ότι τα Δf και θ εκτιμηθούν τέλεια, το

$$\begin{aligned} y'_n &= y_n \cdot e^{-j(\theta + 2\pi n(\Delta f)T)} \\ &= A e^{jk_n \frac{2\pi}{m}} + w'_n \end{aligned}$$

είναι το σύμβολο απαλλαγμένο από το θόρυβο φάσης και το υπόλοιπο συχνότητας. Κατά την αποδιαμόρφωση το μόνο που απασχολεί πλέον είναι η παρουσία του θορύβου του καναλιού, ο οποίος κατά τον πολλαπλασιασμό απλώς υφίσταται μια μετατόπιση φάσης κατά $(\theta + 2\pi n(\Delta f)T)$.

3.4 Εναλλακτική μορφή του εκτιμητή V&V

Για την υπόλοιπη εργασία θα χρησιμοποιηθεί η μορφή του εκτιμητή που εμφανίζεται στο [1] και συνοψίζει όλη την λειτουργία του σε μία εξίσωση. Είναι απόλυτα ισοδύναμη της (3.8) και για το σύμβολο $n = 0$ είναι

$$\hat{\theta} = \frac{1}{m} \arg \left\{ \sum_{n=-N}^N F(\rho_n) e^{jm\phi_n} \right\} \quad (3.15)$$

ή για οποιοδήποτε σύμβολο n

$$\hat{\theta}_n = \frac{1}{m} \arg \left\{ \sum_{l=-N}^N F(\rho_{l+n}) e^{jm\phi_{l+n}} \right\} \quad (3.16)$$

Τα παραπάνω είναι ισοδύναμα με τα

$$\hat{\theta} = \frac{1}{m} \arg \left\{ \frac{1}{2N+1} \sum_{n=-N}^N F(\rho_n) e^{jm\phi_n} \right\} \quad (3.17)$$

$$\hat{\theta}_n = \frac{1}{m} \arg \left\{ \frac{1}{2N+1} \sum_{l=-N}^N F(\rho_{l+n}) e^{jm\phi_{l+n}} \right\} \quad (3.18)$$

3.5 Αμεροληψία εκτιμητή

3.5.1 Εκτιμητής φάσης για BPSK και $F(\bullet) = (\bullet)^2$

Έστω σήμα διαμορφωμένο κατά BPSK, δηλαδή $y_n = A \cdot e^{j(\theta + k_n\pi + 2\pi n(\Delta f)T)} + w_n$, και ότι χρησιμοποιείται η μη γραμμική συνάρτηση $F(\bullet) = (\bullet)^2$. Τότε,

$$\begin{aligned} x'_n + j \cdot y'_n &= F(\rho_n) e^{j2\phi_n} \Rightarrow \\ x'_n + j \cdot y'_n &= (\rho_n \cdot e^{j\phi_n})^2 \end{aligned} \quad (3.19)$$

με

$$\begin{aligned} \rho_n &= |y_n| = \sqrt{x_n^2 + y_n^2} \\ \phi_n &= \theta + k_n\pi + 2\pi n(\Delta f)T + \epsilon_n \end{aligned} \quad \epsilon_n \text{ θόρυβος φάσης λόγω AWGN}$$

Η εκτίμηση του θορύβου φάσης θ για το κεντρικό σύμβολο ενός παραθύρου μήκους $2N + 1$

συμβόλων είναι

$$\begin{aligned}
 \hat{\theta} &= \frac{1}{2} \arg \left\{ \frac{1}{2N+1} \sum_{n=-N}^N F(\rho_n) e^{j2\phi_n} \right\} \\
 &= \frac{1}{2} \arg \left\{ \frac{1}{2N+1} \sum_{n=-N}^N (\rho_n e^{j\phi_n})^2 \right\} \\
 &= \frac{1}{2} \arg \left\{ \frac{1}{2N+1} \sum_{n=-N}^N \left[A e^{j(\theta+k_n\pi+2\pi(\Delta f)Tn)} + w_n \right]^2 \right\} \\
 &\stackrel{\lambda=2\pi(\Delta f)T}{=} \frac{1}{2} \arg \left\{ \frac{1}{2N+1} \sum_{n=-N}^N \left[A e^{j(\theta+k_n\pi+\lambda n)} + w_n \right]^2 \right\} \\
 &= \frac{1}{2} \arg \left\{ \frac{1}{2N+1} \sum_{n=-N}^N \left[A^2 e^{2j(\theta+k_n\pi+\lambda n)} + w_n^2 + A e^{-j(\theta+k_n\pi+\lambda n)} w_n \right] \right\} \\
 &= \frac{1}{2} \arg \left\{ \frac{1}{2N+1} \left[\sum_{n=-N}^N A^2 e^{2j(\theta+k_n\pi+\lambda n)} + \sum_{n=-N}^N w_n^2 + \sum_{n=-N}^N 2A e^{j(\theta+k_n\pi+\lambda n)} w_n \right] \right\}
 \end{aligned}$$

Εφόσον ο θόρυβος w_n είναι μιγαδικός

$$\begin{aligned}
 \sum_{n=-N}^N w_n^2 &= \sum_{n=-N}^N (\operatorname{Re}\{w_n\} + j \cdot \operatorname{Im}\{w_n\})^2 \\
 &= \sum_{n=-N}^N (\operatorname{Re}\{w_n\}^2 - \operatorname{Im}\{w_n\}^2 - 2\operatorname{Re}\{w_n\} \operatorname{Im}\{w_n\}) \\
 &= \sum_{n=-N}^N \left(\frac{\sigma_w^2}{2} - \frac{\sigma_w^2}{2} \right) \\
 &= 0
 \end{aligned} \tag{3.20}$$

και βάση του Νόμου των Μεγάλων Αριθμών (NMA) τότε

$$\frac{1}{2N+1} \sum_{n=-N}^N 2A e^{j(\theta+k_n\pi+\lambda n)} w_n \approx \frac{1}{2N+1} \sum_{n=-N}^N \mathbb{E} \left\{ 2A e^{j(\theta+k_n\pi+\lambda n)} w_n \right\} = 0$$

αφού ο θόρυβος w_n και το σήμα είναι ασυσχέτιστα.

Άρα,

$$\begin{aligned}
 \hat{\theta} &= \frac{1}{2} \arg \left\{ \frac{1}{2N+1} \sum_{n=-N}^N A^2 e^{2j(\theta+k_n\pi+\lambda n)} \right\} \\
 &= \frac{1}{2} \arg \left\{ A^2 \sum_{n=-N}^N e^{j(2\theta+2\lambda n)} \right\} \\
 &= \frac{1}{2} \arg \left\{ e^{j2\theta} \sum_{n=-N}^N e^{j2\lambda n} \right\} \\
 &= \frac{1}{2} \arg \left\{ e^{j2\theta} \sum_{n=0}^{2N} e^{j2\lambda(n-N)} \right\} \\
 &= \frac{1}{2} \arg \left\{ e^{j2\theta} e^{j2\lambda(-N)} \sum_{n=0}^{2N} e^{j2\lambda n} \right\} \\
 &= \frac{1}{2} \arg \left\{ e^{j2\theta} e^{j2\lambda(-N)} \frac{1 - e^{j2\lambda(2N+1)}}{1 - e^{j2\lambda}} \right\} \\
 &= \frac{1}{2} \arg \left\{ e^{j2\theta} \frac{e^{j2\lambda(-N)} - e^{j2\lambda(N+1)}}{1 - e^{j2\lambda}} \right\} \Rightarrow \\
 \hat{\theta} &= \frac{1}{2} \arg \left\{ e^{j2\theta} \cdot \Omega_2 \right\} \tag{3.21}
 \end{aligned}$$

Ο εκτιμητής είναι αμερόληπτος.

Στο [2] αποδεικνύεται πως για κάθε σύμβολο με $n \neq 0$, ο εκτιμητής είναι μεροληπτικός με

$$\mathbb{E} \left[\hat{\theta}_n \right] = 2\pi n(\Delta f)T, \quad -N \leq n \leq N \tag{3.22}$$

Πράγματι, αν πρέπει να εκτιμηθεί ο θόρυβος φάσης θ για ένα οποιοδήποτε σύμβολο n τότε

$$\begin{aligned}
 \hat{\theta}_n &= \frac{1}{2} \arg \left\{ \frac{1}{2N+1} \sum_{l=-N}^N \left[A e^{j(\theta + \frac{k_l+n\pi}{2} + \lambda(l+n))} + w_{l+n} \right]^2 \right\} \\
 &= \frac{1}{2} \arg \left\{ e^{j2(\theta+\lambda n)} \cdot \Omega_2 \right\} \stackrel{\lambda=2\pi(\Delta f)T}{\Rightarrow} \\
 \hat{\theta}_n &= \frac{1}{2} \arg \left\{ e^{j2(\theta+2\pi(\Delta f)Tn)} \cdot \Omega_2 \right\} \tag{3.23}
 \end{aligned}$$

3.5.2 Εκτιμητής φάσης για QPSK και $F(\bullet) = (\bullet)^4$

Έστω σήμα διαμορφωμένο κατά QPSK, δηλαδή $y_n = A \cdot e^{j(\theta + \frac{k_n\pi}{2} + 2\pi n(\Delta f)T)} + w_n$, και ότι χρησιμοποιείται η μη γραμμική συνάρτηση $F(\bullet) = (\bullet)^4$. Τότε,

$$\begin{aligned}
 x'_n + j \cdot y'_n &= F(\rho_n) e^{j4\phi_n} \Rightarrow \\
 x'_n + j \cdot y'_n &= (\rho_n \cdot e^{j\phi_n})^4 \tag{3.24}
 \end{aligned}$$

με

$$\rho_n = |y_n| = \sqrt{x_n^2 + y_n^2}$$

$$\phi_n = \theta + \frac{k_n \pi}{2} + 2\pi n(\Delta f)T + \epsilon_n \quad \epsilon_n \text{ θόρυβος φάσης λόγω AWGN}$$

Συνεπώς η εκτίμηση του θορύβου φάσης θ για το κεντρικό σύμβολο του παραθύρου είναι

$$\begin{aligned} \hat{\theta} &= \frac{1}{4} \arg \left\{ \frac{1}{2N+1} \sum_{n=-N}^N F(\rho_n) e^{j4\phi_n} \right\} \\ &= \frac{1}{4} \arg \left\{ \frac{1}{2N+1} \sum_{n=-N}^N (\rho_n e^{j\phi_n})^4 \right\} \\ &= \frac{1}{4} \arg \left\{ \frac{1}{2N+1} \sum_{n=-N}^N \left[A e^{j(\theta + \frac{k_n \pi}{2} + \lambda n)} + w_n \right]^4 \right\} \end{aligned}$$

Δεδομένου ότι ισχύει $(a+b)^4 = a^4 + b^4 + 4a^3b + 4ab^3 + 6a^2b^2$, τότε

$$\begin{aligned} \frac{1}{2N+1} \sum_{n=-N}^N \left[A e^{j(\theta + \frac{k_n \pi}{2} + \lambda n)} + w_n \right]^4 &= \frac{1}{2N+1} \sum_{n=-N}^N \left[A^4 e^{4j(\theta + \frac{k_n \pi}{2} + \lambda n)} \right] \\ &\quad + \frac{1}{2N+1} \sum_{n=-N}^N [w_n^4] \\ &\quad + \frac{1}{2N+1} \sum_{n=-N}^N \left[4A^3 e^{3j(\theta + \frac{k_n \pi}{2} + \lambda n)} w_n \right] \\ &\quad + \frac{1}{2N+1} \sum_{n=-N}^N \left[4A e^{j(\theta + \frac{k_n \pi}{2} + \lambda n)} w_n^3 \right] \\ &\quad + \frac{1}{2N+1} \sum_{n=-N}^N \left[6A^2 e^{2j(\theta + \frac{k_n \pi}{2} + \lambda n)} w_n^2 \right] \end{aligned}$$

Όμως,

$$\begin{aligned} \sum_{n=-N}^N [w_n^4] &= \sum_{n=-N}^N [w_n^2]^2 = 0 \\ \frac{1}{2N+1} \sum_{n=-N}^N \left[4A^3 e^{3j(\theta + \frac{k_n \pi}{2} + \lambda n)} w_n \right] &= 0 \\ \frac{1}{2N+1} \sum_{n=-N}^N \left[4A e^{j(\theta + \frac{k_n \pi}{2} + \lambda n)} w_n^3 \right] &= 0 \end{aligned}$$

και

$$\begin{aligned} \frac{1}{2N+1} \sum_{n=-N}^N \left[6A^2 e^{2j(\theta + \frac{k_n \pi}{2} + \lambda n)} w_n^2 \right] &= 6A^2 e^{2j\theta} \frac{1}{2N+1} \sum_{n=-N}^N e^{jk_n \pi + 2j\lambda n} w_n^2 \\ &= 0 \end{aligned}$$

αφού

$$\mathbb{E} \left\{ e^{jk_n\pi} \right\} = \frac{1}{4} (e^{0j\pi} + e^{j\pi} + e^{2j\pi} + e^{3j\pi}) = 0$$

Άρα,

$$\begin{aligned} \hat{\theta} &= \frac{1}{4} \arg \left\{ \frac{1}{2N+1} \sum_{n=-N}^N \left[A^4 e^{4j(\theta + \frac{k_n\pi}{2} + \lambda n)} \right] \right\} \\ &= \frac{1}{4} \arg \left\{ e^{j4\theta} \sum_{n=-N}^N \left[e^{j2k_n\pi + j4\lambda n} \right] \right\} \\ &= \frac{1}{4} \arg \left\{ e^{j4\theta} \sum_{n=-N}^N \left[e^{j4\lambda n} \right] \right\} \\ &= \frac{1}{4} \arg \left\{ e^{j4\theta} e^{j4\lambda(-N)} \sum_{n=0}^{2N} \left[e^{j4\lambda n} \right] \right\} \\ &= \frac{1}{4} \arg \left\{ e^{j4\theta} e^{j4\lambda(-N)} \frac{1 - e^{j4\lambda(2N+1)}}{1 - e^{j4\lambda}} \right\} \Rightarrow \\ \hat{\theta} &= \frac{1}{4} \arg \left\{ e^{j4\theta} \cdot \Omega_4 \right\} \end{aligned} \quad (3.25)$$

Παρόμοια με την περίπτωση της BPSK (3.21), ο εκτιμητής είναι αμερόληπτος.

Αν $n \neq 0$ τότε η εκτίμηση είναι

$$\hat{\theta}_n = \frac{1}{4} \arg \left\{ e^{j4(\theta + 2\pi(\Delta f)Tn)} \cdot \Omega_4 \right\} \quad (3.26)$$

και ο εκτιμητής είναι μεροληπτικός.

3.6 Βέλτιστος εκτιμητής V&V

Έστω y_n το λαμβανόμενο σήμα ενός συμβόλου n

$$y_n = A_n \cdot e^{j\phi_n} + w_n \quad \begin{array}{l} A_n \text{ ενέργεια συμβόλου} \\ w_n \text{ μιγαδικός θόρυβος} \end{array} \quad (3.27)$$

με

$$\phi_n = k_n \frac{2\pi}{m} + 2\pi n(\Delta f)T + \theta$$

Οι άγνωστες παράμετροι είναι αφενός το υπόλοιπο συχνότητας Δf , που εισάγεται λόγω ατελούς ανάκτησης της φέρουσας, ο θόρυβος φάσης θ , ο οποίος και αναζητείται, ο μιγαδικός θόρυβος w_n , αλλά και η πληροφορία k_n του συμβόλου. Εφόσον το w_n δεν είναι δυνατόν να υπολογιστεί ή να εκτιμηθεί, η γνώση των k_n και θα επέφερε μία πολύ καλή εκτίμηση των θ και Δf , αλλά και του θορύβου φάσης που προκαλεί το w_n .

Ο εκτιμητής που λαμβάνει υπόψην του την πληροφορία που φέρουν τα σύμβολα ονομάζεται βέλτιστος εκτιμητής V&V (*optimal V&V estimator*) και μπορεί να καταταχθεί στους decision directed εκτιμητές.

3.6.1 Βέλτιστος εκτιμητής V&V για $\Delta f = 0$

Αν τα k_n θεωρηθούν γνωστά εκ των προτέρων και το $\Delta f = 0$, τότε

$$s_n = Ae^{j(k_n \frac{2\pi}{m} + \theta)} = Ae^{j(Z_n + \theta)} \quad (3.28)$$

το σήμα γνωστών συμβόλων.

Ο θόρυβος φάσης θ , για το κεντρικό σύμβολο ενός παραθύρου $2N + 1$ συμβόλων, μπορεί να εκτιμηθεί ως εξής:

$$\begin{aligned} \hat{\theta} &= \arg \max_{\theta} \{f(y|s)\} \\ &= \arg \max_{\theta} \left\{ \prod_{n=-N}^N \frac{1}{\sqrt{2\pi\sigma_w^2}} e^{-\frac{|y_n - s_n|^2}{2\sigma_w^2}} \right\} \\ &= \arg \max_{\theta} \left\{ e^{-\frac{1}{2\sigma_w^2} \sum_{n=-N}^N |y_n - Ae^{j(\theta + Z_n)}|^2} \right\} \\ &= \arg \max_{\theta} \left\{ e^{-\frac{1}{2\sigma_w^2} \sum_{n=-N}^N [|y_n|^2 + |Ae^{j(\theta + Z_n)}|^2 - 2\operatorname{Re}\{y_n \cdot (Ae^{j(\theta + Z_n)})^*\}]} \right\} \\ &= \arg \max_{\theta} \left\{ e^{\frac{1}{\sigma_w^2} \sum_{n=-N}^N \operatorname{Re}\{y_n \cdot (Ae^{j(\theta + Z_n)})^*\}} \right\} \\ &= \arg \max_{\theta} \left\{ e^{\frac{1}{\sigma_w^2} A \sum_{n=-N}^N \operatorname{Re}\{y_n e^{-j(Z_n + \theta)}\}} \right\} \\ &= \arg \max_{\theta} \left\{ \sum_{n=-N}^N \operatorname{Re}\{y_n e^{-jZ_n} \cdot e^{-j\theta}\} \right\} \\ &= \arg \max_{\theta} \left\{ \operatorname{Re}\left\{ \sum_{n=-N}^N (y_n e^{-jZ_n}) \cdot e^{-j\theta} \right\} \right\} \\ &= \arg \max_{\theta} \left\{ \cos \left[\angle \left(\sum_{n=-N}^N y_n e^{-jZ_n} \right) - \theta \right] \right\} \end{aligned}$$

Η συνάρτηση έχει μέγιστο όταν

$$\begin{aligned} \angle \left(\sum_{n=-N}^N y_n e^{-jZ_n} \right) - \theta &= 0 \Rightarrow \\ \theta &= \angle \left(\sum_{n=-N}^N y_n e^{-jZ_n} \right) \end{aligned} \quad (3.29)$$

Συνεπώς, αν η πληροφορία k_n είναι γνωστή, ο βέλτιστος εκτιμητής είναι

$$\hat{\theta} = \angle \left(\sum_{n=-N}^N y_n e^{-jk_n \frac{2\pi}{m}} \right) \quad (3.30)$$

και για οποιοδήποτε σύμβολο

$$\hat{\theta}_n = \angle \left(\sum_{l=-N}^N y_{n+l} e^{-jk_{n+l} \frac{2\pi}{m}} \right) \quad (3.31)$$

3.6.2 Βέλτιστος εκτιμητής V&V για $\Delta f \neq 0$

Αν υποτεθεί ότι το $\Delta f \neq 0$, τα σύμβολα γίνονται

$$s_n = Ae^{j(k_n \frac{2\pi}{m} + 2\pi(\Delta f)nT + \theta)} = Ae^{j(Z_n + \theta + n\Delta)} \quad (3.32)$$

Η εκτίμηση του θορύβου φάσης θ και του υπολοίπου συχνότητας Δf ενός οποιουδήποτε συμβόλου είναι

$$[\hat{\theta}_n, \hat{\Delta f}_n] = \arg \max_{\theta, \Delta f} \left\{ \operatorname{Re} \left\{ e^{-j\theta} \sum_{l=-N}^N \left(y_{n+l} e^{-jZ_{n+l}} \cdot e^{-j2\pi(n+l)(\Delta f)T} \right) \right\} \right\} \quad (3.33)$$

Συνοπτικά, η εκτίμηση του υπολοίπου συχνότητας Δf είναι

$$\hat{\Delta f}_n = \frac{1}{m} \arg \max_{|(\Delta f)T| < \frac{1}{2}} \left\{ \frac{1}{N} \left| \sum_{l=-N}^N y_{n+l} e^{-j2\pi(n+l)(\Delta f)T} \right| \right\} \quad (3.34)$$

και του θορύβου φάσης θ

$$\hat{\theta}_n = \frac{1}{m} \angle \left(\sum_{l=-N}^N y_{n+l} e^{-j2\pi M(n+l)(\Delta f_n)T} \right) \quad (3.35)$$

Η λύση παρουσιάζεται αναλυτικά στο [6].

3.7 Εκτιμητής V&V πολλαπλών περασμάτων

Η (3.31) δίνει την καλύτερη εκτίμηση. Παρόλα αυτά, η πληροφορία δεν είναι γνωστή εκ των προτέρων. Συνεπώς, μια λύση είναι να εκτιμηθεί ο θόρυβος φάσης, να γίνει μια πρώτη προσπάθεια εύρεσης της πληροφορίας k_n και τέλος να γίνει εκ νέου εκτίμηση του θορύβου φάσης.

Ο απλός εκτιμητής V&V, όπως εμφανίζεται στην (3.17), δύναται να επεκταθεί και να χρησιμοποιηθεί *εξέταση υποθέσεων*, ώστε να επιτευχθεί καλύτερη εκτίμηση του θορύβου φάσης θ . Δημιουργείται έτσι ένας εκτιμητής που αρχικά εκτιμά τον θόρυβο φάσης με τον αλγόριθμο V&V, τον αφαιρεί και τον επανεκτιμά, έχοντας αποκωδικοποιήσει τα σύμβολα και χρησιμοποιώντας την λογική του βέλτιστου εκτιμητή. Ουσιαστικά, είναι ένας εκτιμητής V&V με ανατροφοδότηση και χρήση των αποκωδικοποιούμενων συμβόλων, που τον κατατάσσει στην οικογένεια των feedback decision-directed εκτιμητών.

Αυτός ο εκτιμητής θα αναφέρεται από εδώ και στο εξής σαν *εκτιμητής V&V πολλαπλών περασμάτων* (*Multipass V&V estimator*).

Αλγόριθμος εκτιμητή V&V πολλαπλών περασμάτων

Έστω διάνυσμα \mathbf{W} με $6N + 1$ δείγματα που ακολουθούν διαμόρφωση κατά mPSK

$$\mathbf{W} = [y_{n-3N} \dots y_n \dots y_{n+3N}] \quad (3.36)$$

με

$$y_n = A_n e^{j\phi_n} + w_n \quad \begin{array}{l} A_n \text{ ενέργεια συμβόλου,} \\ w_n \sim \mathcal{N}(0, \sigma_w^2) \end{array}$$

και

$$\phi_n = k_n \frac{2\pi}{m} + 2\pi(\Delta f)nT + \theta \quad \begin{array}{l} k_n \text{ πληροφορία} \\ \theta \text{ θόρυβος φάσης} \end{array}$$

Βήμα 1 Εκτίμηση του θορύβου φάσης του κάθε συμβόλου στο παράθυρο χρησιμοποιώντας τον εκτιμητή V&V όπως στην (3.17). Το αποτέλεσμα είναι ένα διάνυσμα Θ με τον εκτιμώμενο θόρυβο φάσης των $4N + 1$ κεντρικών συμβόλων

$$\Theta = [\hat{\theta}_{n-2N} \cdots \hat{\theta}_n \cdots \hat{\theta}_{n+2N}] \quad (3.37)$$

με

$$\hat{\theta}_n = \frac{1}{m} \arg \left\{ \sum_{l=-N}^N F(\rho_{l+n}) e^{jm\phi_{l+n}} \right\} \quad F(\bullet) = (\bullet)^i, \quad i = 1, 2, \dots$$

Βήμα 2 Αφαίρεση του εκτιμώμενου θορύβου φάσης Θ από τα σήματα που απαρτίζουν το διάνυσμα \mathbf{W} . Το διάνυσμα \mathbf{W}' έχει τα δείγματα μετά την αφαίρεση του εκτιμώμενου θορύβου φάσης.

$$\mathbf{W}' = [y'_{n-2N} \cdots y'_n \cdots y'_{n+2N}] \quad (3.38)$$

με

$$\mathbf{W}' = [y_{n-2N} \cdots y_n \cdots y_{n+2N}] \cdot \begin{bmatrix} e^{-j\hat{\theta}_{n-2N}} & 0 & \cdots & 0 \\ & \ddots & & \vdots \\ 0 & & e^{-j\hat{\theta}_n} & 0 \\ \vdots & \ddots & & \ddots \\ 0 & \cdots & 0 & & e^{-j\hat{\theta}_{n+2N}} \end{bmatrix}$$

Βήμα 3 Εύρεση του διανύσματος \mathbf{K}' της πληροφορίας από το διάνυσμα \mathbf{W}' χρησιμοποιώντας εξέταση υποθέσεων

$$\mathbf{K}' = [k'_{n-2N} \cdots k'_n \cdots k'_{n+2N}] \quad (3.39)$$

Για την εύρεση της πληροφορίας, μπορεί να ακολουθηθεί η μεθοδολογία που βρίσκεται στο [8] και παρουσιάζεται παρακάτω.

Ανίχνευση συμβόλων Έστω δύο τυχαίες μεταβλητές, χ_1 και χ_2 , ανεξάρτητες και γκαουσιανές. Οι τυχαίες αυτές μεταβλητές ικανοποιούν τις ακόλουθες m υποθέσεις

$$H_k: \quad \chi_1 = \mathcal{N}\left(A \cos\left(k \frac{2\pi}{m}\right), \sigma_w^2\right), \quad \chi_2 = \mathcal{N}\left(A \sin\left(k \frac{2\pi}{m}\right), \sigma_w^2\right), \quad k = 0, \dots, m-1$$

$$A > 0$$

Η πυκνότητα πιθανότητας των δεδομένων κάτω από κάθε υπόθεση είναι

$$f_k(x_1, x_2) = \frac{1}{2\pi\sigma_w^2} e^{-\frac{1}{2\sigma_w^2} [(x_1 - A \cos(k \frac{2\pi}{m}))^2 + (x_2 - A \sin(k \frac{2\pi}{m}))^2]} \quad k = 0, \dots, m-1$$

με $x_1 + jx_2 = y$ το λαμβανόμενο σήμα.

Εφόσον δεν υπάρχει κάποιος ιδιαίτερος λόγος να μην θεωρηθούν οι υποθέσεις ισοπίθανες, το βέλτιστο τεστ που αποφασίζει υπέρ της υπόθεσης $H_{k'}$, και εφόσον πρόκειται για mPSK υπέρ του συμβόλου k' , είναι

$$\begin{aligned} k' &= \arg \max_{i=0, \dots, m-1} \{f_i(x_1, x_2)\} \\ &= \arg \max_{i=0, \dots, m-1} \left\{ \frac{1}{2\pi\sigma_w^2} e^{-\frac{1}{2\sigma_w^2} [(x_1 - A \cos(i \frac{2\pi}{m}))^2 + (x_2 - A \sin(i \frac{2\pi}{m}))^2]} \right\} \\ &= \arg \max_{i=0, \dots, m-1} \left\{ -\frac{1}{2\sigma_w^2} \left[\left(x_1 - A \cos \left(i \frac{2\pi}{m} \right) \right)^2 + \left(x_2 - A \sin \left(i \frac{2\pi}{m} \right) \right)^2 \right] \right\} \\ &= \arg \min_{i=0, \dots, m-1} \left\{ \left(x_1 - A \cos \left(i \frac{2\pi}{m} \right) \right)^2 + \left(x_2 - A \sin \left(i \frac{2\pi}{m} \right) \right)^2 \right\} \\ &= \arg \min_{i=0, \dots, m-1} \left\{ x_1 \cos \left(i \frac{2\pi}{m} \right) + x_2 \sin \left(i \frac{2\pi}{m} \right) \right\} \\ &= \arg \min_{i=0, \dots, m-1} \left\{ |y| \cos(\angle(y)) \cos \left(i \frac{2\pi}{m} \right) + |y| \sin(\angle(y)) \sin \left(i \frac{2\pi}{m} \right) \right\} \\ &= \arg \min_{i=0, \dots, m-1} \left\{ \cos \left(\angle(y) - i \frac{2\pi}{m} \right) \right\} \Rightarrow \\ & \quad k' = \arg \min_{i=0, \dots, m-1} \left\{ \left| \angle(y) - i \frac{2\pi}{m} \right| \right\} \end{aligned} \quad (3.40)$$

Βήμα 4 Εκ νέου εκτίμηση του θορύβου φάσης για κάθε σήμα του διανύσματος \mathbf{W} , χρησιμοποιώντας τον βέλτιστο εκτιμητή V&V της (3.31) με την με τα διανύσμα \mathbf{K}' και \mathbf{W} .

$$\Theta' = [\hat{\theta}'_{n-N} \cdots \hat{\theta}'_n \cdots \hat{\theta}'_{n+N}] \quad (3.41)$$

με

$$\hat{\theta}'_n = \angle \left(\sum_{l=-N}^N y_{n+l} e^{-jk'_{n+l} \frac{2\pi}{m}} \right)$$

Αν αρχικά στο **Βήμα 1** είχαν χρησιμοποιηθεί μόνον $4N + 1$ δείγματα, τότε ο αλγόριθμος θα κατέληγε στην εκτίμηση του θορύβου φάσης του κεντρικού συμβόλου (n).

Βήμα 5 Αν και ο αλγόριθμος μπορεί να τερματιστεί εδώ και να έχει επιτευχθεί κάποια καλή εκτίμηση του θορύβου φάσης για κάθε ένα από τα σήματα στο διάνυσμα $\mathbf{W}'' = [y_{n-N} \cdots y_n \cdots y_{n+N}]$, υπάρχει η εναλλακτική να επιστρέψει στο **Βήμα 2** με $\Theta \equiv \Theta'$, ώστε να γίνουν και άλλα περάσματα του αλγορίθμου από τα δεδομένα.

Έτσι ξεκινά ένας δεύτερος κύκλος εκτίμησης, εκτιμώντας ακόμα καλύτερα το θόρυβο φάσης. Ο αλγόριθμος μπορεί να τερματιστεί όταν φανεί ότι ο εκτιμώμενος θόρυβος φάσης συγκλίνει προς κάποια τιμή ή όταν απλά τελειώσουν τα διαθέσιμα δείγματα.

Όσα περισσότερα περάσματα γίνονται στον αλγόριθμο, τόσο μεγαλύτερο παράθυρο χρειάζεται αρχικά. Κατά το πρώτο πέρασμα χρησιμοποιούνται $6N + 1$ δείγματα και παράγονται $4N + 1$ εκτιμήσεις. Αυτές χρησιμοποιούνται από τον βέλτιστο εκτιμητή V&V και παράγονται $2N + 1$ εκτιμήσεις. Κάθε επόμενο πέρασμα μειώνει κατά $2N$ τα δείγματα.

Έτσι, αν υποθεθεί ότι κάποια υλοποίηση του αλγορίθμου πρέπει να κάνει x περάσματα – με το πρώτο να προσμετράται, τότε ο αρχικός αριθμός συμβόλων στο παράθυρο για την εκτίμηση του κεντρικού συμβόλου n δίνεται από την σχέση

$$\#_{\text{symbols}} = (x + 1)2N + 1 \quad (3.42)$$

Ο παραπάνω αλγόριθμος εκτίμησης δουλεύει καλά όταν το $(\Delta f)T \rightarrow 0$. Αν χρειάζεται ακόμα μεγαλύτερη ακρίβεια στην εκτίμηση, ο αλγόριθμος παραπάνω μπορεί να επεκταθεί στο **Βήμα 4** με εκτίμηση και του Δf , χρησιμοποιώντας τις (3.34) και (3.35). Έπειτα, στο **Βήμα 2** μπορεί να αφαιρεθεί και το Δf από το σύμβολο, ώστε να προσεγγίζεται η ακρίβεια της εκτίμησης του βέλτιστου εκτιμητή.

Κεφάλαιο 4

ΕΚΤΙΜΗΤΕΣ ΒΑΡΥΚΕΝΤΡΟΥ

4.1 Εισαγωγή στους εκτιμητές βαρυκέντρου

Αν και ο εκτιμητής $V\&V$ είναι αρκετά καλός, ιδίως όταν $SNR > 2$ dB, η υλοποίηση του σε συστήματα είναι σχετικά πολύπλοκη. Η προσπάθεια λειτουργίας αποκλειστικά σε πολικές συντεταγμένες για την μείωση της κατασκευαστικής πολυπλοκότητας των εκτιμητών $V\&V$ οδήγησε στην ανάπτυξη των *εκτιμητών βαρυκέντρου* [3]. Έτσι αφαιρέθηκε το βάρος των συνεχών μετατροπών από πολικές σε καρτεσιανές συντεταγμένες, και το ανάποδο, από τις υλοποιήσεις. Η αφαίρεση αυτών των μετατροπών απλοποιεί σε μεγάλο βαθμό τους εκτιμητές και την υλοποίηση τους.

Οι εκτιμητές βαρυκέντρου βασίζονται και αυτοί στην εύρεση του μέσου όρου του υπολοίπου φάσης κάποιου πλήθους n συμβόλων, μετά την αφαίρεση της διαμόρφωσης από τα σύμβολα. Πρόκειται για εκτιμητές που, όπως ο $V\&V$, ανήκουν στους feed-forward NDA εκτιμητές.

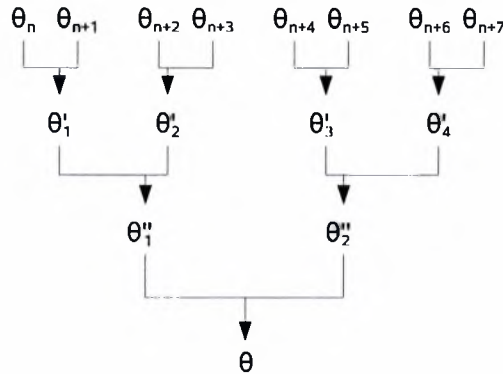
4.2 Εύρεση μέσου όρου υπολοίπου φάσης από πλήθος n συμβόλων

Για να βρεθεί η τελική εκτίμηση του θορύβου φάσης στους εκτιμητές βαρυκέντρου, απαιτούνται συνολικά τρία βήματα:

1. Επιλέγονται n σύμβολα και αφαιρείται η διαμόρφωσή τους. Έτσι προκύπτει ο θόρυβος στη φάση για το κάθε ένα σύμβολο ή αλλιώς το υπόλοιπο φάσης (θόρυβος φάσης).
2. Οι αρχικές n τιμές του υπολοίπου φάσης αθροίζονται ανά δύο χρησιμοποιώντας κάποια κατάλληλη συνάρτηση (συνάρτηση βαρυκέντρου).
3. Οι προκύπτουσες $\frac{n}{2}$ τιμές αθροίζονται εκ νέου με την συνάρτηση βαρυκέντρου που έχει επιλεγθεί. Το βήμα επαναλαμβάνεται μέχρι να καταλήξει σε μία και μοναδική τιμή, την εκτίμηση φάσης για το πρώτο από τα n αρχικά σύμβολα.

Ο αλγόριθμος μπορεί να λειτουργήσει μόνον εάν $n = 2^l$, $l \in \mathbb{N}$.

Οπτικά, ο αλγόριθμος εμφανίζεται στο Σχήμα(4.1). Το κάθε βήμα του αλγορίθμου πραγματοποιείται από μία συνάρτηση που απαντάται στην βιβλιογραφία σαν $MEMARB(st, sv)$, με st ο



Σχήμα 4.1: Στιγμιότυπο εύρεσης μέσου όρου 8 τιμών

αριθμός του βήματος και sv οι τιμές που πρέπει να αθροιστούν ανά δύο. Η εύρεση του μέσου όρου δύο τιμών πραγματοποιείται από τις συναρτήσεις βαρυκέντρου που θα παρουσιαστούν παρακάτω.

4.3 Αφαίρεση διαμόρφωσης – συνάρτηση *FOLD*

Πριν από οποιοδήποτε άλλο βήμα των αλγορίθμων των εκτιμητών βαρυκέντρου πρέπει να αφαιρεθεί η διαμόρφωση από τα σύμβολα, ειδικά οι υπολογισμοί δεν έχουν κάποιο θεωρητικό ή πρακτικό νόημα.

Στο [3] εμφανίζεται ένας τρόπος αφαίρεσης της διαμόρφωσης ο οποίος βασίζεται σε επαναληπτικό αλγόριθμο. Η εν λόγω συνάρτηση που επιστρέφει την φάση του συμβόλου μετά από την αφαίρεση της διαμόρφωσης ονομάζεται *FOLD*.

Αλγόριθμος αφαίρεσης διαμόρφωσης για QPSK

Έστω ότι ένα σύμβολο QPSK περιγράφεται από την εξίσωση $y_n = A \cdot e^{j\phi_n}$. Ο αλγόριθμος που αφαιρεί την διαμόρφωση, ή αλλιώς η συνάρτηση *FOLD*, ακολουθεί τα παρακάτω βήματα:

Βήμα 1 Έστω $\phi'_n = \phi_n$

Βήμα 2 Αν $\phi'_n = -\frac{\pi}{4}$ τότε τέλος, ειδικά συνεχίσε

Βήμα 3 Αν $|\phi'_n| < \frac{\pi}{4}$ τότε τέλος, ειδικά συνεχίσε

Βήμα 4 Αν $\phi'_n > 0$ τότε $\phi'_n = \phi'_n - \frac{\pi}{2}$, ειδικά $\phi'_n = \phi'_n + \frac{\pi}{2}$

Βήμα 5 Επέστρεψε στο Βήμα 2

Είναι προφανές πως από τα παραπάνω, αν $\phi_n \in [-\pi, \pi[$, τότε $\phi'_n \in [-\frac{\pi}{4}, \frac{\pi}{4}[$

Αλγόριθμος αφαίρεσης διαμόρφωσης για mPSK

Αν το σύστημα βασίζεται σε κάποια mPSK, τότε ο γενικότερος αλγόριθμος της *FOLD* είναι

Βήμα 1 Έστω $\phi'_n = \phi_n$

Βήμα 2 Αν $\phi'_n = -\frac{\pi}{m}$ τότε τέλος, ειδάλλως συνέχισε

Βήμα 3 Αν $|\phi'_n| < \frac{\pi}{m}$ τότε τέλος, ειδάλλως συνέχισε

Βήμα 4 Αν $\phi'_n > 0$ τότε $\phi'_n = \phi'_n - \frac{\pi}{m/2}$, ειδάλλως $\phi'_n = \phi'_n + \frac{\pi}{m/2}$

Βήμα 5 Επέστρεψε στο Βήμα 2

4.4 Συναρτήσεις εύρεσης μέσου όρου υπολοίπου φάσης

4.4.1 Η αρχική συνάρτηση βαρυκέντρου

Αναφέρθηκε παραπάνω ότι η *MEMARB* χρησιμοποιεί κάποιες κατάλληλες συναρτήσεις για την εύρεση του μέσου όρου του υπολοίπου φάσης από τα σύμβολα. Ο συνδυασμός της *MEMARB* με κάποια από τις συναρτήσεις που παρουσιάζονται παρακάτω, οι οποίες ονομάζονται συναρτήσεις βαρυκέντρου, δημιουργεί τον αντίστοιχο εκτιμητή.

Να τονισθεί πως η λύση του αριθμητικού μέσου όρου δεν δουλεύει σωστά στην πολική μορφή, αφού οι συναρτήσεις $\sin()$ και $\cos()$ είναι περιοδικές. Συγκεκριμένα, αν 2 σύμβολα mPSK με υπόλοιπο φάσης

$$\begin{aligned}\theta_1 &= \frac{\pi}{m} - \epsilon \\ \theta_2 &= -\frac{\pi}{m} + \epsilon\end{aligned}$$

με $0 < \epsilon \ll 1$, ο αριθμητικός μέσος όρος δύο τιμών θα έδινε

$$\hat{\theta} = \frac{\theta_1 + \theta_2}{2} = 0$$

το οποίο δεν ανταποκρίνεται στην πραγματικότητα. Στην συγκεκριμένη περίπτωση θα έπρεπε $\hat{\theta} \rightarrow \frac{\pi}{m}$.

Στο [3] προτείνεται η χρήση της συνάρτησης *BAR*, που έχει την μορφή

$$\hat{\theta} = BAR(\theta_1, \theta_2) = \theta_1 \oplus \left[\frac{1}{2} (\theta_2 \ominus \theta_1) \right] \quad (4.1)$$

με

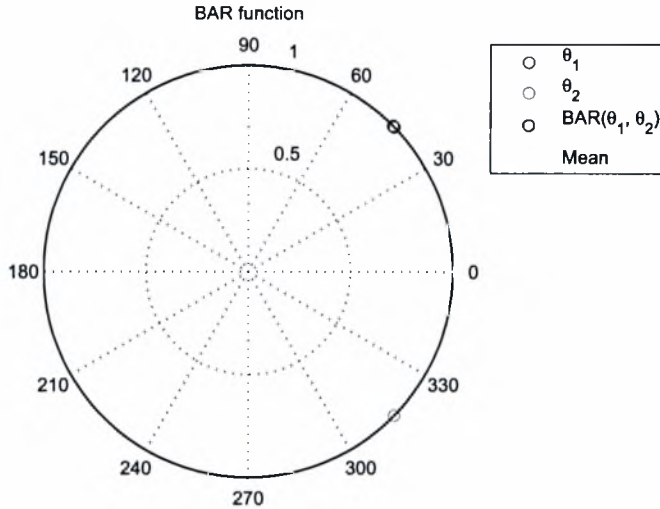
$$\begin{aligned}\theta_1, \theta_2 & \text{ υπόλοιπο φάσης συμβόλου 1 και 2 αντίστοιχα} \\ a \oplus b &= (a + b) \bmod \left(\frac{2\pi}{m} \right) \\ a \ominus b &= (a - b) \bmod \left(\frac{2\pi}{m} \right)\end{aligned}$$

Έτσι,

$$\begin{cases} \theta_2 \ominus \theta_1 &= 2\epsilon \\ \theta_1 \oplus \left[\frac{1}{2} (2\epsilon) \right] &= \frac{\pi}{m} \end{cases} \Rightarrow BAR(\theta_1, \theta_2) = \frac{\pi}{m}$$

Από τα παραπάνω, οπτικοποιημένα στο Σχήμα(4.2), φαίνεται καθαρά ότι η *BAR* βρίσκει την σωστή φάση. Με τον αριθμητικό μέσο όρο η φάση είναι 0.

Σύμφωνα με το [1], εκτιμητές που βασίζονται στην *BAR* δουλεύουν αρκετά καλά σε συνθήκες όπου το *SNR* είναι της τάξεως των 10 dB.



Σχήμα 4.2: Αποτέλεσμα συνάρτησης BAR

4.4.2 Άλλες συναρτήσεις βαρυκέντρου

Εκτός από την (4.1) έχουν προταθεί και άλλες συναρτήσεις όπως οι $EBAR$, $LBAR$ και $QLBAR$. Η γενική ιδέα πίσω από την ανάπτυξή τους ήταν η εισαγωγή μιας συνάρτησης αξιοπιστίας (reliability function) που καθορίζει με συντελεστές την βαρύτητα της κάθε τιμής στους υπολογισμούς. Ουσιαστικά, αξιολογεί τις τιμές και αυτή η αξιολόγησή τους, ή αλλιώς η αξιοπιστία τους, τις συνοδεύει σε κάθε υπολογισμό.

Όταν ξεκινάει ο υπολογισμός σε κάθε παράθυρο, όλες οι τιμές θεωρούνται αξιόπιστες. Όσο προχωράει ο αλγόριθμος εύρεσης του συνολικού μέσου όρου, τότε ανάλογα υπολογίζονται οι τιμές αξιοπιστίας των αποτελεσμάτων κάθε βήματος.

EBAR

Στον εκτιμητή $EBAR$ χρησιμοποιείται η συνάρτηση $EBAR$, η οποία στηρίζεται σε κάποια ποσότητα b_i – ένα bit – για να καθορίσει την αξιοπιστία της κάθε τιμής που ανθροίζει.

Συγκεκριμένα, αν θ_1 και θ_2 τα υπόλοιπα φάσης δύο συμβόλων και b_1, b_2 τα bits αξιοπιστίας των, τότε

$$\hat{\theta} = EBAR(\theta_1, \theta_2, b_1, b_2) = [b_1\theta_1 + (1 - b_1)\theta_2] \oplus \frac{1}{2} \{ [b_2\theta_2 + (1 - b_2)\theta_1] \ominus [b_1\theta_1 + (1 - b_1)\theta_2] \} \quad (4.2)$$

και η συνάρτηση που επιστρέφει το bit αξιοπιστίας είναι

$$\hat{b} = f_b(\theta_1, \theta_2, b_1, b_2) = \begin{cases} 0 & \text{αν } (b_1 = 0 \text{ και } b_2 = 0) \text{ ή } (|\theta_2 - \theta_1| > \text{κατώφλι}) \\ 1 & \text{αλλιώς} \end{cases} \quad (4.3)$$

Αυτό στην ουσία που κάνει είναι να δηλώνει σαν αναξιόπιστους τους μέσους όρους οι οποίοι δημιουργήθηκαν από προηγούμενες τιμές που δεν ήταν αξιόπιστες ή έχουν διαφορά μεγαλύτερη του κατωφλιού μεταξύ τους. Ένα καλό κατώφλι για την QPSK είναι το $\frac{3\pi}{8}$ [1].

Ο εκτιμητής EBAR δουλεύει ικανοποιητικά για SNR της τάξεως των 6 dB για διαμόρφωση QPSK.

LBAR και QLBAR

Η κακή συμπεριφορά του EBAR σε $SNR < 6$ dB οδήγησε στην δημιουργία του αλγόριθμου LBAR και στην 'χβαντισμένη' του μορφή, τον QLBAR.

Στην LBAR και για ένα σύστημα mPSK, η αξιοπιστία κάθε τιμής θ_i είναι η f_i και βάση των τύπων παρακάτω βρίσκεται η νέα τιμή $\hat{\theta}$ και η αξιοπιστία της, η \hat{f} :

$$\hat{\theta} = LBAR(\theta_1, \theta_2, f_1, f_2) = \theta_1 \oplus \left[\frac{f_2}{f_1 + f_2} (\theta_2 \ominus \theta_1) \right] \quad (4.4)$$

και

$$\hat{f} = f(\theta_1, \theta_2) = 1 - \frac{m}{2\pi} |\theta_2 - \theta_1| \quad (4.5)$$

Το πεδίο τιμών της $f(\theta_1, \theta_2)$ είναι το $[0, 1]$. Διαισθητικά, όσο πιο 'κοντά' είναι οι θ_1 και θ_2 , τόσο πλησιάζει η $\hat{\theta}$ το 1 και άρα τόσο πιο αξιόπιστος είναι ο μέσος όρος.

Η ανάγκη όμως για περαιτέρω απλοποίηση της υλοποίησης οδήγησε στον QLBAR, όπου αντί να υπάρχει η συνεχής $f(\theta_1, \theta_2)$, επιλέχθηκε μια $f_{\text{discrete}}(\theta_1, \theta_2)$ που επιστρέφει 4 τιμές - που μπορούν να κωδικοποιηθούν με 2 bits. Η συνάρτηση για την περίπτωση της QPSK έχει την μορφή

$$f_{\text{discrete}}(\theta_1, \theta_2) = \begin{cases} 1 & \text{αν } |\theta_2 - \theta_1| < \frac{\pi}{8} \\ 0.5 & \text{αν } \frac{\pi}{8} \leq |\theta_2 - \theta_1| < \frac{\pi}{4} \\ 0.25 & \text{αν } \frac{\pi}{4} \leq |\theta_2 - \theta_1| < \frac{3\pi}{8} \\ 0 & \text{αλλιώς} \end{cases} \quad (4.6)$$

Η QLBAR ορίζεται σαν

$$\hat{\theta} = QLBAR(\theta_1, \theta_2, f_1, f_2) = \theta_1 \oplus [F(f_1, f_2)(\theta_2 \ominus \theta_1)] \quad (4.7)$$

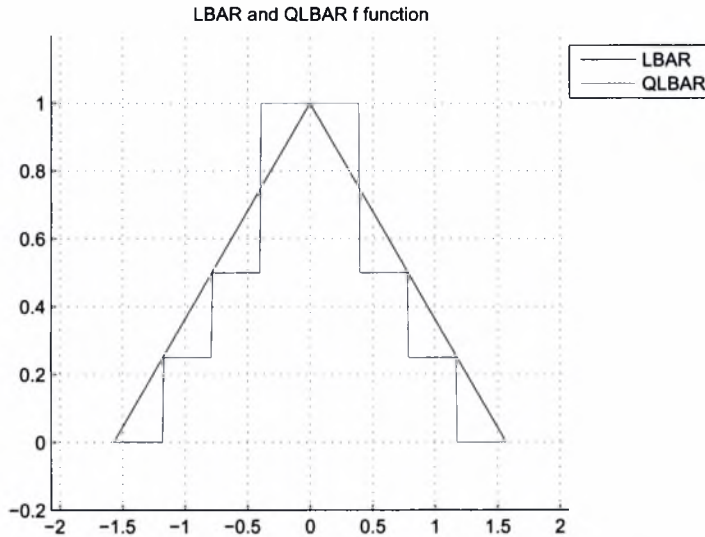
με

$$F(f_1, f_2) = \begin{cases} 0 & \text{αν } f_2 = 0 \\ 1 & \text{αν } f_1 = 0 \\ 0.25 & \text{αν } 0 < f_2 < f_1 \\ 0.5 & \text{αν } 0 < f_1 < f_2 \end{cases} \quad (4.8)$$

Είναι λογικό να υποθεθεί πως η χβάντιση των συναρτήσεων αξιοπιστίας βλάπτει την απόδοση του αλγόριθμου QLBAR σε σχέση με τον LBAR, γεγονός που αποδεικνύεται στο Κεφάλαιο 3 του [1]. Στο Σχήμα(4.3) εμφανίζεται η μορφή των συναρτήσεων αξιοπιστίας για τις LBAR και QLBAR.

4.5 Κυριότερες διαφορές εκτιμητών βαρυκέντρου και V&V

Αν και στηρίζονται στην ίδια φιλοσοφία, αυτή της εύρεσης του μέσου όρου του θορύβου φάσης από ένα παράθυρο κάποιου πλήθους συμβόλων, οι εκτιμητές βαρυκέντρου και V&V έχουν κάποιες σημαντικές διαφορές μεταξύ τους.



Σχήμα 4.3: Συναρτήσεις αξιοπιστίας *LBAR* και *QLBAR*

Η κυριότερη είναι πως ενώ οι εκτιμητές V&V εκτιμούν το θόρυβο φάσης για το κεντρικό σύμβολο του παραθύρου, οι εκτιμητές βαρυκέντρου εκτιμούν το θόρυβο φάσης για το πρώτο στοιχείο του παραθύρου. Από αυτό το γεγονός πηγάζει η απαίτηση να υπάρχουν $2N + 1$ σύμβολα διαθέσιμα για τους V&V, ενώ οι βαρυκέντρου αρκούνται σε $2N$ σύμβολα στο παράθυρό τους. Από την άλλη, η μέθοδος μέσου όρου στους εκτιμητές βαρυκέντρου επιβάλλει το παράθυρο να έχει μήκος δύναμης του 2, ενώ στους V&V το παράθυρο μπορεί να έχει οποιοδήποτε μήκος, αρκεί να είναι περιττό.

Τέλος, οι εκτιμητές βαρυκέντρου χρειάζονται μόνον τις τιμές του θορύβου φάσης. Συνεπώς, πρέπει να προηγηθεί αποδιαμόρφωση των συμβόλων. Από την άλλη, ο εκτιμητής V&V δουλεύει με τα αρχικά σύμβολα.

Κεφάλαιο 5

ΠΡΟΣΟΜΟΙΩΣΕΙΣ

5.1 Σχέση μεταξύ SNR και $\frac{E_b}{N_0}$ για QPSK

Σε σχετικές εργασίες, είναι πολύ συνηθισμένο να απαντάται ο λόγος $\frac{E_b}{N_0}$ αντί του SNR . Το N_0 είναι η φασματική πυκνότητα ισχύος θορύβου (*noise power spectral density*) και εκφράζει την ισχύ του θορύβου σε bandwidth 1 Hz. Εκφράζεται σε W/Hz.

Αν $w(t)$ ο λευκός θόρυβος τότε σύμφωνα με το θεώρημα ενέργειας Rayleigh

$$\mathbb{E} [w(t)^2] = \frac{E_s N_0}{2} \quad (5.1)$$

Έστω ότι $s(t)$ το σήμα χωρίς θόρυβο, τότε

$$\gamma = \frac{\mathbb{E} [s(t)^2]}{\mathbb{E} [w(t)^2]} \quad (5.2)$$

και

$$SNR = 10 \cdot \log_{10} (\gamma) \quad (5.3)$$

Προφανώς, η σχέση που συνδέει το SNR με το $\frac{E_b}{N_0}$ είναι

$$\gamma = \frac{\mathbb{E} [s(t)^2]}{\mathbb{E} [w(t)^2]} = \frac{E_s^2}{\frac{E_s N_0}{2}} \Rightarrow$$

$$\gamma = \frac{2E_s}{N_0} \quad \text{ή} \quad \frac{4E_b}{N_0} \quad (\text{QPSK}) \quad (5.4)$$

Συνεπώς,

$$SNR = 10 \cdot \log_{10} \left(\frac{4E_b}{N_0} \right) \quad (5.5)$$

$$\frac{E_b}{N_0} = \frac{1}{4} 10^{\left(\frac{SNR}{10}\right)} \quad (5.6)$$

SNR (dB)	0	1	2	3	5	6	8
$\frac{E_b}{N_0}$ (dB)	-6.0206	-5.0206	-4.0206	-3.0206	-1.0206	-0.0206	1.9794

Πίνακας 5.1: Τυπικές τιμές SNR και $\frac{E_b}{N_0}$

5.2 Παράμετροι προσομοίωσης

Στην παρούσα εργασία δημιουργήθηκε για τις προσομοιώσεις ένα μοντέλο ενός συστήματος QPSK με τα χαρακτηριστικά που παρουσιάζονται στον Πίνακα(5.2). Επιπλέον, δεν υπάρχει κωδικοποίηση στα σύμβολα και υπάρχει τέλειος συγχρονισμός ρυθμού.

$M = 4$	αριθμός συμβόλων (QPSK)
$f_c = 30 \cdot 10^9$	συχνότητα λειτουργίας στα 30 GHz
$n_c = 498$	
$E_b = 1$	ενέργεια bit
$E_s = 2E_b$	ενέργεια συμβόλου
$T = n_c/f_c$	περίοδος συμβόλου στα 166 us
$A = \sqrt{2E_s/T}$	πλάτος παλμού

Πίνακας 5.2: Παράμετροι προσομοίωσης

Κατά τις προσομοιώσεις υπάρχει η δυνατότητα επιλογής τιμών για τις εξής παραμέτρους:

- SNR : Όπως έχει αναφερθεί, στις δορυφορικές επικοινωνίες ο σηματοθορυβικός λόγος είναι κοντά στα 0 dB. Επιλέχθηκαν σαν τυπικές τιμές τα 0 dB για την χειρότερη περίπτωση, 4 dB για την μέση και 10 dB για την καλύτερη δυνατή.
- Δf : Το Δf είναι το υπόλοιπο της συχνότητας λόγω ατελούς αποδιαμόρφωσης. Ορίζεται σαν $(\Delta f)T \ll 1$, οπότε δεδομένου ότι $T = 166$ us, καλές τιμές μπορούν να θεωρηθούν $\Delta f = [0 \text{ Hz}, 100 \text{ Hz}, 10000 \text{ Hz}]$.
- θ : Ο θόρυβος φάσης είναι η κατεξοχήν κρίσιμη παράμετρος, αφού είναι και η τιμή που εκτιμάται. Λογικό σύνολο τιμών θα μπορούσε να θεωρηθεί το $\theta = [\frac{\pi}{100}, \frac{\pi}{10}, \frac{\pi}{5}]$. Οι αρνητικές τιμές θεωρήθηκε πλεονασμός να χρησιμοποιηθούν, αφού το μόνο που κάνουν είναι να περιστρέφουν το σύμβολο με αντίθετη φορά από τις θετικές. Οποιαδήποτε προσπάθεια για την εκτίμηση και την διόρθωση του θορύβου φάσης μεγαλύτερου του $\pm \frac{\pi}{4}$ για κάποιο QPSK σύστημα δεν έχει κάποιο πρακτικό νόημα, αφού τα σύμβολα ήδη έχουν καταστραφεί.
- Παράθυρο συμβόλων: Μια καλή τιμή για το παράθυρο συμβόλων που εμφανίζεται και στην σχετική βιβλιογραφία είναι 32 σύμβολα – στους εκτιμητές V&V είναι συνολικά 33 σύμβολα, αφού πρέπει να μετρηθεί και το σύμβολο για το οποίο εκτιμάται ο θόρυβος φάσης. Επιλέχθηκε $W = [32, 64]$, αφενός γιατί 128 σύμβολα θα επέβαλλαν μια πολύ περίπλοκη και δύσκολη υλοποίηση και αφετέρου να μπορεί να γίνει σύγκριση και με τους εκτιμητές βαρυκέντρου, επειδή δουλεύουν με αριθμό συμβόλων που είναι δυνάμεις του 2.

Παρόλα αυτά, υπάρχουν και προσομοιώσεις που ελέγχουν ένα μεγαλύτερο εύρος τιμών για μια πιο ολοκληρωμένη άποψη της λειτουργίας του εκάστοτε εκτιμητή.

5.3 Προσομοιώσεις των εκτιμητών V&V

Στις παρακάτω ενότητες μελετάται η συμπεριφορά του εκτιμητή V&V, με διαφορετική συνάρτηση $F(\rho) = (\rho)^n$. Παράλληλα, συγκρίνεται με τον βέλτιστο εκτιμητή V&V, όπως εκφράζεται από την (3.31) και με υλοποιήσεις του V&V πολλαπλών περασμάτων της Ενότητας 3.7.

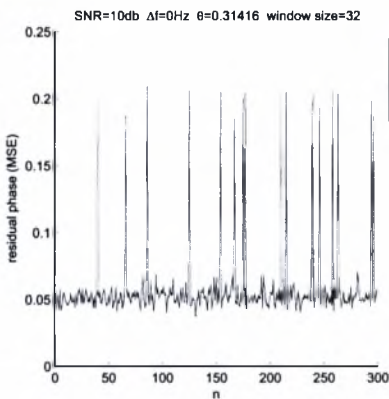
Σε κάθε ενότητα, εμφανίζονται τα αποτελέσματα των προσομοιώσεων κάτω από διαφορετικές συνθήκες και ακολουθούν συγκεκριμένες παρατηρήσεις και συμπεράσματα πάνω σε αυτές.

Οι προσομοιώσεις έχουν γραφτεί έτσι ώστε να καλύψουν το μεγαλύτερο δυνατό εύρος παραμέτρων λειτουργίας.

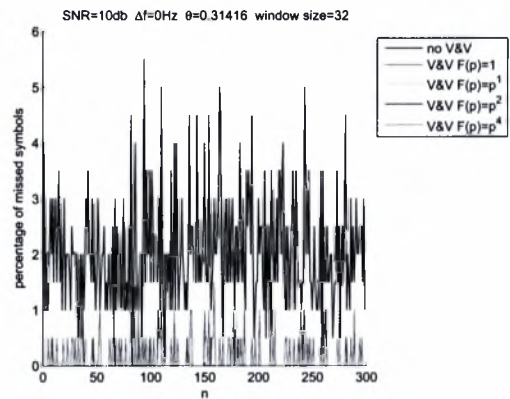
5.3.1 Μελέτη εκτιμητή V&V για διαφορετικό $F(\bullet)$

Προσομοίωση σε επίπεδο συμβόλων

Προσομοίωση του εκτιμητή V&V με $F(\rho) = (\rho)^n$, $n = [0, 1, 2, 4]$ σε επίπεδο συμβόλων, σε διαφορετικά περιβάλλοντα θορύβου.

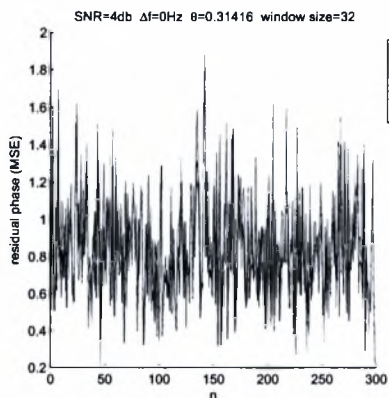


(α) Υπόλοιπο διαφοράς φάσης (MSE)

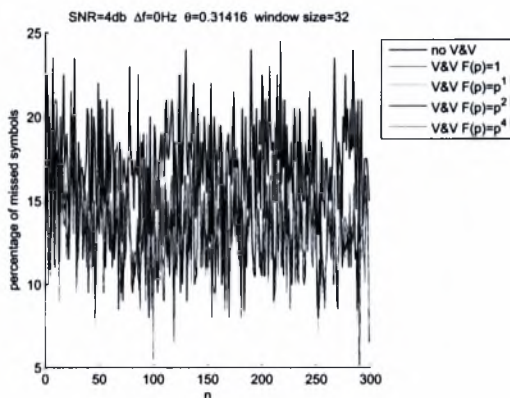


(β) Ποσοστό εσφαλμένων συμβόλων

Σχήμα 5.1: $SNR = 10 \text{ dB}$

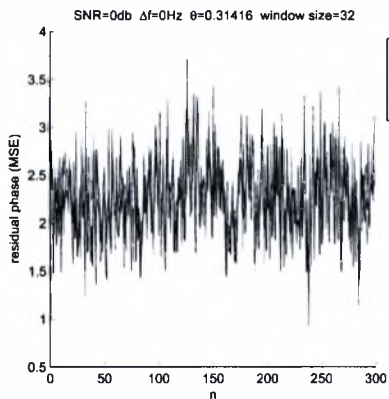


(α') Υπόλοιπο διαφοράς φάσης (MSE)

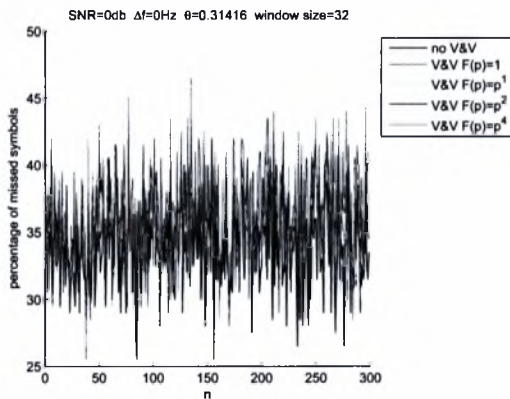


(β') Ποσοστό εσφαλμένων συμβόλων

Σχήμα 5.2: $SNR = 4 \text{ dB}$

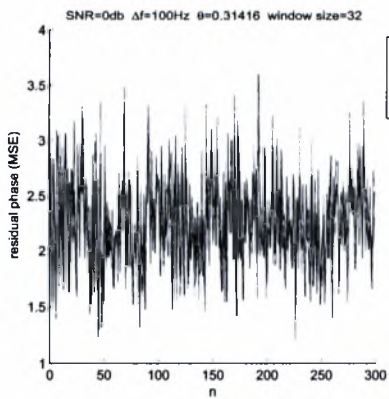


(α') Υπόλοιπο διαφοράς φάσης (MSE)

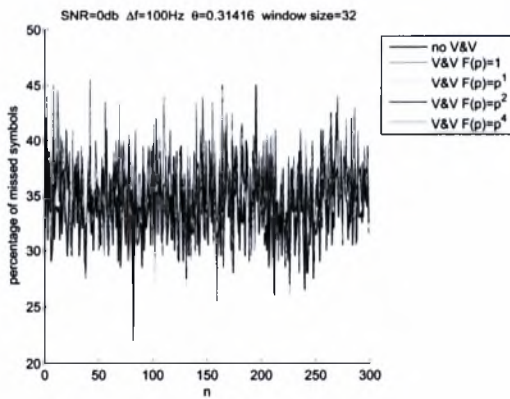


(β') Ποσοστό εσφαλμένων συμβόλων

Σχήμα 5.3: $SNR = 0 \text{ dB}$



(α') Υπόλοιπο διαφοράς φάσης (MSE)



(β') Ποσοστό εσφαλμένων συμβόλων

Σχήμα 5.4: $SNR = 0 \text{ dB}$ και μικρό Δf

Αποτέλεσμα εκτέλεσης για $SNR = 10 \text{ dB}$ – Σχήμα(5.1)

SNR: 10.00dB Phase error: 18.00 Df: 0.00Hz
 Residual phase MSE without V&V: 0.245326
 Residual phase MSE of V&V $F(p) = 1$: 0.058585
 Residual phase MSE of V&V $F(p) = p^1$: 0.058686
 Residual phase MSE of V&V $F(p) = p^2$: 0.057410
 Residual phase MSE of V&V $F(p) = p^4$: 0.059068
 Missed symbols without V&V: 2.058%
 Missed symbols of V&V $F(p) = 1$: 0.172%
 Missed symbols of V&V $F(p) = p^1$: 0.178%
 Missed symbols of V&V $F(p) = p^2$: 0.177%
 Missed symbols of V&V $F(p) = p^4$: 0.197%

Αποτέλεσμα εκτέλεσης για $SNR = 4 \text{ dB}$ – Σχήμα(5.2)

SNR: 4.00dB Phase error: 18.00 Df: 0.00Hz
 Residual phase MSE without V&V: 1.092987
 Residual phase MSE of V&V $F(p) = 1$: 0.849248
 Residual phase MSE of V&V $F(p) = p^1$: 0.822336
 Residual phase MSE of V&V $F(p) = p^2$: 0.820606
 Residual phase MSE of V&V $F(p) = p^4$: 0.863966
 Missed symbols without V&V: 17.468%
 Missed symbols of V&V $F(p) = 1$: 13.172%
 Missed symbols of V&V $F(p) = p^1$: 12.678%
 Missed symbols of V&V $F(p) = p^2$: 12.705%
 Missed symbols of V&V $F(p) = p^4$: 13.377%

Αποτέλεσμα εκτέλεσης για $SNR = 0 \text{ dB}$ – Σχήμα(5.3)

SNR: 0.00dB Phase error: 18.00 Df: 0.00Hz
 Residual phase MSE without V&V: 2.168451
 Residual phase MSE of V&V $F(p) = 1$: 2.309640
 Residual phase MSE of V&V $F(p) = p^1$: 2.249578
 Residual phase MSE of V&V $F(p) = p^2$: 2.244076
 Residual phase MSE of V&V $F(p) = p^4$: 2.277366
 Missed symbols without V&V: 33.767%
 Missed symbols of V&V $F(p) = 1$: 36.050%
 Missed symbols of V&V $F(p) = p^1$: 34.942%
 Missed symbols of V&V $F(p) = p^2$: 34.888%
 Missed symbols of V&V $F(p) = p^4$: 35.347%

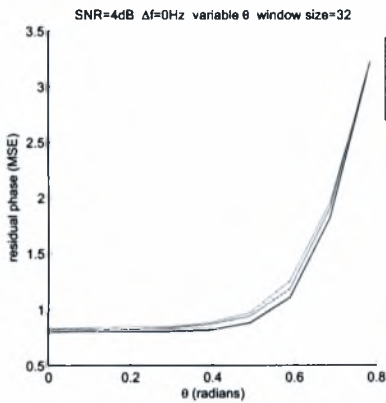
Αποτέλεσμα εκτέλεσης για $SNR = 0 \text{ dB}$ και $\Delta f = 100 \text{ Hz}$ – Σχήμα(5.4)

SNR: 0.00db Phase error: 18.00 Df: 100.00Hz
 Residual phase MSE without V&V: 2.177562
 Residual phase MSE of V&V $F(p) = 1$: 2.293448
 Residual phase MSE of V&V $F(p) = p^1$: 2.234389
 Residual phase MSE of V&V $F(p) = p^2$: 2.222855
 Residual phase MSE of V&V $F(p) = p^4$: 2.271416
 Missed symbols without V&V: 33.723%
 Missed symbols of V&V $F(p) = 1$: 36.173%
 Missed symbols of V&V $F(p) = p^1$: 35.212%
 Missed symbols of V&V $F(p) = p^2$: 34.998%
 Missed symbols of V&V $F(p) = p^4$: 35.442%

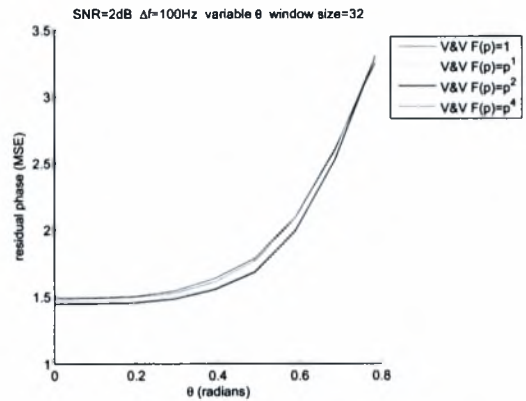
Προσομοίωση για μεταβλητό θόρυβο φάσης

Ακολουθούν προσομοιώσεις για μεταβλητό θόρυβο φάσης θ . Παρουσιάζονται τόσο ως προς το ποσοστό κακώς αποδιαμορφωμένων συμβόλων μετά την διόρθωση, αλλά και ως προς το υπόλοιπο διαφοράς φάσης.

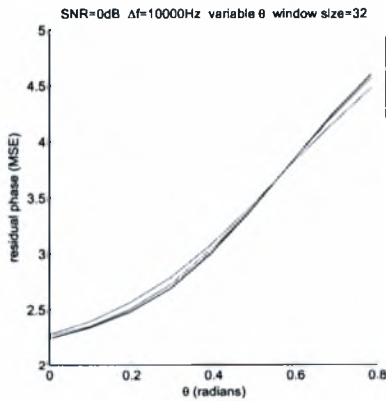
Το υπόλοιπο διαφοράς φάσης είναι ουσιαστικά το μέσο τετραγωνικό σφάλμα (MSE) της διαφοράς της φάσης του ιδανικού σήματος από το λαμβανόμενο, αφού έχει γίνει στο τελευταίο διόρθωση φάσης. Το ποσοστό των λάθος αποδιαμορφωμένων συμβόλων είναι έκφραση του πόσα σύμβολα αποφασίστηκαν λάθος μετά την διόρθωση του θορύβου φάσης.



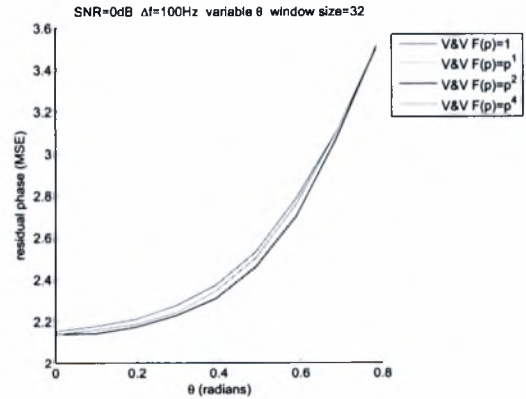
(α') $SNR = 4\text{ dB}$, $\Delta f = 0\text{ Hz}$



(β') $SNR = 2\text{ dB}$, $\Delta f = 100\text{ Hz}$

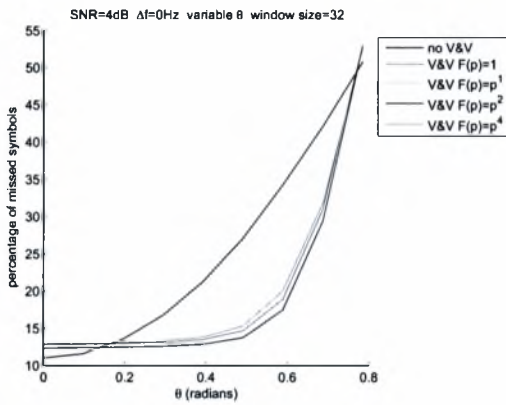


(γ') $SNR = 0\text{ dB}$, $\Delta f = 10000\text{ Hz}$

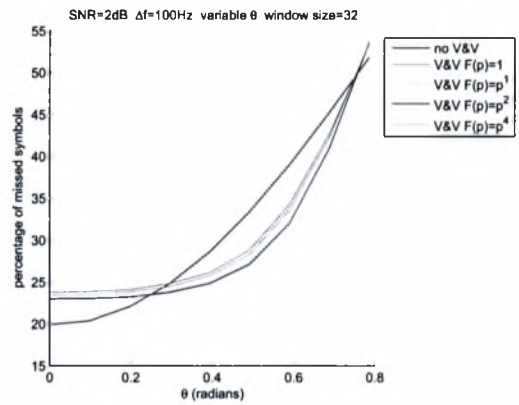


(δ') $SNR = 0\text{ dB}$, $\Delta f = 100\text{ Hz}$

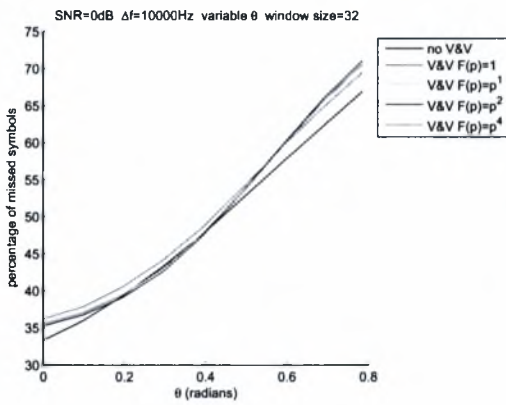
Σχήμα 5.5: Υπόλοιπο διαφοράς φάσης



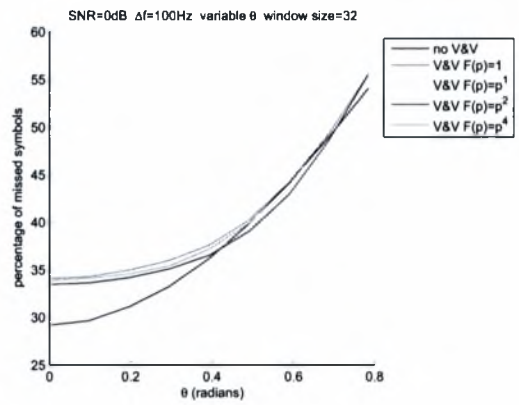
(α) $SNR = 4 \text{ dB}, \Delta f = 0 \text{ Hz}$



(β) $SNR = 2 \text{ dB}, \Delta f = 100 \text{ Hz}$



(γ) $SNR = 0 \text{ dB}, \Delta f = 10000 \text{ Hz}$

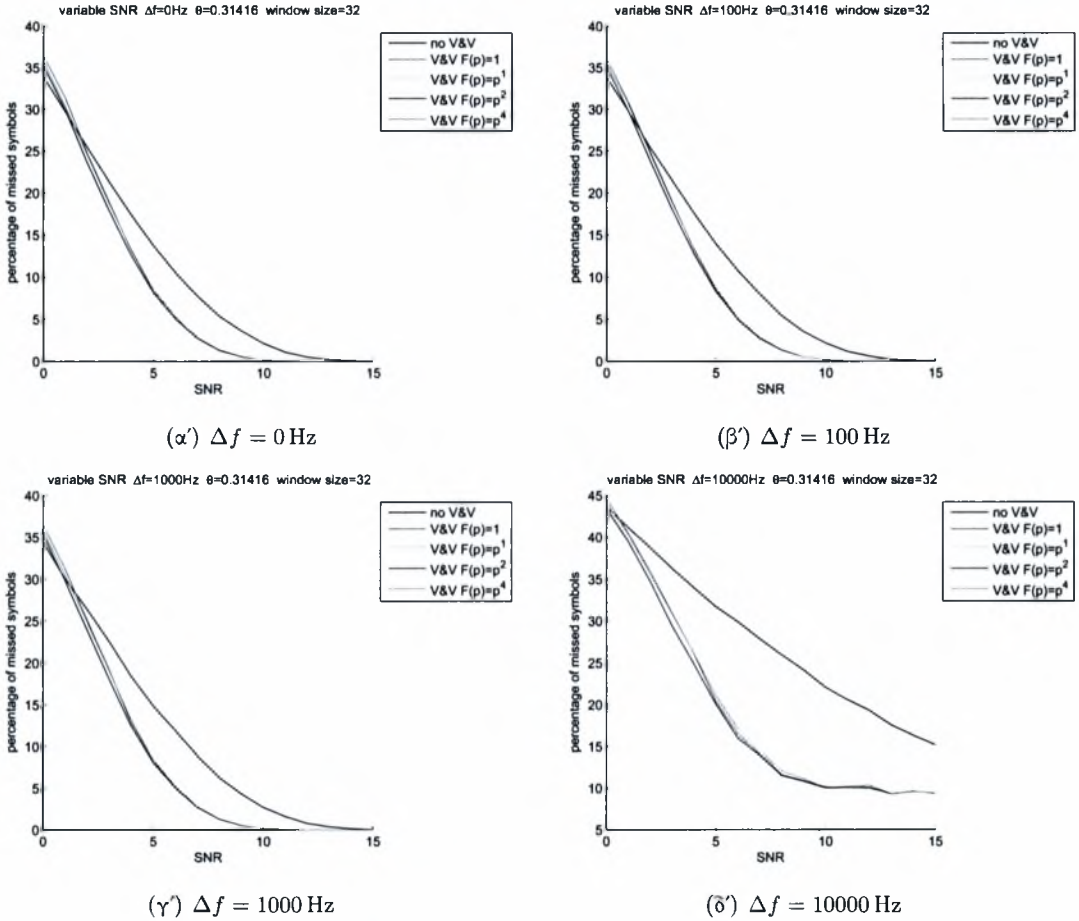


(δ) $SNR = 0 \text{ dB}, \Delta f = 100 \text{ Hz}$

Σχήμα 5.6: Ποσοστό εσφαλμένων συμβόλων

Προσομοίωση για $SNR \in [0 \text{ dB}, 15 \text{ dB}]$

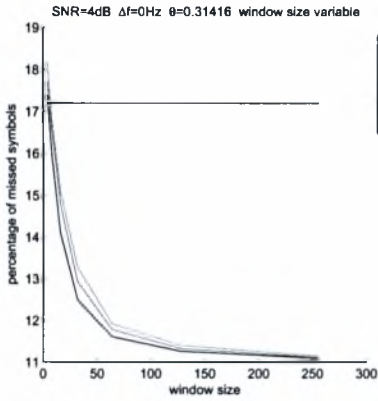
Σε αυτήν την υποενότητα παρουσιάζονται προσομοιώσεις για αυξανόμενο SNR και η επίπτωσή του στο ποσοστό των λάθος αποδιαμορφωμένων συμβόλων.



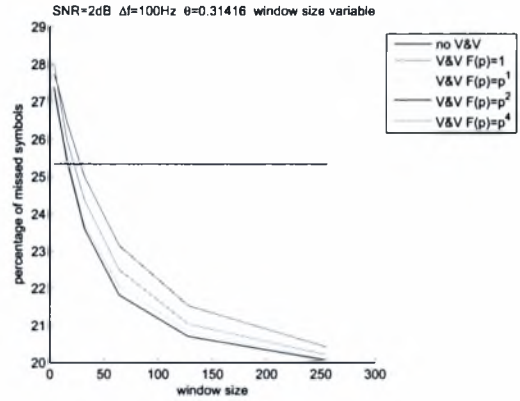
Σχήμα 5.7: Ποσοστό εσφαλμένων συμβόλων

Προσομοίωση για μεταβλητό μήκος παραθύρου

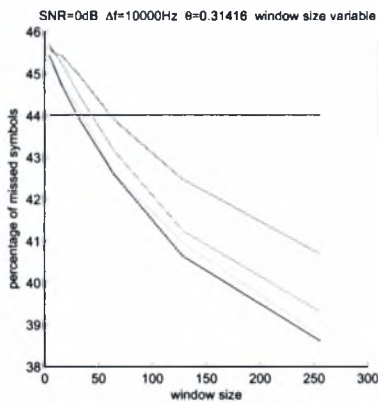
Προσομοιώσεις για την επίπτωση στην ακρίβεια της εκτίμησης που έχουν διαφορετικά μήκη παραθύρων συμβόλων, κάτω από την επίδραση διαφορετικής έντασης θορύβου.



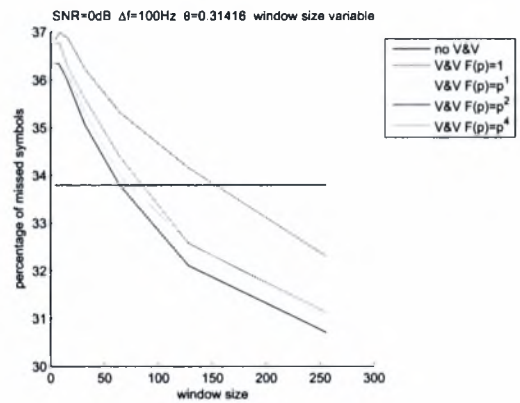
(α) $SNR = 4 \text{ dB}$, $\Delta f = 0 \text{ Hz}$



(β) $SNR = 2 \text{ dB}$, $\Delta f = 100 \text{ Hz}$



(γ) $SNR = 0 \text{ dB}$, $\Delta f = 10000 \text{ Hz}$

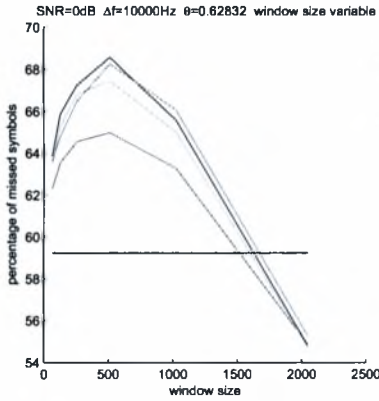


(δ) $SNR = 0 \text{ dB}$, $\Delta f = 100 \text{ Hz}$

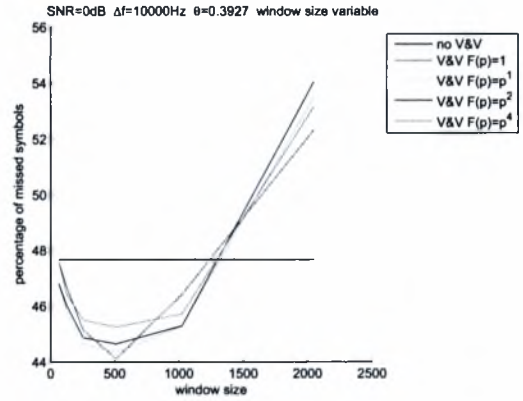
Σχήμα 5.8: Ποσοστό εσφαλμένων συμβόλων

Προσομοίωση σε συνθήκες υψηλού θορύβου

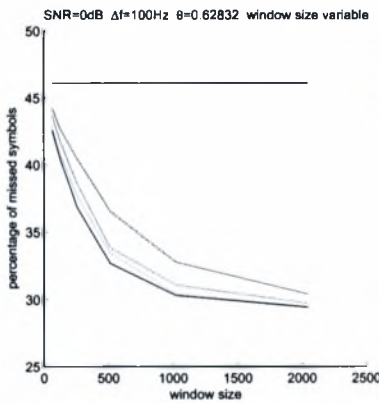
Ακολουθούν προσομοιώσεις σε υψηλά επίπεδα θορύβου, για την πλήρη μελέτη της αποτελεσματικότητας μεγαλύτερων παραθύρων στους εκτιμητές V&V.



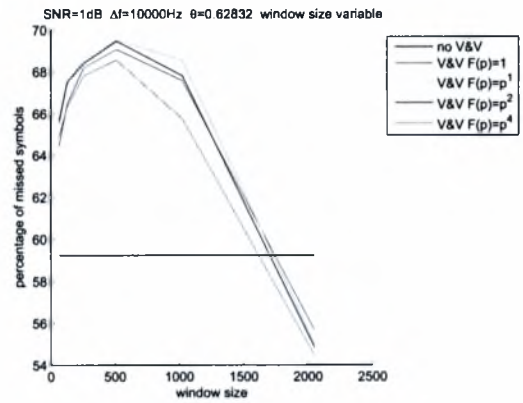
(α') $SNR = 0\text{ dB}$, $\Delta f = 10000\text{ Hz}$, $\theta = \frac{\pi}{5}$



(β') $SNR = 0\text{ dB}$, $\Delta f = 10000\text{ Hz}$, $\theta = \frac{\pi}{8}$



(γ') $SNR = 0\text{ dB}$, $\Delta f = 100\text{ Hz}$, $\theta = \frac{\pi}{5}$



(δ') $SNR = 1\text{ dB}$, $\Delta f = 10000\text{ Hz}$, $\theta = \frac{\pi}{5}$

Σχήμα 5.9: Ποσοστό εσφαλμένων συμβόλων για παράθυρα [64, 128, 256, 512, 1024, 2048] συμβόλων

Παρατηρήσεις

Επίδραση του n Η καλύτερη συνολικά λύση για υλοποίηση του εκτιμητή V&V είναι για $F(\rho) = \rho^2$. Από τα παραπάνω, παρατηρείται το φαινόμενο πως όσο μεγαλύτερο είναι το SNR , τόσο καλύτερη εκτίμηση κάνουν τα μικρότερα n . Παρόλα αυτά, μια καλή επιλογή για κάθε περίπτωση είναι η $n = 2$. Δουλεύει ικανοποιητικά σε σχέση με άλλα n , τόσο σε χαμηλό SNR , όσο και σε υψηλότερα. Στις ακραίες περιπτώσεις θορύβου, όπως στο Σχήμα(5.9), είναι δύσκολο να επιλεγεί ένα βέλτιστο n . Ούτως ή άλλως, η απόδοση του εκτιμητή σε τέτοιες συνθήκες είναι απογοητευτική.

Επίδραση του θορύβου του καναλιού και του θορύβου φάσης Είναι εμφανές πως ο εκτιμητής δεν δουλεύει καλά σε SNR που πλησιάζει τα 0 dB. Μάλιστα, σε περιπτώσεις υψηλού θορύβου καναλιού ή/και υψηλού θορύβου φάσης, είναι προτιμότερο να μην χρησιμοποιηθεί κάποιος εκτιμητής V&V, εκτός και αν υπάρχει δυνατότητα υποστήριξης μεγάλων μηκών παραθύρου. Σε ακραίες περιπτώσεις θορύβου – υψηλός θόρυβος καναλιού, υψηλός θόρυβος φάσης – ο εκτιμητής δεν μπορεί να καταφέρει κάτι καλύτερο από μια βελτίωση του 0.5% – 1% στο σύνολο των σωστά αποδιαμορφωμένων συμβόλων για παράθυρο μεγέθους 32 συμβόλων.

Είναι προφανές πως όταν ο θόρυβος του καναλιού υπερिσχύει του θορύβου φάσης, η εκτίμηση να μην είναι πολύ καλή. Είναι αποτέλεσμα του γεγονότος πως ο θόρυβος του καναλιού είναι τυχαίος και ως εκ τούτου αδύνατο έστω να εκτιμηθεί. Σε αυτό συνηγορεί το φαινόμενο να υπάρχει κακή εκτίμηση στην περίπτωση μικρού θορύβου φάσης, δηλαδή για παράδειγμα $< \frac{\pi}{16}$ με $SNR = 2$ dB.

Όταν ο θόρυβος φάσης πλησιάζει τα $\pm \frac{\pi}{4}$, τότε το ποσοστό των λάθος αποδιαμορφωμένων συμβόλων αγγίζει το 70% σε μερικές περιπτώσεις. Αν αποφασιζόταν τυχαία για το ποιο σύμβολο έλαβε το σύστημα, τότε το ποσοστό λάθους είναι 75%.

Επίδραση του Δf Κατά τις προσομοιώσεις με διάφορα Δf , παρατηρείται το φαινόμενο σε SNR κοντά στο 4–5 dB να βελτιώνεται η εκτίμηση ασχέτως του n , όταν το Δf είναι της τάξης των 100 Hz. Στην γενική περίπτωση πάντως, είναι εμφανές πως η ύπαρξη κάποιας διαφοράς στην συχνότητα Δf , υποβαθμίζει το σήμα και επιδρά αρνητικά στην σωστή αποδιαμόρφωση των συμβόλων. Μάλιστα, όταν το $(\Delta f)T$ είναι της τάξης του 10^{-3} , δηλαδή για το προσομοιούμενο σύστημα $\Delta f = 10000$ Hz, επιβάλλει ισχυρό θόρυβο στα σύμβολα, σε σημείο που οι δύο τάξεις μεγέθους μεταξύ του $\Delta f = 100$ Hz και $\Delta f = 10000$ Hz, να έχουν μεγάλη διαφορά, με τα μεγαλύτερα λάθη στην αποδιαμόρφωση να εμφανίζονται για το δεύτερο.

Επίδραση του παραθύρου Όσο αυξάνεται το παράθυρο, τόσο αυξάνεται και η ακρίβεια της εκτίμησης. Φυσικά, το καλύτερο παράθυρο είναι αυτό που έχει όσο το δυνατόν μεγαλύτερο πλήθος συμβόλων. Παρόλα αυτά, λόγοι ταχύτητας όχι μόνον της προσομοίωσης, αλλά και της τυχόν υλοποίησης, όπως και η ανάγκη για ελαχιστοποίηση της απαιτούμενης μνήμης και επεξεργαστικής ισχύος στα on board συστήματα των δορυφόρων, επιβάλλουν κάποιο παράθυρο 32, 64 ή 128 συμβόλων. Τέλος, παρατηρήθηκε το φαινόμενο όπου όταν ο θόρυβος φάσης είναι στα $\frac{\pi}{5}$, με $SNR = 0$ dB και $\Delta f = 10000$ Hz, τα πολύ μεγάλα παράθυρα των 1024 ή των 2048 συμβόλων να επιδεικνύουν χειρότερη συμπεριφορά σε σχέση με μικρότερα παράθυρα. Είναι ένα ακόμα αποτέλεσμα του τυχαίου θορύβου.

Συμπεράσματα

Από τα Σχήμα(5.5) και Σχήμα(5.6) είναι εμφανές πως για τις ίδιες συνθήκες, τα διαγράμματα που εμφανίζουν το ποσοστό των λάθος αποδιαμορφωμένων συμβόλων και του υπολοίπου φάσης ταυτίζονται απόλυτα. Στην παρούσα εργασία υπάρχει μια σαφής προτίμηση προς την πρώτη μορφή διαγράμματος που μπορεί να απεικονίσει καλύτερα τι γίνεται και στην περίπτωση που δεν υπάρχει κάποια διόρθωση στην φάση.

Ο εκτιμητής V&V είναι προβληματικός σε $SNR = 0\text{dB}$. Ο αριθμός των συμβόλων που εν τέλει δεν αποδιαμορφώνονται σωστά, μετά την εκτίμηση και την διόρθωση της φάσης, είναι αρκετά μεγάλος. Είναι σαφές πως ο εκτιμητής δεν είναι και τόσο κατάλληλος για τέτοιες εφαρμογές.

Τα χαμηλά SNR εμφανίζονται να είναι η αχίλλειος πτέρνα των εκτιμητών V&V. Μάλιστα η αύξηση των λάθος αποδιαμορφωμένων συμβόλων είναι εκθετική ως προς την μείωση του SNR . Για αυτό το γεγονός είναι υπεύθυνο το γεγονός ότι οι εκτιμητές V&V προσπαθούν να εκτιμήσουν τον θόρυβο φάσης, που στις προσομοιώσεις θεωρείται κάποια σταθερή ποσότητα και όχι την συμπεριφορά του τυχαίου θορύβου του καναλιού. Ιδανικά, αν το μήκος του παραθύρου έτεινε στο $+\infty$, ο μέσος όρος του θορύβου φάσης που θα εκτιμόταν θα ήταν 0 – η μέση τιμή του γκαουσιανού θορύβου. Ο πεπερασμένος αριθμός των συμβόλων στο παράθυρο όμως οδηγεί σε λάθος υπολογισμό της μέσης τιμής.

Το $(\Delta f)T$ είναι μια σημαντική παράμετρος που δυσχεραίνει την αποδιαμόρφωση. Ιδίως στην περίπτωση που είναι της τάξης του 10^{-3} και μεγαλύτερο, δημιουργεί περισσότερα προβλήματα και από τον ίδιο τον θόρυβο φάσης. Γενικά, εμφανίζεται η τάση του εκτιμητή να είναι μεροληπτικός και να λαμβάνει αρκετά υπόψη την παρουσία του Δf .

Ενδιαφέρον παρουσιάζει η συμπεριφορά του εκτιμητή σε συνθήκες υψηλού θορύβου. Επαληθεύονται τα συμπεράσματα στα [1],[2]. Μάλιστα, εφόσον στην παρούσα εργασία ελέγχεται και η συμπεριφορά με $\Delta f \gg 0\text{Hz}$, υπάρχει μια πιο ολοκληρωμένη εικόνα, που υποδηλώνει την αδυναμία της λειτουργίας σε πολύ υψηλό θόρυβο και την ανάγκη για καλύτερο συγχρονισμού συχνότητας και αύξησης του SNR (έλεγχος ισχύος, φιλτράρισμα).

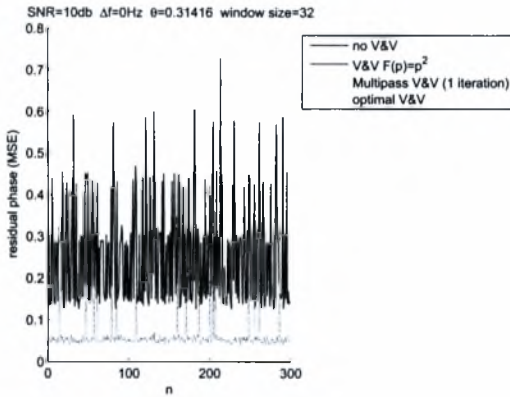
Στις προσομοιώσεις που ακολουθούν, το $n = 2$ θα θεωρείται η βέλτιστη επιλογή για τον εκτιμητή V&V, όπως επίσης και το μέγεθος παραθύρου στα 32 σύμβολα. Στην αντίστοιχη βιβλιογραφία, επιλέγεται το ίδιο παράθυρο, κυρίως για λόγους απλοποίησης της υλοποίησης και ταχύτητας του αλγορίθμου.

5.3.2 Μελέτη βέλτιστου εκτιμητή V&V και εκτιμητή V&V πολλαπλών περασμάτων

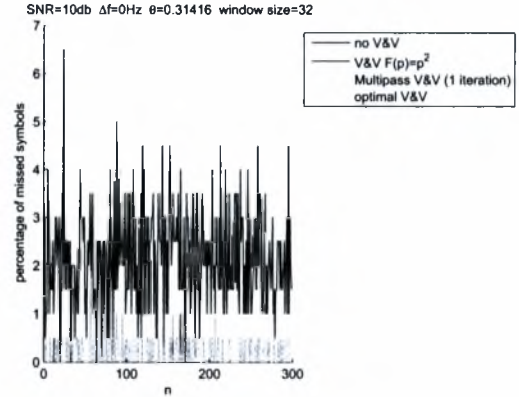
Στο Κεφάλαιο 3 παρουσιάζονται ο βέλτιστος εκτιμητής V&V (Εξίσωση (3.31)) και ο εκτιμητής V&V πολλαπλών περασμάτων (Ενότητα 3.7). Παρακάτω ακολουθούν προσομοιώσεις των εν λόγω εκτιμητών, αλλά και σύγκρισή τους με τον καλύτερο απλό V&V, με συνάρτηση $F(\rho) = (\rho)^2$.

Προσομοίωση σε επίπεδο συμβόλων

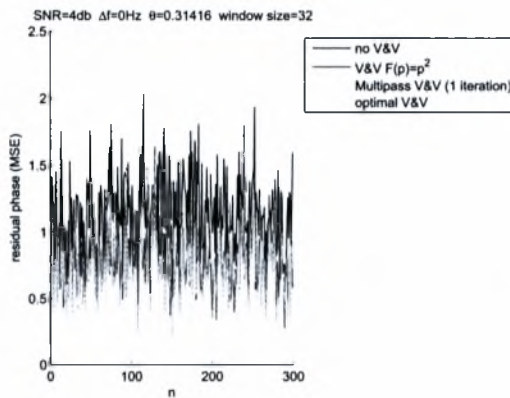
Προσομοιώσεις σε επίπεδο συμβόλων, όπως ακριβώς έγιναν και στην αρχή της προηγούμενης ενότητας.



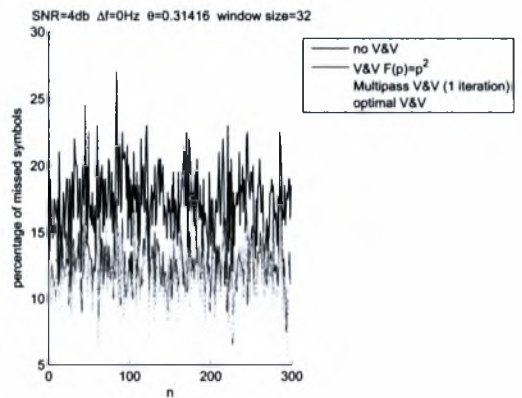
(α) Υπόλοιπο διαφοράς φάσης (MSE)



(β') Ποσοστό εσφαλμένων συμβόλων

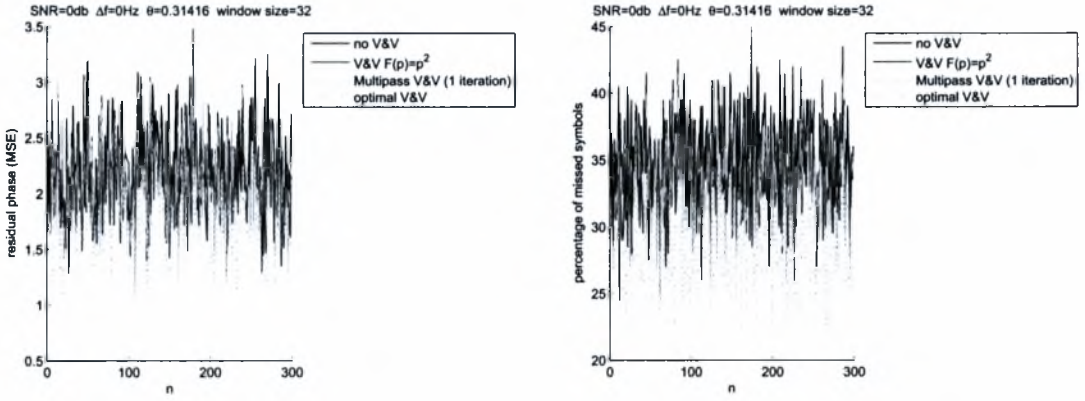
Σχήμα 5.10: $SNR = 10\text{ dB}$ 

(α') Υπόλοιπο διαφοράς φάσης (MSE)



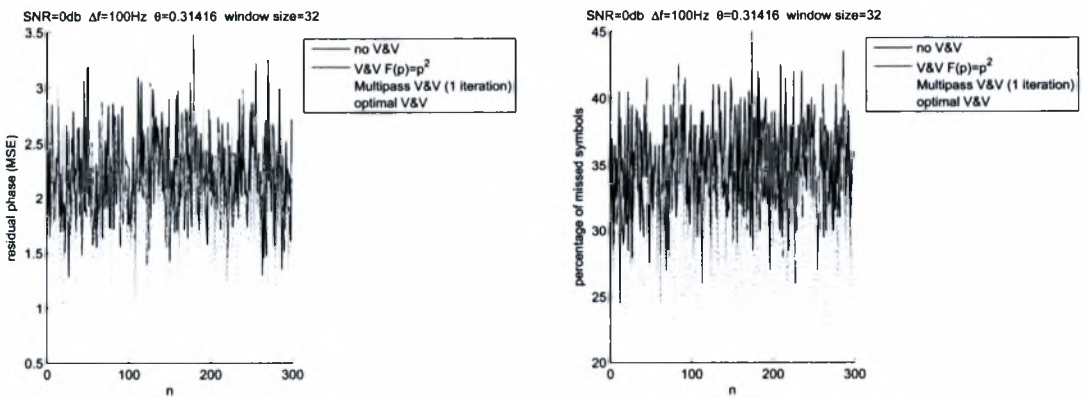
(β') Ποσοστό εσφαλμένων συμβόλων

Σχήμα 5.11: $SNR = 4\text{ dB}$



(α') Υπόλοιπο διαφοράς φάσης (MSE)

(β') Ποσοστό εσφαλμένων συμβόλων

Σχήμα 5.12: $SNR = 0$ dB

(α') Υπόλοιπο διαφοράς φάσης (MSE)

(β') Ποσοστό εσφαλμένων συμβόλων

Σχήμα 5.13: $SNR = 0$ dB και μικρό Δf

Αποτέλεσμα εκτέλεσης για $SNR = 10\text{ dB}$ – Σχήμα(5.10)

SNR: 10.00db Phase error: 18.00 Df: 0.00Hz
 Residual phase MSE without V&V: 0.255898
 Residual phase MSE of optimal V&V: 0.057256
 Residual phase MSE of V&V $F(p) = p^2$: 0.060092
 Residual phase MSE of Multipass V&V $F(p) = p^2$: 0.059713
 Missed symbols without V&V: 2.150%
 Missed symbols of optimal V&V: 0.145%
 Missed symbols of V&V $F(p) = p^2$: 0.188%
 Missed symbols of Multipass V&V $F(p) = p^2$: 0.192%

Αποτέλεσμα εκτέλεσης για $SNR = 4\text{ dB}$ – Σχήμα(5.11)

SNR: 4.00db Phase error: 18.00 Df: 0.00Hz
 Residual phase MSE without V&V: 1.126331
 Residual phase MSE of optimal V&V: 0.706760
 Residual phase MSE of V&V $F(p) = p^2$: 0.816445
 Residual phase MSE of Multipass V&V $F(p) = p^2$: 0.785526
 Missed symbols without V&V: 17.452%
 Missed symbols of optimal V&V: 10.510%
 Missed symbols of V&V $F(p) = p^2$: 12.570%
 Missed symbols of Multipass V&V $F(p) = p^2$: 11.998%

Αποτέλεσμα εκτέλεσης για $SNR = 0\text{ dB}$ – Σχήμα(5.12)

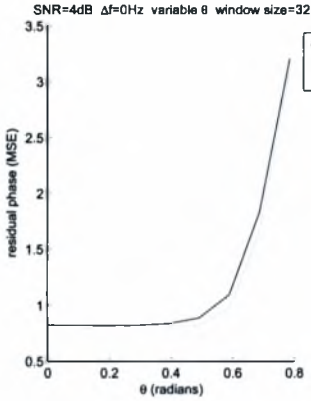
SNR: 0.00db Phase error: 18.00 Df: 0.00Hz
 Residual phase MSE without V&V: 2.152983
 Residual phase MSE of optimal V&V: 1.862181
 Residual phase MSE of V&V $F(p) = p^2$: 2.248334
 Residual phase MSE of Multipass V&V $F(p) = p^2$: 2.112168
 Missed symbols without V&V: 33.725%
 Missed symbols of optimal V&V: 28.695%
 Missed symbols of V&V $F(p) = p^2$: 35.660%
 Missed symbols of Multipass V&V $F(p) = p^2$: 33.342%

Αποτέλεσμα εκτέλεσης για $SNR = 0\text{ dB}$ και $\Delta f = 100\text{ Hz}$ – Σχήμα(5.13)

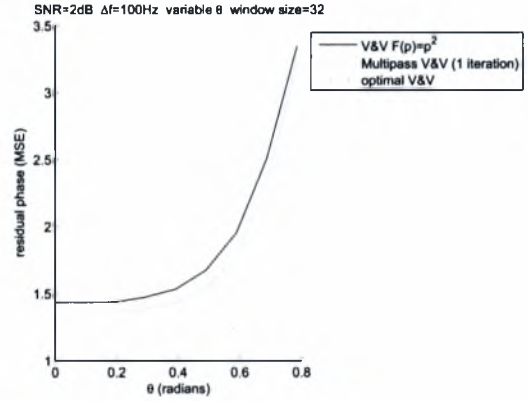
SNR: 0.00db Phase error: 18.00 Df: 100.00Hz
 Residual phase MSE without V&V: 2.156145
 Residual phase MSE of optimal V&V: 1.862586
 Residual phase MSE of V&V $F(p) = p^2$: 2.248507
 Residual phase MSE of Multipass V&V $F(p) = p^2$: 2.113142
 Missed symbols without V&V: 33.778%
 Missed symbols of optimal V&V: 28.687%
 Missed symbols of V&V $F(p) = p^2$: 35.682%
 Missed symbols of Multipass V&V $F(p) = p^2$: 33.358%

Προσομοίωση για μεταβλητό θόρυβο φάσης

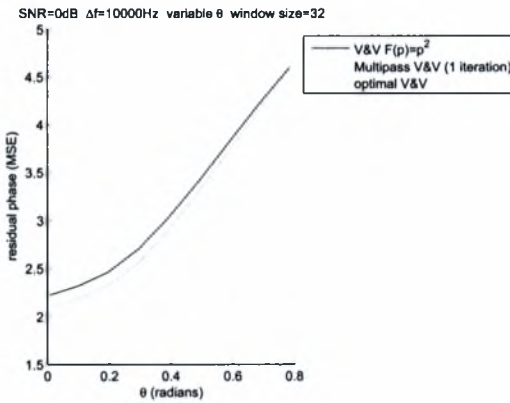
Προσομοιώσεις του απλού V&V, του βέλτιστου V&V και V&V πολλαπλών περασμάτων για αυξανόμενο θόρυβο φάσης.



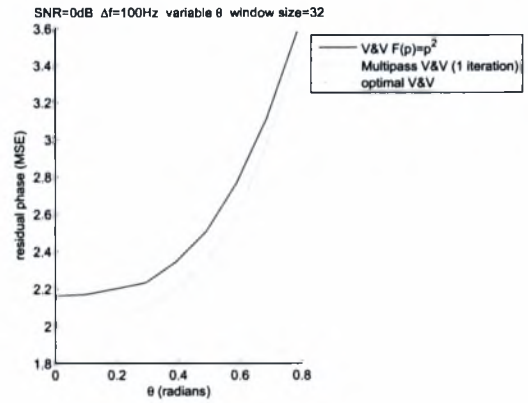
(α) $SNR = 4\text{ dB}$, $\Delta f = 0\text{ Hz}$



(β) $SNR = 2\text{ dB}$, $\Delta f = 100\text{ Hz}$

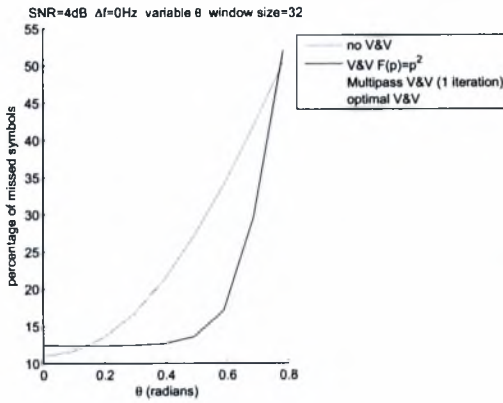


(γ) $SNR = 0\text{ dB}$, $\Delta f = 10000\text{ Hz}$

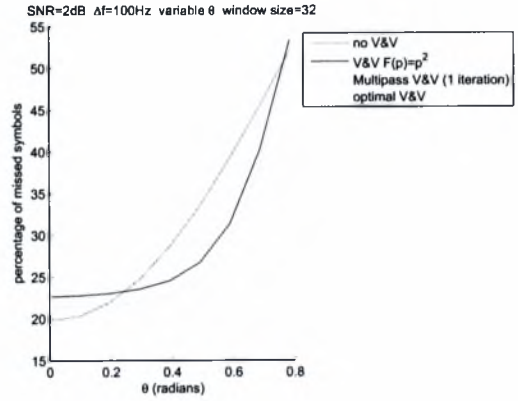


(δ) $SNR = 0\text{ dB}$, $\Delta f = 100\text{ Hz}$

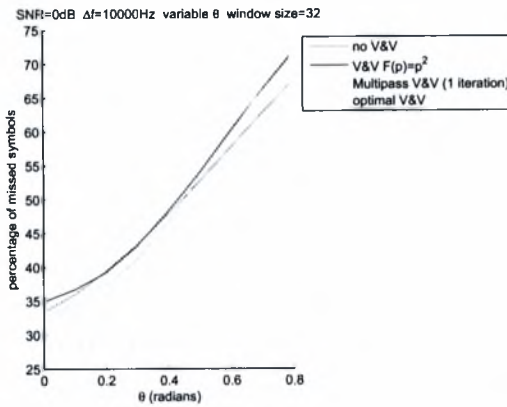
Σχήμα 5.14: Υπόλοιπο διαφοράς φάσης



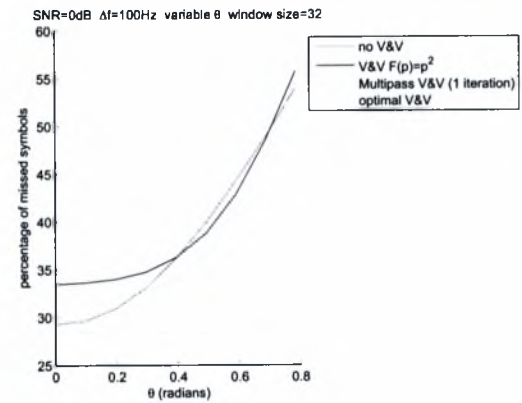
(α) $SNR = 4 \text{ dB}$, $\Delta f = 0 \text{ Hz}$



(β) $SNR = 2 \text{ dB}$, $\Delta f = 100 \text{ Hz}$



(γ) $SNR = 0 \text{ dB}$, $\Delta f = 10000 \text{ Hz}$



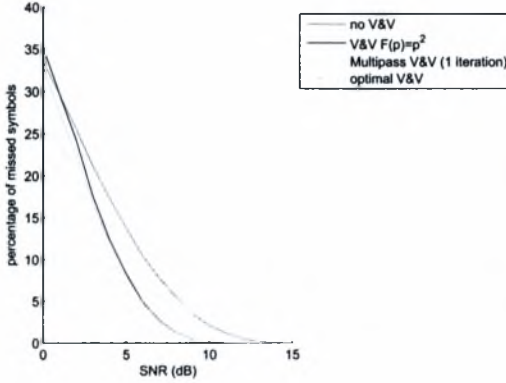
(δ) $SNR = 0 \text{ dB}$, $\Delta f = 100 \text{ Hz}$

Σχήμα 5.15: Ποσοστό εσφαλμένων συμβόλων

Προσομοίωση για $SNR \in [0 \text{ dB}, 15 \text{ dB}]$

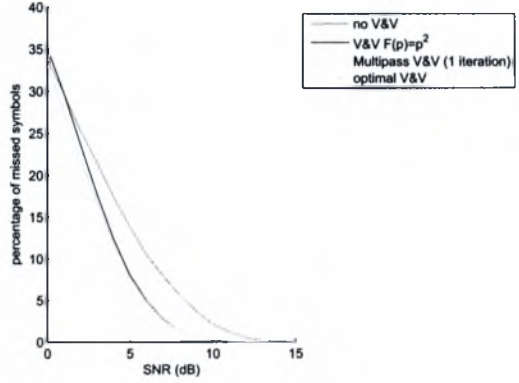
Όπως ακριβώς και στην περίπτωση του απλού V&V.

variable SNR $\Delta f=0\text{Hz}$ $\theta=0.31416$ window size=32



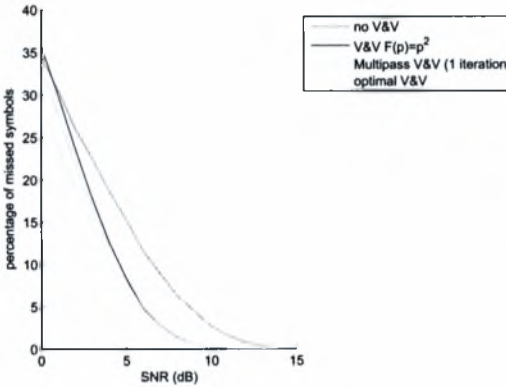
(α') $\Delta f = 0 \text{ Hz}$

variable SNR $\Delta f=100\text{Hz}$ $\theta=0.31416$ window size=32



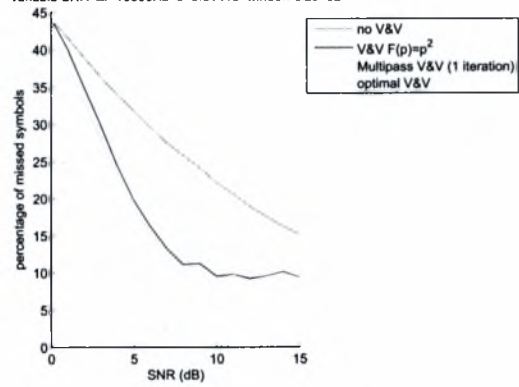
(β') $\Delta f = 100 \text{ Hz}$

variable SNR $\Delta f=1000\text{Hz}$ $\theta=0.31416$ window size=32



(γ') $\Delta f = 1000 \text{ Hz}$

variable SNR $\Delta f=10000\text{Hz}$ $\theta=0.31416$ window size=32



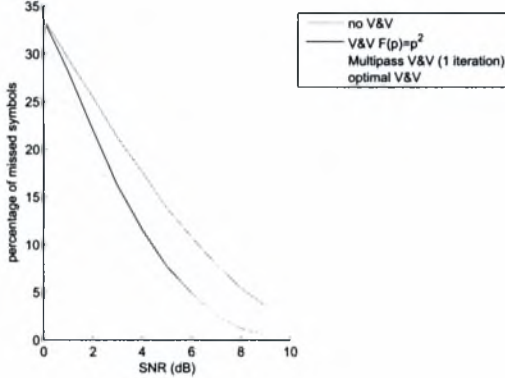
(δ') $\Delta f = 10000 \text{ Hz}$

Σχήμα 5.16: Ποσοστό εσφαλμένων συμβόλων

Προσομοίωση για μεγαλύτερα μήκη παραθύρου

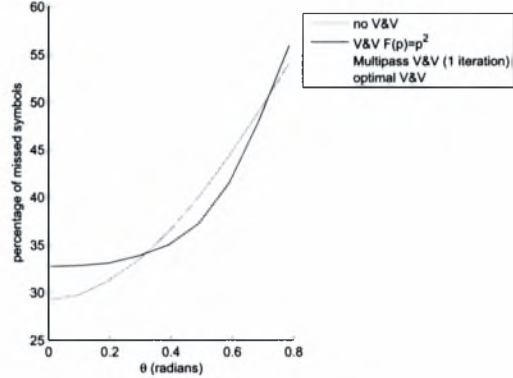
Θεωρήθηκε σκόπιμο να δοκιμαστούν οι αλγόριθμοι σε υψηλό θόρυβο με διαφορετικά μήκη παραθύρου σε λογικά πλαίσια, ώστε να θεωρείται ένα σύστημα υλοποιήσιμο, σε αντιδιαστολή με την 'ελευθερία', ως προς το μήκος του παραθύρου, που επιλέχθηκε για τον απλό V&V σε προηγούμενες προσομοιώσεις.

variable SNR $\Delta f=100\text{Hz}$ $\theta=0.31416$ window size=64



(α') Μεταβλητό SNR

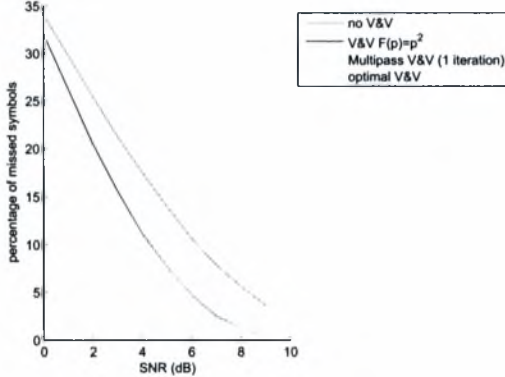
SNR=0dB $\Delta f=100\text{Hz}$ variable θ window size=64



(β') Μεταβλητός θόρυβος φάσης

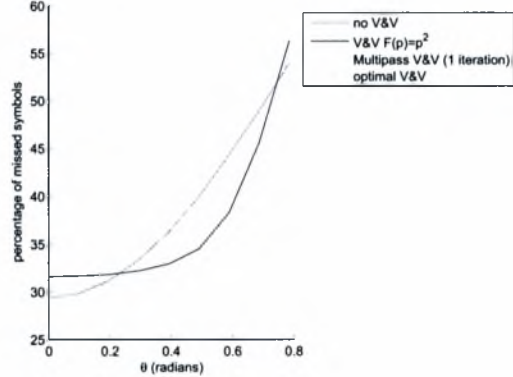
Σχήμα 5.17: Μήκος παραθύρου 64 σύμβολα

variable SNR $\Delta f=100\text{Hz}$ $\theta=0.31416$ window size=128



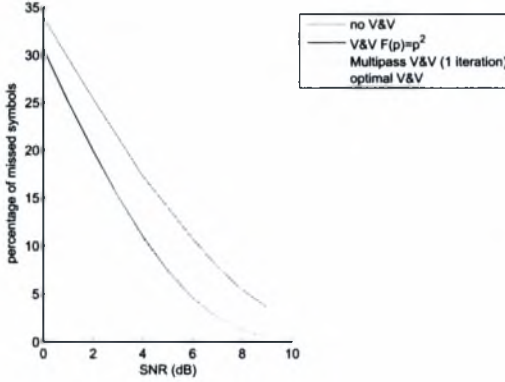
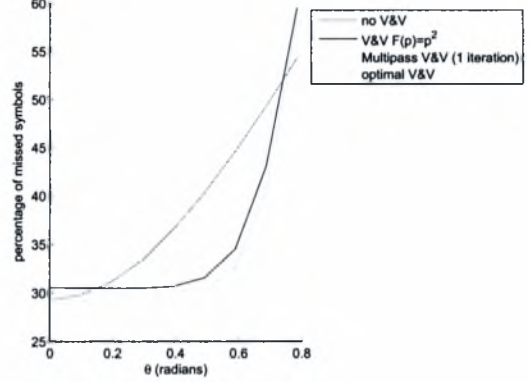
(α') Μεταβλητό SNR

SNR=0dB $\Delta f=100\text{Hz}$ variable θ window size=128



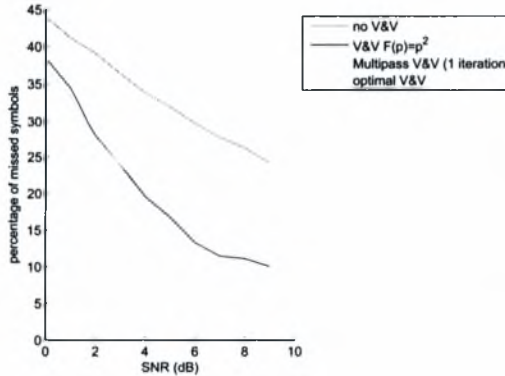
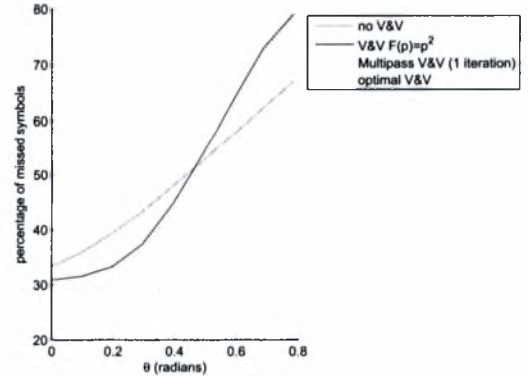
(β') Μεταβλητός θόρυβος φάσης

Σχήμα 5.18: Μήκος παραθύρου 128 σύμβολα

variable SNR $\Delta f=100\text{Hz}$ $\theta=0.31416$ window size=256(α') Μεταβλητό SNR SNR=0dB $\Delta f=100\text{Hz}$ variable θ window size=256

(β') Μεταβλητός θόρυβος φάσης

Σχήμα 5.19: Μήκος παραθύρου 256 σύμβολα

variable SNR $\Delta f=10000\text{Hz}$ $\theta=0.31416$ window size=256(α') Μεταβλητό SNR SNR=0dB $\Delta f=10000\text{Hz}$ variable θ window size=256

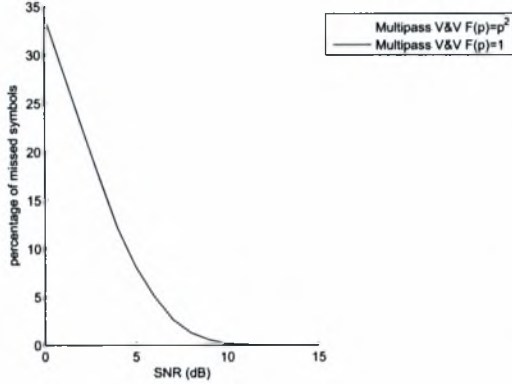
(β') Μεταβλητός θόρυβος φάσης

Σχήμα 5.20: Μήκος παραθύρου 256 σύμβολα και $\Delta f = 10000\text{Hz}$

Προσομοίωση εκτιμητή V&V πολλαπλών περασμάτων με διαφορετικό $F(\bullet)$

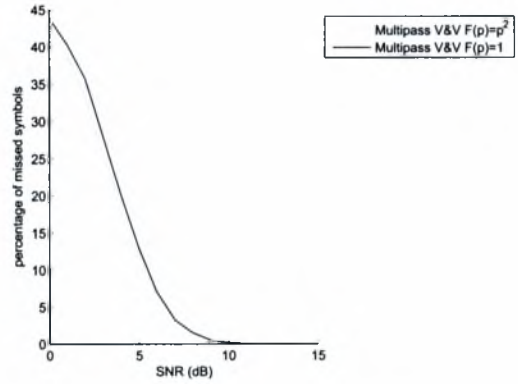
Σε αυτήν την υποενότητα μελετάται η επίπτωση της επιλογής διαφορετικής μη γραμμικής συνάρτησης $F(\rho)$ στον εκτιμητή V&V πολλαπλών περασμάτων.

variable SNR $\Delta f=100\text{Hz}$ $\theta=0.31416$ window size=32



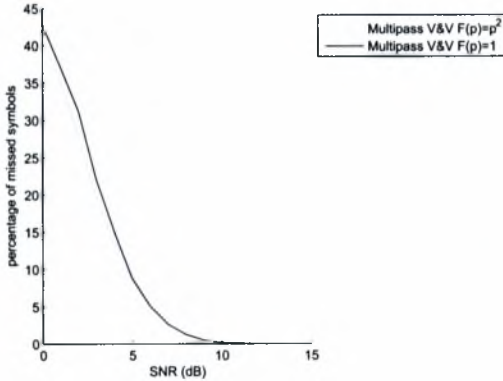
(α) $\theta = \frac{\pi}{10}$, $\Delta f = 100$ Hz, παράθυρο 32 συμβόλων

variable SNR $\Delta f=100\text{Hz}$ $\theta=0.62832$ window size=32



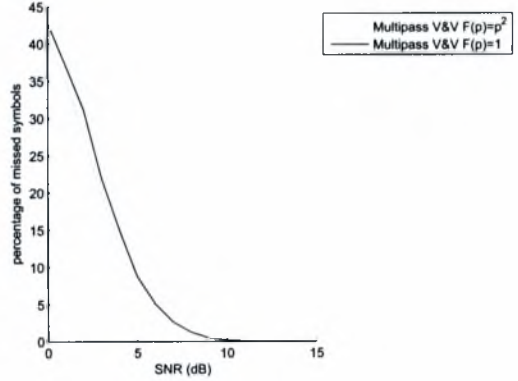
(β) $\theta = \frac{\pi}{5}$, $\Delta f = 100$ Hz, παράθυρο 32 συμβόλων

variable SNR $\Delta f=100\text{Hz}$ $\theta=0.62832$ window size=64



(γ) $\theta = \frac{\pi}{5}$, $\Delta f = 100$ Hz, παράθυρο 64 συμβόλων

variable SNR $\Delta f=100\text{Hz}$ $\theta=0.62832$ window size=64



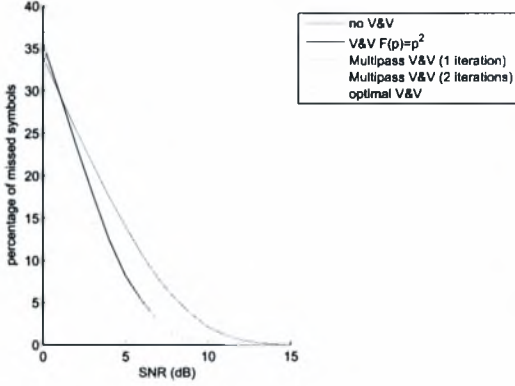
(δ) $\theta = \frac{\pi}{5}$, $\Delta f = 100$ Hz, παράθυρο 128 συμβόλων

Σχήμα 5.21: $F(\bullet) = 1$ και $F(\bullet) = (\bullet)^2$

Προσομοίωση εκτιμητή V&V πολλαπλών περασμάτων για 1 και 2 περάσματα

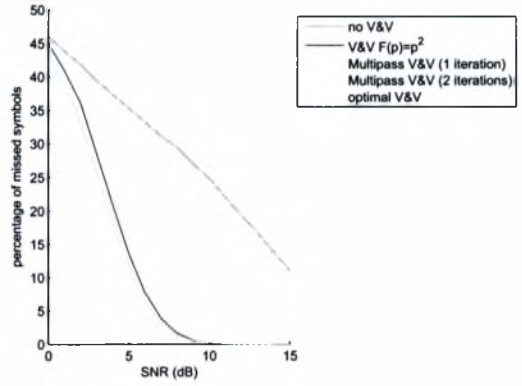
Μελέτη της βελτίωσης της εκτίμησης, όταν επιλεγούν περισσότερα από 1 περάσματα για τον εκτιμητή V&V πολλαπλών περασμάτων.

variable SNR $\Delta f=100\text{Hz}$ $\theta=0.31416$ window size=32



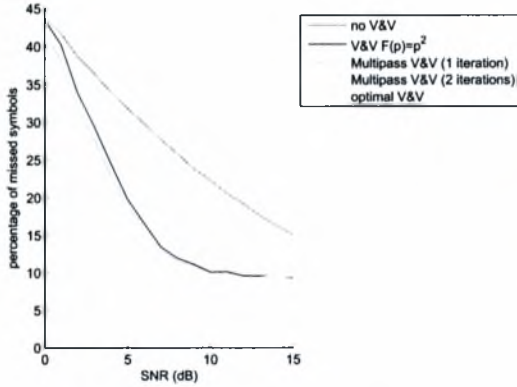
(α') $\theta = \frac{\pi}{10}$, $\Delta f = 100$ Hz, παράθυρο 32 συμβόλων

variable SNR $\Delta f=100\text{Hz}$ $\theta=0.62832$ window size=32



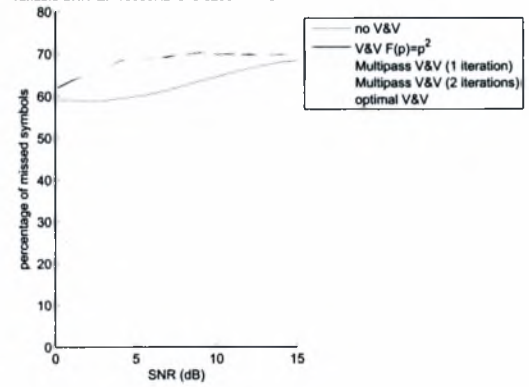
(β') $\theta = \frac{\pi}{5}$, $\Delta f = 100$ Hz, παράθυρο 32 συμβόλων

variable SNR $\Delta f=10000\text{Hz}$ $\theta=0.31416$ window size=32



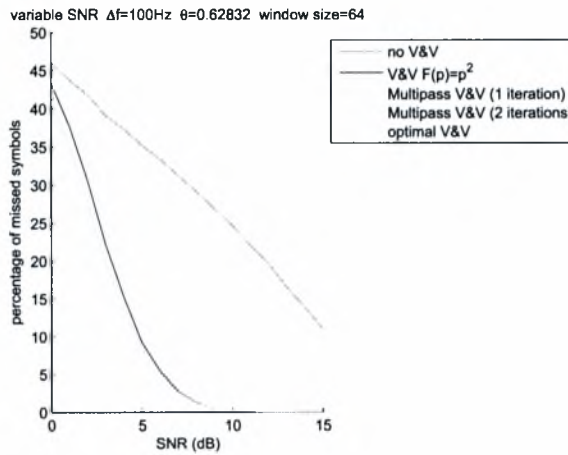
(γ') $\theta = \frac{\pi}{10}$, $\Delta f = 10000$ Hz, παράθυρο 32 συμβόλων

variable SNR $\Delta f=10000\text{Hz}$ $\theta=0.62832$ window size=32

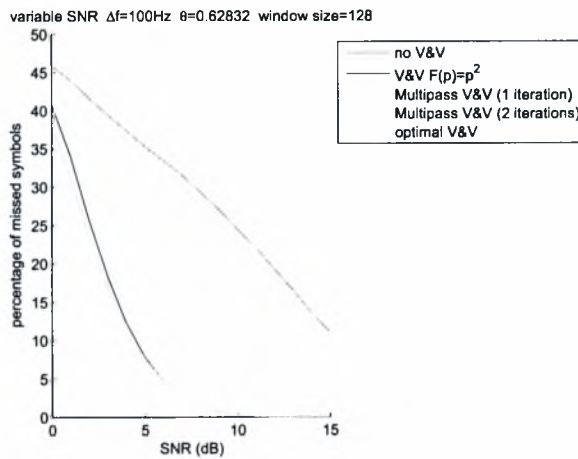


(δ') $\theta = \frac{\pi}{5}$, $\Delta f = 10000$ Hz, παράθυρο 32 συμβόλων

Σχήμα 5.22: 1 και 2 περάσματα για τον εκτιμητή V&V πολλαπλών περασμάτων



(α) $\theta = \frac{\pi}{5}$, $\Delta f = 100$ Hz, παράθυρο 64 συμβόλων



(β) $\theta = \frac{\pi}{5}$, $\Delta f = 100$ Hz, παράθυρο 128 συμβόλων

Σχήμα 5.23: 1 και 2 περάσματα για τον εκτιμητή V&V πολλαπλών περασμάτων, με παράθυρα των 64 και 128 συμβόλων

Παρατηρήσεις

Όπως είναι αναμενόμενο, από τις παραπάνω προσομοιώσεις γίνεται προφανές πως και ο βέλτιστος εκτιμητής V&V και ο V&V πολλαπλών περασμάτων είναι καλύτεροι σε σχέση με τον απλό V&V με $F(\rho) = (\rho)^2$. Ο καλύτερος από τους τρεις είναι φυσικά ο βέλτιστος και ακολουθούν ο πολλαπλών περασμάτων και ο απλός V&V.

Επίδραση του θορύβου του καναλιού και του θορύβου φάσης Η τυχαιότητα του θορύβου του καναλιού επηρεάζει πολύ την απόδοση των εκτιμητών. Όταν ο θόρυβος είναι αρκετά υψηλός, όλοι οι εκτιμητές δυσκολεύονται να καταλήξουν σε κάποια σωστή εκτίμηση, οδηγώντας σε πολλά λάθος αποδιαμορφωμένα σύμβολα. Η καλύτερη συμπεριφορά του εκτιμητή πολλαπλών περασμάτων φαίνεται σε περιπτώσεις υψηλού θορύβου. Τέλος, ο θόρυβος φάσης επηρεάζει όλους τους εκτιμητές, εκτός από τον βέλτιστο.

Επίδραση του Δf Όταν υπάρχει υψηλό Δf , άσχετα με τον σηματοθορυβικό λόγο, υπάρχει μια αρνητική επίδραση στην εκτίμηση της φάσης. Ο βέλτιστος εκτιμητής V&V δεν φαίνεται να επηρεάζεται από το Δf . Σε περιπτώσεις υψηλού Δf , ο απλός V&V και ο πολλαπλών περασμάτων εμφανίζονται να έχουν παρόμοια απόδοση.

Επίδραση του παραθύρου Για τον βέλτιστο αλγόριθμο, το μέγεθος του παραθύρου δεν παίζει κανένα ρόλο. Η απόδοση του παραμένει ίδια, ασχέτως του πόσο μεγάλο ή μικρό είναι το παράθυρο. Από την άλλη, ο εκτιμητής πολλαπλών περασμάτων προσεγγίζει καλύτερα την καμπύλη του βέλτιστου όσο αυξάνεται το παράθυρο. Ομοίως, και ο απλός V&V πλησιάζει όλο και περισσότερο τον πολλαπλών περασμάτων όσο αυξάνεται το παράθυρο.

Επίδραση του n στον εκτιμητή V&V πολλαπλών περασμάτων Από τις προσομοιώσεις, η επιλογή του n δεν επηρεάζει ιδιαίτερα την αποδοτικότητα του εκτιμητή. Ακόμα και με $n = 1$, τα λάθος αποδιαμορφωμένα σύμβολα σε σχέση με το $n = 2$ είναι μόλις 1% περισσότερα, και αυτό μόνον όταν ο σηματοθορυβικός λόγος πλησιάζει τα 0 dB. Γενικά, η λειτουργία με οποιοδήποτε n είναι παρόμοια.

Συμπεράσματα

Παρατηρείται το φαινόμενο ο εκτιμητής πολλαπλών περασμάτων απλώς να είναι οριακά καλύτερος του απλού V&V. Αυτό είναι φυσιολογικό, αν αναλογιστεί κανείς πως για την αρχική εκτίμηση του θορύβου φάσης που θα οδηγήσει σε σωστή ή μη αποδιαμόρφωση των συμβόλων, στηρίζεται στον απλό V&V. Συνεπώς, η αρχική επιλογή του αλγορίθμου εκτίμησης επηρεάζει την συμπεριφορά του εκτιμητή πολλαπλών περασμάτων. Οι προσομοιώσεις με $F(\rho) = (\rho)^2$ και $F(\rho) = 1$ δείχνουν πως ακόμα και στην επιλογή του δεύτερου, που αν και είναι απλούστερος σαν υλοποίηση και χειρότερος από άποψη απόδοσης, η εκτίμηση που δίνει ο εκτιμητής πολλαπλών περασμάτων είναι το ίδιο καλή με την επιλογή του πρώτου.

Ο βέλτιστος εκτιμητής V&V επηρεάζεται μόνον από τον θόρυβο του καναλιού. Συγκεκριμένα, η τυχαιότητα του θορύβου δυσχαιρένει την εκτίμηση ακόμα και στον βέλτιστο εκτιμητή. Αυτό είναι αναμενόμενο, αφού ακόμα και με γνώση των συμβόλων, ο τυχαίος θόρυβος δεν γίνεται να εκτιμηθεί

ή να υπολογιστεί, παρά μόνον η μέση τιμή του, που από πεπερασμένα σύμβολα απέχει από την πραγματική μέση τιμή.

Παρόλα αυτά, το Δf και ο θόρυβος φάσης δεν επηρεάζουν την λειτουργία του. Ο βέλτιστος εκτιμητής μπορεί να εξάγει με εξαιρετική ακρίβεια κάποιον σταθερό ή ελαφρά μεταβαλλόμενο θόρυβο φάσης, όπως είναι εμφανές σε περιπτώσεις υψηλού θορύβου φάσης και υψηλού Δf .

Τα μεγαλύτερα παράθυρα επηρεάζουν θετικά όλους τους εκτιμητές. Συγκεκριμένα, όσο μεγαλώνει το παράθυρο, τόσο προσεγγίζεται ο βέλτιστος εκτιμητής από τους άλλους δύο. Παρόλα αυτά, όταν ο θόρυβος φάσης και ο θόρυβος καναλιού είναι ισχυροί, η απόδοση σε περίπτωση αύξησης του παραθύρου μετριάζεται. Δεδομένου της ταχύτητας επεξεργασίας των δεδομένων που χρειάζεται, παράθυρα μήκους 128 συμβόλων ήδη είναι αρκετά μεγάλα.

Η χαμηλή, σε σχέση με την πολυπλοκότητά του, απόδοση του εκτιμητή πολλαπλών περασμάτων δεν τον κάνει και πολύ επιθυμητή επιλογή για κάποιο πραγματικό σύστημα. Πρέπει να ληφθεί υπόψιν ότι στηρίζεται σε δύο περάσματα από τα δεδομένα, πως χρειάζεται αρκετά μεγάλα παράθυρα και επιπλέον υλικό για την αποδιαμόρφωση των συμβόλων.

5.4 Προσομοιώσεις των εκτιμητών βαρυκέντρου

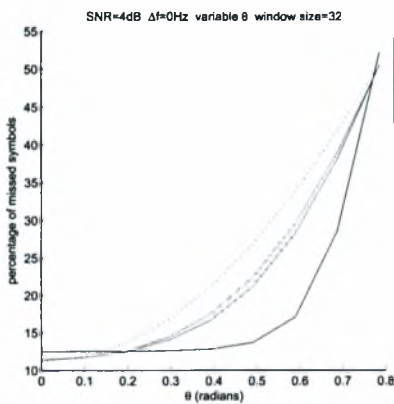
Το παρόν κεφάλαιο αποτελείται από δύο ενότητες.

Στην πρώτη ενότητα συγκρίνονται οι επιδόσεις των εκτιμητών BAR, EBAR και LBAR σε σχέση με τον εκτιμητή V&V με $F(\rho) = (\rho)^2$, για λόγους σύγκρισης. Οι προσομοιώσεις είναι παρόμοιες με αυτές της προηγούμενης ενότητας, οπότε οποιαδήποτε σύγκριση μεταξύ των δύο είναι εφικτή. Ο αναγνώστης πρέπει επίσης να έχει υπόψιν του πως πάντα οι εκτιμητές V&V χρησιμοποιούν ένα σύμβολο παραπάνω, αφού εκτιμούν τον θόρυβο φάσης του κεντρικού συμβόλου. Αντίθετα, οι εκτιμητές βαρυκέντρου χρησιμοποιούν το παράθυρο για να εκτιμήσουν τον θόρυβο φάσης του πρώτου συμβόλου στο παράθυρο.

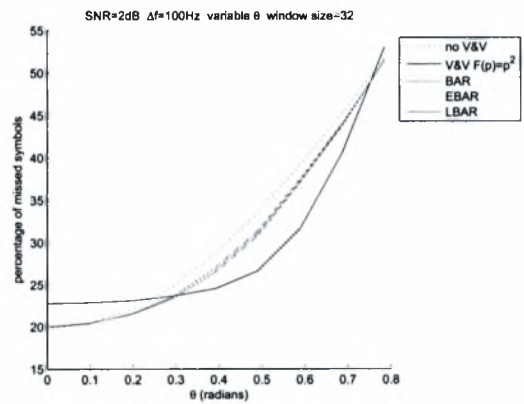
Στην δεύτερη ενότητα συγκρίνονται οι εκτιμητές LBAR και QLBAR, ώστε να αποδειχθεί ο ισχυρισμός ότι ο QLBAR είναι οριακά χειρότερος από τον μη-χβαντισμένο 'αδελφό' του, τον LBAR.

5.4.1 Μελέτη εκτιμητών βαρυκέντρου σε σχέση με τον V&V

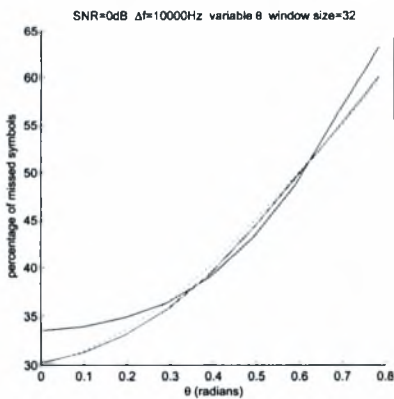
Προσομοίωση για μεταβλητό θόρυβο φάσης



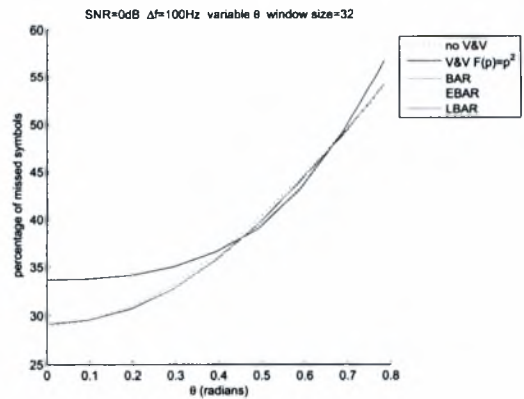
(α') $SNR = 4 \text{ dB}$, $\Delta f = 0 \text{ Hz}$



(β') $SNR = 2 \text{ dB}$, $\Delta f = 100 \text{ Hz}$



(γ') $SNR = 0 \text{ dB}$, $\Delta f = 10000 \text{ Hz}$

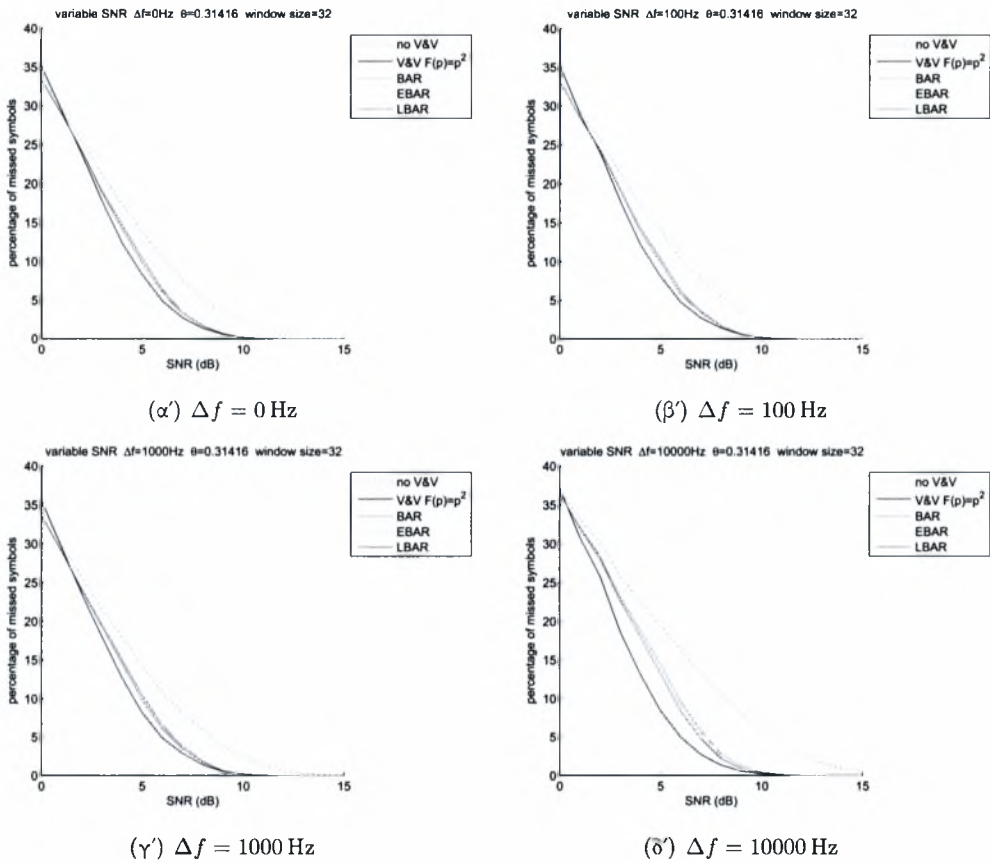


(δ') $SNR = 0 \text{ dB}$, $\Delta f = 100 \text{ Hz}$

Σχήμα 5.24: Ποσοστό εσφαλμένων συμβόλων

Προσομοίωση για $SNR \in [0 \text{ dB}, 15 \text{ dB}]$

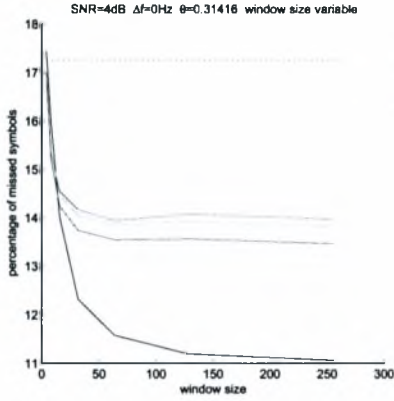
Προσομοιώσεις για αυξανόμενα επίπεδα θορύβου.



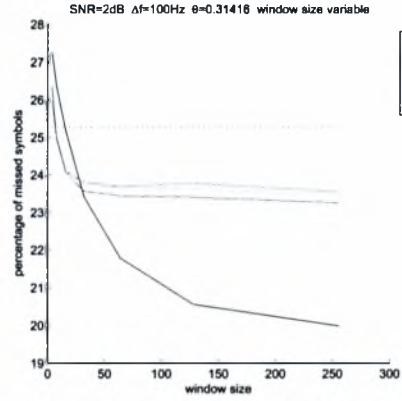
Σχήμα 5.25: Ποσοστό εσφαλμένων συμβόλων

Προσομοίωση για μεταβλητό μήκος παραθύρου

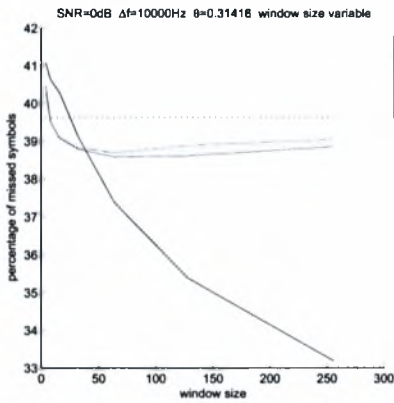
Προσομοιώσεις που μελετούν την επίδραση μεγαλύτερων παραθύρων στους εκτιμητές βαρυκέντρου, σε σχέση με τον εκτιμητή V&V.



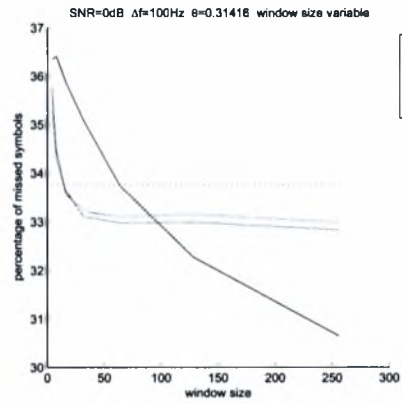
(α') $SNR = 4 \text{ dB}$, $\Delta f = 0 \text{ Hz}$



(β') $SNR = 2 \text{ dB}$, $\Delta f = 100 \text{ Hz}$



(γ') $SNR = 0 \text{ dB}$, $\Delta f = 10000 \text{ Hz}$



(δ') $SNR = 0 \text{ dB}$, $\Delta f = 100 \text{ Hz}$

Σχήμα 5.26: Ποσοστό εσφαλμένων συμβόλων

Παρατηρήσεις

Επίδραση του θορύβου του καναλιού και του θορύβου φάσης Συνολικά, όλοι οι εκτιμητές βαρυκέντρου παρουσιάζονται χειρότεροι σε απόδοση σε σχέση με τον εκτιμητή V&V. Όσο αυξάνεται ο θόρυβος του καναλιού, ο θόρυβος φάσης και το Δf , αυτό το φαινόμενο γίνεται πιο έντονο. Στο σημείο που οι εκτιμητές βαρυκέντρου υπερέχουν είναι στην περίπτωση χαμηλού θορύβου φάσης. Σε αυτές τις περιπτώσεις ο V&V κάνει περισσότερο κακό, παρά καλό, αφού αποτυγχάνει να εκτιμήσει σωστά το θόρυβο φάσης. Οι εκτιμητές βαρυκέντρου εμφανίζονται να έχουν σαν κάτω όριο απόδοσης την απόδοση του συστήματος χωρίς διόρθωση φάσης.

Από όλες τις προσομοιώσεις, η σειρά κατά αύξουσας απόδοσης είναι: LBAR, EBAR και BAR. Σε καμία περίπτωση αυτή η σειρά δεν ανατρέπεται. Πλησιάζοντας τα 10 dB, εν τέλει όλοι οι αλγόριθμοι συγκλίνουν στην καμπύλη του απλού V&V.

Επίδραση του παραθύρου Μέχρι παράθυρο μήκους 32 συμβόλων, όλοι οι αλγόριθμοι επωφελούνται από αύξηση του μήκους του παραθύρου. Πέραν των 32 συμβόλων, δεν υπάρχει κάποια αξιοσημείωτη βελτίωση.

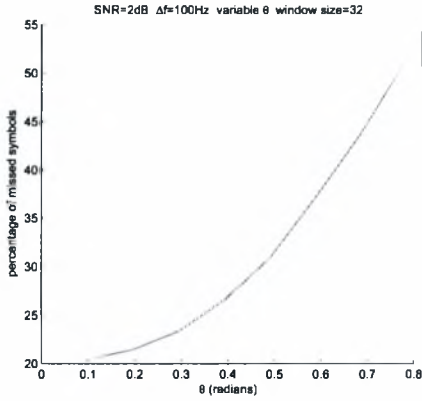
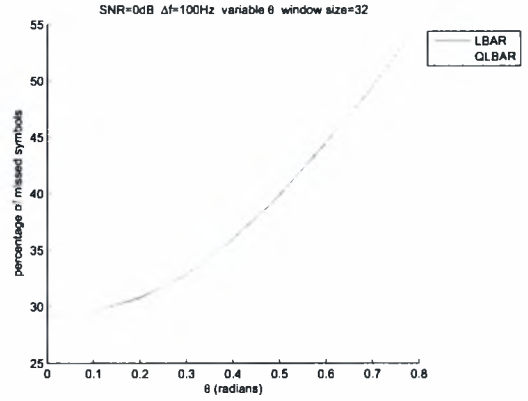
Συμπεράσματα

Οι εκτιμητές βαρυκέντρου είναι τελικά χειρότεροι από τους εκτιμητές V&V. Η μόνη περίπτωση που εμφανίζονται καλύτεροι είναι όταν ο θόρυβος φάσης είναι αρκετά μικρός σε σχέση με τον θόρυβο του καναλιού. Και από προηγούμενες προσομοιώσεις, είχε γίνει αντιληπτό το γεγονός πως σε τέτοιες περιπτώσεις, ο V&V με παράθυρο 32 συμβόλων οδηγεί σε περισσότερα λάθη από ότι αν δεν υπήρχε καμία διόρθωση φάσης. Σε αυτές τις περιπτώσεις οι εκτιμητές βαρυκέντρου επιτρέπουν κατά 0.5% περίπου λιγότερα λάθη στα αποδιαμορφωμένα σύμβολα σε σχέση με το να μην υπήρχε καμία διόρθωση θορύβου φάσης.

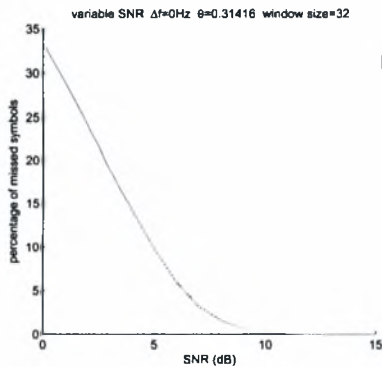
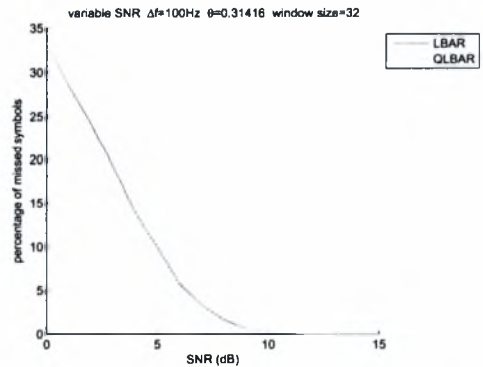
Το βέλτιστο παράθυρο εμφανίζεται στα 32 σύμβολα. Πέρα από αυτήν την τιμή, δεν υπάρχει κάποια αξιοσημείωτη βελτίωση, εν αντιθέσει με τον εκτιμητή V&V.

5.4.2 Σύγκριση εκτιμητών LBAR και QLBAR

Έχει αναφερθεί ότι ο εκτιμητής QLBAR είναι η χβαντισμένη μορφή του LBAR. Μένει συνεπώς να μελετηθεί η αποτελεσματικότητά του κάτω από ίδιες συνθήκες.

(α') $SNR = 2 \text{ dB}$, $\Delta f = 100 \text{ Hz}$ (β') $SNR = 0 \text{ dB}$, $\Delta f = 100 \text{ Hz}$

Σχήμα 5.27: Ποσοστό εσφαλμένων συμβόλων για μεταβλητό θόρυβο φάσης

(α) $\Delta f = 0 \text{ Hz}$ (β) $\Delta f = 100 \text{ Hz}$

Σχήμα 5.28: Ποσοστό εσφαλμένων συμβόλων για διαφορετικά SNR

Συμπεράσματα

Ο εκτιμητής QLBAR υπολείπεται ελάχιστα του LBAR σε θέματα απόδοσης. Να τονιστεί πως ο QLBAR είναι πιο κατάλληλος για υλοποίηση, αφού η συνάρτηση αξιοπιστίας έχει 4 διακριτές τιμές που μπορούν να κωδικοποιηθούν μόνον με 2 bits. Από αυτήν την άποψη, η σχεδιάσή του είναι επιτυχημένη.

Παράρτημα Α'

ΧΡΗΣΙΜΕΣ ΣΧΕΣΕΙΣ

Α'.1 Σχέση ενέργειας συμβόλου με ενέργεια bit

Αν E_s η ενέργεια ενός συμβόλου και E_b η ενέργεια ενός bit, τότε ισχύει

$$E_s = \begin{cases} E_b & \text{BPSK} \\ 2 \cdot E_b & \text{QPSK} \\ m \cdot E_b & \text{mPSK} \end{cases}$$

Α'.2 Χρήσιμες τριγωνομετρικές σχέσεις

$$\cos(-\alpha) = \cos(\alpha) \quad (\text{A'.1})$$

$$\sin(-\alpha) = -\sin(\alpha) \quad (\text{A'.2})$$

$$\sin(\alpha) = \cos\left(\frac{\pi}{2} - \alpha\right) \quad (\text{A'.3})$$

$$\cos^2(a) = \frac{1}{2} + \frac{1}{2} \cos(2a) \quad (\text{A'.4})$$

$$\sin^2(a) = \frac{1}{2} - \frac{1}{2} \cos(2a) \quad (\text{A'.5})$$

$$\sin(a + b) = \sin(a) \cos(b) + \cos(a) \sin(b) \quad (\text{A'.6})$$

$$\sin(a - b) = \sin(a) \cos(b) - \cos(a) \sin(b) \quad (\text{A'.7})$$

$$\cos(a + b) = \cos(a) \cos(b) - \sin(a) \sin(b) \quad (\text{A'.8})$$

$$\cos(a - b) = \cos(a) \cos(b) + \sin(a) \sin(b) \quad (\text{A'.9})$$

$$\cos(a + b) + \cos(a - b) = 2 \cos(a) \cos(b) \quad (\text{A'.10})$$

$$\cos(a + b) - \cos(a - b) = -2 \sin(a) \sin(b) \quad (\text{A'.11})$$

$$\sin(a + b) + \sin(a - b) = 2 \sin(a) \cos(b) \quad (\text{A'.12})$$

$$\sin(a + b) - \sin(a - b) = 2 \cos(a) \sin(b) \quad (\text{A'.13})$$

$$\frac{d}{dt} \left(\frac{\sin(a + xt)}{x} \right) = \cos(a + xt) \quad (\text{A'.14})$$

$$\frac{d}{dt} \left(\frac{\cos(a + xt)}{x} \right) = -\sin(a + xt) \quad (\text{A'.15})$$

A'.3 Χρήσιμες μιγαδικές σχέσεις

Έστω $x = a + j \cdot b, x \in \mathbb{C}$

$$x^2 = (a + j \cdot b)^2 = a^2 - b^2 + 2jab \quad (\text{A'.16})$$

Αν $a, b \in \mathbb{C}$

$$\begin{aligned} |a - b|^2 &= (a - b)\overline{(a - b)} \\ &= (a - b)(\bar{a} - \bar{b}) \\ &= a\bar{a} - a\bar{b} - \bar{a}b + b\bar{b} \\ &= |a|^2 - 2\text{Re}\{a\bar{b}\} + |b|^2 \end{aligned} \quad (\text{A'.17})$$

Παράρτημα Β'

ΚΩΔΙΚΑΣ MatLab

Σε αυτό το παράρτημα παρουσιάζεται ο κώδικας MatLab που χρησιμοποιήθηκε στην παρούσα εργασία.

Β'.1 Κώδικας Κεφαλαίου 2

Β'.1.1 Κώδικας δημιουργίας του Σχήματος(2.5)

```
1 clear;
2 % generate the 4 QPSK symbols
3 Z = [pi/4:pi/2:2*pi];
4 % phase error
5 th = pi/10;
6 % original signal without phase error
7 y = exp(j*Z);
8 % signal with positive phase error
9 yp = exp(j*(Z+th));
10 % signal with negative phase error
11 ym = exp(j*(Z-th));
12 % plot signal phases
13 polar(angle(y), abs(y), 'ob');
14 hold on;
15 polar(angle(yp), abs(yp), 'xr');
16 polar(angle(ym), abs(ym), 'xm');
17 legend('Signal_without_phase_error', 'Signal_with_positive_phase_error', ...
18       'Signal_with_negative_phase_error', 'Location', 'BestOutside');
```

Β'.2 Κώδικας Κεφαλαίου 4

Β'.2.1 Κώδικας δημιουργίας του Σχήματος(4.2)

```
1 clear;
2 % number of symbols
```

```

3 M = 4;
4 % phases
5 e = 0.01;
6 th1 = pi/M - e;
7 th2 = -pi/M + e;
8 % BAR function
9 th = mod(th1 + 0.5*mod(th2-th1, 2*pi/M), 2*pi/M);
10 % mean
11 m = (th1+th2)/2;
12 % plot
13 figure;
14 polar(th1, 1, 'ob');
15 hold on;
16 polar(th2, 1, 'or');
17 polar(th, 1, 'ok');
18 polar(m, 1, 'y');
19 title('BAR_function');
20 legend('\theta_1', '\theta_2', 'BAR(\theta_1,\theta_2)', 'Mean', ...
21       'Location', 'BestOutside');

```

Β'.2.2 Κώδικας δημιουργίας του Σχήματος(4.3)

```

1 clear;
2 % number of symbols
3 M = 4;
4 % create phase difference
5 th_diff = [-2*pi/M:0.01:2*pi/M];
6 ath_diff = abs(th_diff);
7 % LBAR reliability
8 f_lbar = 1 - (M/(2*pi))*ath_diff;
9 % QLBAR reliability
10 g1 = ath_diff<pi/8;
11 g2 = ath_diff<pi/4;
12 g3 = ath_diff<(3*pi/8);
13 gg1 = g1;
14 gg2 = g2 - g1;
15 gg3 = g3 - g2;
16 f_qlbar = 1*gg1 + 0.5*gg2 + 0.25*gg3;
17 % plot
18 figure;
19 axis([(-2*pi/M)-0.5 (2*pi/M)+0.5 -0.2 1.2]);
20 hold on;
21 grid;
22 plot(th_diff, f_lbar, 'b');
23 plot(th_diff, f_qlbar, 'r');
24 title('LBAR_and_QLBAR_f_function');
25 legend('LBAR', 'QLBAR', 'Location', 'BestOutside');

```


B'.3 Συναρτήσεις προσομοιώσεων

B'.3.1 Κώδικας μετατροπής ακτινίων σε μοίρες

Βασίζεται στην γνωστή σχέση

$$\frac{\text{degrees}}{180^\circ} = \frac{\text{radians}}{\pi} \quad (\text{B'.1})$$

```

1 function deg = rad2deg(rad)
2 %rad2deg Converts radians to degrees.
3 %
4 %   rad2deg(r) returns angle r in degrees.
5
6 %   Author(s): I. Papadopoulos
7 %   Copyright 2005
8 %   Revision : 1.1 Date : 2005/07/25
9
10 deg = (180/pi) * rad;
```

B'.3.2 Κώδικας εύρεσης φάσης συμβόλου στο διάστημα $[-\frac{\pi}{4}, \frac{3\pi}{2} + \frac{\pi}{4}]$

Οι μετρήσεις υπολοίπου φάσης απαιτούν μία συνάρτηση για την ανάκτηση της φάσης του συμβόλου. Η συνάρτηση `angle()` του MatLab έχει πεδίο τιμών $[-\pi, \pi]$. Συνεπώς, αν κάποιο σύμβολο είχε φάση π και προστίθετο κάποιος θόρυβος φάσης θ , τότε η `angle()` θα επέστρεφε $-\pi + \theta$. Για τις ανάγκες του υπόλοιπου κώδικα, η παρακάτω συνάρτηση επιστρέφει $\pi + \theta$ και δουλεύει για οποιαδήποτε PSK.

```

1 function a = symbol_phase(y, M)
2 %symbol_phase Returns the phase of a MPSK symbol.
3 %
4 %   symbol_phase(y,M) returns the phase of symbol y, in radians, in
5 %   [-pi/M, 2*pi-pi/M].
6
7 %   Author(s): I. Papadopoulos
8 %   Copyright 2005
9 %   Revision : 1.1 Date : 2005/08/01
10
11 pi2 = 2*pi;
12 a = mod(pi2 + angle(y) + pi/M, pi2) - pi/M;
```

B'.3.3 Κώδικας ανίχνευσης συμβόλων

Η συνάρτηση `symbol_testing(y, m)` ελέγχει σε ποια τιμή φάσης είναι πιο κοντά – δηλαδή σε ποια περιοχή ανήκει – η φάση του συμβόλου y . m είναι το σύνολο των φάσεων των ιδανικών συμβόλων. Υλοποιεί την ανίχνευση συμβόλων.

```

1 function k = symbol_testing(y, m)
2 %symbol_testing Hypothesis testing on a MPSK symbol.
3 %
4 % symbol_testing(y, m) returns which symbol is carried by sample y, compared
5 % to symbol vector m. m has the symbol phase, in radians, of the system.
6
7 % Author(s): I. Papadopoulos
8 % Copyright 2005
9 % Revision : 1.1 Date : 2005/08/01
10
11 a = symbol_phase(y, length(m));
12 [Y, k] = min( abs(a-m) );
13 k = k-1; % symbols are 0-based, index 1-based

```

B'.3.4 Κώδικας εκτιμητή V&V

Υλοποιεί τον εκτιμητή V&V, χρησιμοποιώντας την συνάρτηση `angle()`.

```

1 function th = viterbis(p, f, n, M)
2 %viterbis Implements a simpler form of Viterbi & Viterbi estimator, as found
3 % in Catherine Morlet's PhD thesis.
4 %
5 % viterbis(p,f,n,M) implements the Viterbi & Viterbi estimator. It uses the
6 % power vector p, phase vector f and n power for the non-linear function F(p)
7 % to estimate the phase error for the central symbol of the vector.
8
9 % Author(s): I. Papadopoulos
10 % Copyright 2005
11 % Revision : 1.1 Date : 2005/06/18
12
13 th = angle( sum( (p.^n).*exp(j*M*f) ) )/M;

```

B'.3.5 Κώδικας βέλτιστου εκτιμητή V&V

```

1 function th = viterbi_o(y, Z)
2 %viterbi_o Implements the optimal Viterbi & Viterbi algorithm.
3 %
4 % viterbi_o(y,Z) estimates the phase error of the central symbol in vector y.
5 % The symbol data must be provided through vector Z, or optimal V&V will not
6 % work.
7 %
8 % This algorithm presumes that symbol data is known and performs near-optimal
9 % phase error estimation.
10
11 % Author(s): I. Papadopoulos
12 % Copyright 2005
13 % Revision : 1.2 Date : 2005/07/25
14
15 s = y.*exp(-j*Z);
16 th = angle( sum(s) );

```

B'.3.6 Κώδικας συνάρτησης αφαίρεσης διαμόρφωσης *FOLD*

Η συνάρτηση *FOLD()* επιστρέφει τιμές στο διάστημα $[-\frac{\pi}{M}, \frac{\pi}{M}[$. Αν υποτεθεί ότι η φάση που της δίνεται σαν όρισμα είναι $f_n = k_n \frac{\pi}{M/2} + \theta$, με k_n το σύμβολο MPSK και θ κάποιον θόρυβο, αν ακολουθηθεί η απλή λογική του πολλαπλασιασμού με M και μετά modulo ώστε να αφαιρεθούν τα πολλαπλάσια του 2π και διαίρεση ώστε να επιστραφεί το σωστό θ , τότε θα ήταν

$$\begin{aligned} \frac{\text{mod}((f_n)M, 2\pi)}{M} &= \frac{\text{mod}\left(\left(k_n \frac{\pi}{M/2} + \theta\right)M, 2\pi\right)}{M} \\ &= \frac{\text{mod}(2k_n\pi + M\theta, 2\pi)}{M} \\ &= \frac{\text{mod}(M\theta, 2\pi)}{M} \end{aligned}$$

που σημαίνει ότι η επιστρεφόμενη τιμή είναι στο διάστημα $[-\frac{\pi}{M/2}, \frac{\pi}{M/2}[$.

Αν από την άλλη χρησιμοποιηθεί το εξής

$$\begin{aligned} \frac{\text{mod}(M(f_n + \frac{\pi}{M}), 2\pi)}{M} - \frac{\pi}{M} &= \frac{\text{mod}(2k_n\pi + M\theta + \pi, 2\pi)}{M} - \frac{\pi}{M} \\ &= \frac{\text{mod}(M\theta + \pi, 2\pi)}{M} - \frac{\pi}{M} \end{aligned}$$

η επιστρεφόμενη τιμή βρίσκεται στο διάστημα $[-\frac{\pi}{M}, \frac{\pi}{M}[$ που είναι και το ζητούμενο.

```

1 function fnn = FOLD(fn,M)
2 %FOLD Removes MPSK modulation from the given angle.
3 % FOLD(fn,M) removes modulation from angle f, given in radians, of a MPSK
4 % symbol.
5
6 % Author(s): I. Papadopoulos
7 % Copyright 2005
8 % Revision : 1.1 Date : 2005/06/18
9
10 % This is the original, iterative implementation
11 % pim = pi/M;
12 % pim2 = pi/(M/2);
13 % fnn = fn;
14 % while 1
15 %     if fnn==--pim
16 %         return;
17 %     end
18 %     if abs(fnn)<pim
19 %         return;
20 %     end
21 %     if fnn>0
22 %         fnn = fnn - pim2;
23 %     else
24 %         fnn = fnn + pim2;
25 %     end
26 % end
27
28 % However for Matlab the code below is more efficient
29 fnn = mod(M*(fn+pi/M), 2*pi)/M - pi/M;

```

B'3.7 Κώδικας συνάρτησης βαρυκέντρου *BAR*

```

1 function p = BAR(th1, th2, M)
2 %BAR The original barycenter averaging function for MPSK.
3 % BAR(th1,th2,M) averages two phase angles, in radians, using the barycenter
4 % algorithm for a MPSK system.
5
6 % Author(s): I. Papadopoulos
7 % Copyright 2005
8 % Revision : 1.3 Date : 2005/09/27
9
10
11 % This is the original, iterative implementation
12 % D = f2 - f1;
13 %
14 % if abs(D)<pi/4
15 %     f2 = f1 + D;
16 % else
17 %     if D<0
18 %         f2 = f1 + D + pi/2;
19 %     else
20 %         f2 = f1 + D - pi/2;
21 %     end
22 % end
23 %
24 % fa = (f2 - f1)/2;
25 %
26 % while abs(fa)>pi/4
27 %     if fa<0
28 %         fa = fa + pi/2;
29 %     else
30 %         fa = fa - pi/2;
31 %     end
32 % end
33
34 % However for Matlab the code below is more efficient
35 mv = 2*pi/M;
36 p = rem(th1 + 0.5*rem(th2-th1, mv), mv);

```

B'3.8 Κώδικας συνάρτησης βαρυκέντρου *EBAR*

```

1 function [p,b] = EBAR(th1, th2, b1, b2, thresh, M)
2 %EBAR The enhanced barycenter averaging function for MPSK.
3 % EBAR(th1,th2,b1,b2,thresh,M) averages two phase angles, in radians, using
4 % the enhanced barycenter algorithm for a MPSK system. It needs the fiability
5 % bits b1 and b2 for phase angles th1 and th2 respectively and the threshold
6 % to generate the fiability bit of the returned values.
7 %
8 % It is more commonly used as [p,b] = EBAR(th1,th2,b1,b2,thresh,M) where p is
9 % the computed average phase and b the fiability bit.
10
11 % Author(s): I. Papadopoulos
12 % Copyright 2005
13 % Revision : 1.2 Date : 2005/09/27

```

```

14
15 a1 = b1*th1 + (1-b1)*th2;
16 a2 = b2*th2 + (1-b2)*th1;
17
18 mv = 2*pi/M;
19 p = rem(a1 + 0.5*rem(a2-a1, mv), mv);
20 b = (b1|b2) | (abs(th2-th1)>thresh);

```

Β.3.9 Κώδικας συνάρτησης βαρυκέντρου LBAR

```

1 function [p,f] = LBAR(th1, th2, f1, f2, M)
2 %LBAR The LBAR barycenter averaging function for MPSK.
3 % LBAR(th1,th2,f1,f2,M) averages two phase angles, in radians, using the
4 % LBAR barycenter algorithm for a MPSK system. It needs the fiability values
5 % f1 and f2 for phase angles th1 and th2 respectively.
6 %
7 % It is more commonly used as [p,f] = LBAR(th1,th2,f1,f2,M) where p is the
8 % computed average phase and f the fiability value.
9
10 % Author(s): I. Papadopoulos
11 % Copyright 2005
12 % Revision :1.2 Date : 2005/09/18
13
14 mv = 2*pi/M;
15 fiab = f2/(f1+f2);
16 p = rem(th1 + fiab*rem(th2-th1, mv), mv);
17 f = 1 - (M/(2*pi))*abs(th2-th1);

```

Β.3.10 Κώδικας συνάρτησης βαρυκέντρου QLBAR

```

1 function [p,f] = QLBAR(th1, th2, f1, f2)
2 %QLBAR The QLBAR barycenter averaging function for QPSK.
3 % QLBAR(th1,th2,f1,f2) averages two phase angles, in radians, using the
4 % QLBAR barycenter algorithm for a QPSK system. It is a quantized version of
5 % the LBAR algorithm. It needs the fiability values f1 and f2, which are in
6 % the space[1,0.5,0.25,0], for phase angles th1 and th2 respectively.
7 %
8 % It is more commonly used as [p,f] = QLBAR(th1,th2,f1,f2) where p is the
9 % computed average phase and f the fiability value.
10
11 % Author(s): I. Papadopoulos
12 % Copyright 2005
13 % Revision :1.2 Date : 2005/09/18
14
15 if f2==0
16     fiab = 0;
17 elseif f1==0
18     fiab = 1;
19 elseif f2<f1
20     fiab = 0.25;
21 else
22     fiab = 0.5;

```

```

23 end
24
25 mv = 2*pi/4;
26 p = rem(th1 + fiab*rem(th2-th1, mv), mv);
27
28 d = abs(th2-th1);
29 if d<pi/8
30     f = 1;
31 elseif d<pi/4
32     f = 0.5;
33 elseif d<3*pi/8
34     f = 0.25;
35 else
36     f = 0;
37 end

```

B'.3.11 Κώδικας εκτιμητή BAR

```

1 function th = MEMARB(f, M)
2 %MEMARB Estimates the phase of the signal phase vector f and the BAR function.
3 %
4 % MEMARB(f,M) estimates phase error of the signal phase vector f, given in
5 % radians, in MPSK systems using the BAR algorithm. length(f) must be a power
6 % of 2.
7
8 % Author(s): I. Papadopoulos
9 % Copyright 2005
10 % Revision : 1.1 Date : 2005/06/18
11
12 ft = f;
13 while length(ft)>1
14     nf = zeros(1, length(ft)/2);
15     j = 1;
16     for i=1:2:length(ft)
17         nf(j) = BAR(ft(i), ft(i+1), M);
18         j = j+1;
19     end
20     ft = nf;
21 end
22 th = ft(1);

```

B'.3.12 Κώδικας εκτιμητή EBAR

```

1 function th = MEMARB_EBAR(f, thres, M)
2 %MEMARB_EBAR Estimates the phase of the signal phase vector f and the EBAR
3 % function.
4 %
5 % MEMARB_EBAR(f,thres,M) estimates phase error of the signal phase vector f,
6 % given in radians, in MPSK systems using the EBAR algorithm. length(f) must
7 % be a power of 2. thres is the threshold for the fiability bits.
8
9 % Author(s): I. Papadopoulos

```

```

10 % Copyright 2005
11 % Revision : 1.3 Date : 2005/09/21
12
13 bt = ones(1, length(f));
14 ft = f;
15 while length(ft)>1
16     nf = zeros(1, length(ft)/2);
17     nb = nf;
18
19     j = 1;
20     for i=1:2:length(ft)
21         [nf(j),nb(j)] = EBAR(ft(i), ft(i+1), bt(i), bt(i+1), thres, M);
22         j = j+1;
23     end
24     ft = nf;
25     bt = nb;
26 end
27 th = ft(1);

```

Β.3.13 Κώδικας εκτιμητή LBAR

```

1 function th = MEMARB_LBAR(f, M)
2 %MEMARB_LBAR Estimates the phase of the signal phase vector f and the LBAR
3 % function.
4 %
5 % MEMARB_LBAR(f,M) estimates phase error of the signal phase vector f, given
6 % in radians, in MPSK systems using the LBAR algorithm. length(f) must be a
7 % power of 2.
8
9 % Author(s): I. Papadopoulos
10 % Copyright 2005
11 % Revision : 1.3 Date : 2005/09/21
12
13 bt = ones(1, length(f));
14 ft = f;
15 while length(ft)>1
16     nf = zeros(1, length(ft)/2);
17     nb = nf;
18
19     j = 1;
20     for i=1:2:length(ft)
21         [nf(j),nb(j)] = LBAR(ft(i), ft(i+1), bt(i), bt(i+1), M);
22         j = j+1;
23     end
24     ft = nf;
25     bt = nb;
26 end
27 th = ft(1);

```

Β.3.14 Κώδικας εκτιμητή QLBAR

```

1 function th = MEMARB_QLBAR(f)

```

```

2 %MEMARB_QLBAR Estimates the phase of the signal phase vector f and the QLBAR
3 %               function.
4 %
5 % MEMARB_QLBAR(f) estimates phase error of the signal phase vector f, given in
6 % radians, in QPSK systems using the QLBAR algorithm. length(f) must be a
7 % power of 2.
8
9 % Author(s): I. Papadopoulos
10 % Copyright 2005
11 % Revision : 1.3 Date : 2005/09/21
12
13 bt = ones(1, length(f));
14 ft = f;
15 while length(ft)>1
16     nf = zeros(1, length(ft)/2);
17     nb = nf;
18
19     j = 1;
20     for i=1:2:length(ft)
21         [nf(j),nb(j)] = QLBAR(ft(i), ft(i+1), bt(i), bt(i+1));
22         j = j+1;
23     end
24     ft = nf;
25     bt = nb;
26 end
27 th = ft(1);

```

Β'.4 Κώδικας προσομοιώσεων εκτιμητών V&V

Β'.4.1 Κώδικας Ενότητας 5.3.1

```

1 clear;
2 close all;
3
4 %
5 % Compares instances of the V&V algorithm with different n.
6 %
7 % Author(s): I. Papadopoulos
8 % Copyright 2005
9 % Revision : 1.1 Date : 2005/08/01
10
11 %
12 % Simulation parameters
13 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
14
15 % frequency uncertainty
16 df = 10000;
17 % phase error
18 th = pi/10;
19 % signal-to-noise ratio
20 SNR = 4;
21 % number of symbols used (N preceding and N following)
22 N = 16;
23

```



```

24 %
25 % System constants
26 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
27
28 % number of symbols - 4 is QPSK (4-PSK)
29 M = 4;
30 % frequency
31 fc = 30*10^(9); % 30GHz
32 % nc
33 nc = 498;
34 % bit energy
35 Eb = 1;
36 % symbol energy
37 Es = (M/2)*Eb;
38 % symbol period
39 T = nc/fc; % about 166us
40 % Pulse width
41 A = sqrt(2*Es/T);
42 % symbol phase
43 sym = [0:(2*pi)/M:2*pi-0.01];
44
45 %
46 % Simulation
47 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
48
49 %
50 % Simulation constants
51 %
52 % nof samples
53 samples = 250;
54 % nof times
55 times = 300;
56 % Delta
57 D = (2*pi*df*T);
58 % variance
59 sigma2 = (A^2)/(10^(SNR/10));
60 % window
61 window = N*2;
62
63 %
64 % Generate signals
65 %
66 % we must have samples at both ends
67 ns = window + samples;
68 % symbols
69 m = round((M-1)*rand(times, ns));
70 % symbol phase
71 Z = m*( (2*pi)/M );
72 % complex AWGN noise
73 w = sqrt(sigma2/2)*(randn(times, ns) + j*randn(times, ns));
74 % frequency uncertainty
75 tmp = D*[(-N):1:(samples+N-1)];
76 for i=1:times
77     DF(i, :) = tmp;
78 end
79 % combined phase error and frequency offset
80 Terr = DF + th;
81 % transmitted signal
82 s = A*exp( j*Z );

```

```

83 % received signal
84 y = s.*exp( j*Terr ) + w;
85 % signal power
86 p = abs(y);
87 % symbol phase
88 f = angle(y);
89
90 %
91 % Estimating noise
92 %
93 for i=1:times
94     for k=1:samples
95         % estimations are stored for every sample in the same line
96         tempw = [k:1:k+window];
97         ptmp = p(i, tempw);
98         ftmp = f(i, tempw);
99         % V&V (as found in Catherine Morlet's PhD)
100        % F(p) = 1
101        thhat_m0(i, k) = viterbi_s(ptmp, ftmp, 0, M);
102        % F(p) = p^1
103        thhat_m1(i, k) = viterbi_s(ptmp, ftmp, 1, M);
104        % F(p) = p^2
105        thhat_m2(i, k) = viterbi_s(ptmp, ftmp, 2, M);
106        % F(p) = p^4
107        thhat_m4(i, k) = viterbi_s(ptmp, ftmp, 4, M);
108    end
109 end
110
111 %
112 % Demodulate
113 %
114 % useful index - everything else is garbage
115 index = [N+1:1:ns-N];
116 ss = s(:, index);
117 ys = y(:, index);
118 ms = m(:, index);
119 % phase noise removal
120 yhat_m0 = ys.*exp(-j*thhat_m0);
121 yhat_m1 = ys.*exp(-j*thhat_m1);
122 yhat_m2 = ys.*exp(-j*thhat_m2);
123 yhat_m4 = ys.*exp(-j*thhat_m4);
124 % find symbols
125 for i=1:times
126     for k=1:samples
127         mw(i,k) = symbol_testing(ys(i,k), sym);
128         mhat_m0(i,k) = symbol_testing(yhat_m0(i,k), sym);
129         mhat_m1(i,k) = symbol_testing(yhat_m1(i,k), sym);
130         mhat_m2(i,k) = symbol_testing(yhat_m2(i,k), sym);
131         mhat_m4(i,k) = symbol_testing(yhat_m4(i,k), sym);
132     end
133 end
134
135 %
136 % Calculate error
137 %
138 % residual symbol phase
139 Zs = Z(:, index);
140 res_w = (symbol_phase(ys, M) - Zs);
141 res_m0 = (symbol_phase(yhat_m0, M) - Zs);

```

```

142 res_m1 = (symbol_phase(yhat_m1, M) - Zs);
143 res_m2 = (symbol_phase(yhat_m2, M) - Zs);
144 res_m4 = (symbol_phase(yhat_m4, M) - Zs);
145 % residual symbol phase MSE
146 resmse_w = sum(res_w.^2, 1)/times;
147 resmse_m0 = sum(res_m0.^2, 1)/times;
148 resmse_m1 = sum(res_m1.^2, 1)/times;
149 resmse_m2 = sum(res_m2.^2, 1)/times;
150 resmse_m4 = sum(res_m4.^2, 1)/times;
151 % nof changed symbols
152 mdiff_w = 100*sum((ms-mw)≠0, 1)/times;
153 mdiff_m0 = 100*sum((ms-mhat_m0)≠0, 1)/times;
154 mdiff_m1 = 100*sum((ms-mhat_m1)≠0, 1)/times;
155 mdiff_m2 = 100*sum((ms-mhat_m2)≠0, 1)/times;
156 mdiff_m4 = 100*sum((ms-mhat_m4)≠0, 1)/times;
157
158 %
159 % Print MSE
160 %
161 fprintf('Samples_%d\tTimes_%d\n', samples, times);
162 fprintf('SNR:_%2.2fdb\tPhase_Error:_%2.2f\tDf:_%2.2fHz\n', ...
163     SNR, rad2deg(th), df);
164 fprintf('Residual_phase_MSE_without_V&V:_%f\n', ...
165     sum(resmse_w)/length(resmse_w));
166 fprintf('Residual_phase_MSE_of_V&V_F(p)_=1:_%f\n', ...
167     sum(resmse_m0)/length(resmse_m0));
168 fprintf('Residual_phase_MSE_of_V&V_F(p)_=p^1:_%f\n', ...
169     sum(resmse_m1)/length(resmse_m1));
170 fprintf('Residual_phase_MSE_of_V&V_F(p)_=p^2:_%f\n', ...
171     sum(resmse_m2)/length(resmse_m2));
172 fprintf('Residual_phase_MSE_of_V&V_F(p)_=p^4:_%f\n', ...
173     sum(resmse_m4)/length(resmse_m4));
174 fprintf('Missed_symbols_without_V&V:_%2.3f%%\n', ...
175     sum(mdiff_w)/length(mdiff_w));
176 fprintf('Missed_symbols_of_V&V_F(p)_=1:_%2.3f%%\n', ...
177     sum(mdiff_m0)/length(mdiff_m0));
178 fprintf('Missed_symbols_of_V&V_F(p)_=p^1:_%2.3f%%\n', ...
179     sum(mdiff_m1)/length(mdiff_m1));
180 fprintf('Missed_symbols_of_V&V_F(p)_=p^2:_%2.3f%%\n', ...
181     sum(mdiff_m2)/length(mdiff_m2));
182 fprintf('Missed_symbols_of_V&V_F(p)_=p^4:_%2.3f%%\n', ...
183     sum(mdiff_m4)/length(mdiff_m4));
184
185 %
186 % Visualize
187 %
188 si = [0:1:samples-1];
189 t = strcat('SNR=', num2str(SNR), 'dB\_\Delta', 'f=', num2str(df), ...
190     'Hz\_\theta=', num2str(th), '_window_size=', num2str(window));
191 % comparison of residual phase error
192 figure;
193 hold on;
194 plot(si, resmse_m0, 'r');
195 plot(si, resmse_m1, 'g');
196 plot(si, resmse_m2, 'b');
197 plot(si, resmse_m4, 'm');
198 ylabel('residual_phase_(MSE)');
199 xlabel('n');
200 legend('V&V_F(p)=1', 'V&V_F(p)=p^1', 'V&V_F(p)=p^2', 'V&V_F(p)=p^4', ...

```

```

201     'Location', 'BestOutside');
202 title(t);
203 % # of changed symbols
204 figure;
205 hold on;
206 plot(si, mdiff_w, 'k');
207 plot(si, mdiff_m0, 'r');
208 plot(si, mdiff_m1, 'g');
209 plot(si, mdiff_m2, 'b');
210 plot(si, mdiff_m4, 'm');
211 ylabel('percentage_of_missed_symbols');
212 xlabel('n');
213 legend('no_V&V', 'V&V-F(p)=1', 'V&V-F(p)=p^1', 'V&V-F(p)=p^2', ...
214        'V&V-F(p)=p^4', 'Location', 'BestOutside');
215 title(t);

```

```

1 clear;
2 close all;
3
4 %
5 % Compares instances of the V&V algorithm with different n and various SNRs.
6 %
7 % Author(s): I. Papadopoulos
8 % Copyright 2005
9 % Revision: 1.1 Date: 2005/08/01
10
11 %
12 % Simulation parameters
13 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
14
15 % frequency uncertainty
16 df = 100;
17 % phase error
18 th = pi/10;
19 % signal-to-noise ratio
20 SNR = [0:1:15];
21 % number of symbols used (N preceding and N following)
22 N = 16;
23
24 %
25 % System constants
26 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
27
28 % number of symbols - 4 is QPSK (4-PSK)
29 M = 4;
30 % frequency
31 fc = 30*10^(9); % 30GHz
32 % nc
33 nc = 498;
34 % bit energy
35 Eb = 1;
36 % symbol energy
37 Es = (M/2)*Eb;
38 % symbol period
39 T = nc/fc; % about 166us
40 % Pulse width
41 A = sqrt(2*Es/T);

```

```

42 % symbol phase
43 sym = [0:(2*pi)/M:2*pi-0.01];
44
45 %
46 % Simulation
47 %%%%%%%%%%%
48
49 %
50 % Simulation constants
51 %
52 % nof samples
53 samples = 500;
54 % nof times
55 times = 200;
56 % Delta
57 D = (2*pi*df*T);
58 % variance
59 sigma2 = (A^2)./(10.^(SNR/10));
60 % window
61 window = N*2;
62
63 %
64 % Generate signals
65 %
66 % we must have samples at both ends
67 ns = window + samples;
68 % symbols
69 m = round((M-1)*rand(times, ns));
70 % symbol phase
71 Z = m*( (2*pi)/M );
72 % frequency uncertainty
73 tmp = D*[(-N):1:(samples+N-1)];
74 for i=1:times
75     DF(i, :) = tmp;
76 end
77 % combined phase error and frequency offset
78 Terr = DF + th;
79 % transmitted signal
80 s = A*exp( j*Z );
81 for snri=1:length(SNR)
82     % complex AWGN noise
83     w = sqrt(sigma2(snri)/2)*(randn(times, ns) + j*randn(times, ns));
84     % received signal
85     y = s.*exp( j*Terr ) + w;
86     % signal power
87     p = abs(y);
88     % symbol phase
89     f = angle(y);
90     %
91     % Estimating noise
92     %
93     for i=1:times
94         for k=1:samples
95             % estimations are stored for every sample in the same line
96             tempw = [k:1:k+window];
97             ptmp = p(i, tempw);
98             ftmp = f(i, tempw);
99             % V&V (as found in Catherine Morlet's PhD)
100            % F(p) = 1

```

```

101         thhat_m0(i, k) = viterbi_s(ptmp, ftmp, 0, M);
102         % F(p) = p^1
103         thhat_m1(i, k) = viterbi_s(ptmp, ftmp, 1, M);
104         % F(p) = p^2
105         thhat_m2(i, k) = viterbi_s(ptmp, ftmp, 2, M);
106         % F(p) = p^4
107         thhat_m4(i, k) = viterbi_s(ptmp, ftmp, 4, M);
108     end
109 end
110 %
111 % Demodulate
112 %
113 % useful index - everything else is garbage
114 index = [N+1:1:ns-N];
115 ss = s(:, index);
116 ys = y(:, index);
117 ms = m(:, index);
118 % phase noise removal
119 yhat_m0 = ys.*exp(-j*thhat_m0);
120 yhat_m1 = ys.*exp(-j*thhat_m1);
121 yhat_m2 = ys.*exp(-j*thhat_m2);
122 yhat_m4 = ys.*exp(-j*thhat_m4);
123 % find symbols
124 for i=1:times
125     for k=1:samples
126         mw(i,k) = symbol_testing(ys(i,k), sym);
127         mhat_m0(i,k) = symbol_testing(yhat_m0(i,k), sym);
128         mhat_m1(i,k) = symbol_testing(yhat_m1(i,k), sym);
129         mhat_m2(i,k) = symbol_testing(yhat_m2(i,k), sym);
130         mhat_m4(i,k) = symbol_testing(yhat_m4(i,k), sym);
131     end
132 end
133 %
134 % Calculate error
135 %
136 % residual symbol phase
137 Zs = Z(:, index);
138 res_w = (symbol_phase(ys, M) - Zs);
139 res_m0 = (symbol_phase(yhat_m0, M) - Zs);
140 res_m1 = (symbol_phase(yhat_m1, M) - Zs);
141 res_m2 = (symbol_phase(yhat_m2, M) - Zs);
142 res_m4 = (symbol_phase(yhat_m4, M) - Zs);
143 % residual symbol phase MSE
144 resmse_w(snri) = sum(sum(res_w.^2, 1)/times)/samples;
145 resmse_m0(snri) = sum(sum(res_m0.^2, 1)/times)/samples;
146 resmse_m1(snri) = sum(sum(res_m1.^2, 1)/times)/samples;
147 resmse_m2(snri) = sum(sum(res_m2.^2, 1)/times)/samples;
148 resmse_m4(snri) = sum(sum(res_m4.^2, 1)/times)/samples;
149 % nof changed symbols
150 mdiff_w(snri) = sum(100*sum( ((ms-mw)~=0), 1)/times)/samples;
151 mdiff_m0(snri) = sum(100*sum( ((ms-mhat_m0)~=0), 1)/times)/samples;
152 mdiff_m1(snri) = sum(100*sum( ((ms-mhat_m1)~=0), 1)/times)/samples;
153 mdiff_m2(snri) = sum(100*sum( ((ms-mhat_m2)~=0), 1)/times)/samples;
154 mdiff_m4(snri) = sum(100*sum( ((ms-mhat_m4)~=0), 1)/times)/samples;
155 end
156
157 %
158 % Visualize
159 %

```

```

160 t = strcat('variable_SNR\Delta', 'f=', num2str(df), 'Hz\theta=', ...
161     num2str(th), '\window_size=', num2str(window));
162 % # of changed symbols
163 figure;
164 hold on;
165 plot(SNR, mdiff_w, 'k');
166 plot(SNR, mdiff_m0, 'r');
167 plot(SNR, mdiff_m1, 'g');
168 plot(SNR, mdiff_m2, 'b');
169 plot(SNR, mdiff_m4, 'm');
170 ylabel('percentage_of_missed_symbols');
171 xlabel('SNR');
172 legend('no_V&V', 'V&V_F(p)=1', 'V&V_F(p)=p^1', 'V&V_F(p)=p^2', ...
173     'V&V_F(p)=p^4', 'Location', 'BestOutside');
174 title(t);

```

```

1 clear;
2 close all;
3
4 %
5 % Compares instances of the V&V algorithm with different n and various windows.
6 %
7 % Author(s): I. Papadopoulos
8 % Copyright 2005
9 % Revision: 1.2 Date: 2005/08/12
10
11 %
12 % Simulation parameters
13 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
14
15 % frequency uncertainty
16 df = 100;
17 % phase error
18 th = pi/10;
19 % signal-to-noise ratio
20 SNR = 0;
21 % number of symbols used (N preceding and N following)
22 N = [2,4,8,16,32,64,128];
23
24 %
25 % System constants
26 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
27
28 % number of symbols - 4 is QPSK (4-PSK)
29 M = 4;
30 % frequency
31 fc = 30*10^(9); % 30GHz
32 % nc
33 nc = 498;
34 % bit energy
35 Eb = 1;
36 % symbol energy
37 Es = (M/2)*Eb;
38 % symbol period
39 T = nc/fc; % about 166us
40 % Pulse width
41 A = sqrt(2*Es/T);

```

```

42 % symbol phase
43 sym = [0:(2*pi)/M:2*pi-0.01];
44
45 %
46 % Simulation
47 %%%%%%%%%%
48
49 %
50 % Simulation constants
51 %
52 % nof samples
53 samples = 500;
54 % nof times
55 times = 200;
56 % Delta
57 D = (2*pi*df*T);
58 % variance
59 sigma2 = (A^2)/(10.^(SNR/10));
60 % window
61 window = N*2;
62
63 %
64 % Generate signals
65 %
66 % maximum samples
67 ns = window(length(window))+samples;
68 % symbols
69 m = round((M-1)*rand(times, ns));
70 % symbol phase
71 Z = m*(2*pi)/M;
72 % complex AWGN noise
73 w = sqrt(sigma2/2)*(randn(times, ns) + j*randn(times, ns));
74 % frequency uncertainty
75 tmp = D*[(-N(length(window))):1:(samples+N(length(window))-1)];
76 for i=1:times
77     DF(i, :) = tmp;
78 end
79 % combined phase error and frequency offset
80 Terr = DF + th;
81 % transmitted signal
82 s = A*exp(j*Z);
83 % received signal
84 y = s.*exp(j*Terr) + w;
85 % signal power
86 p = abs(y);
87 % symbol phase
88 f = angle(y);
89 % useful index - everything else is garbage
90 index = [N(length(N))+1:1:samples+N(length(N))];
91 for wini=1:length(window)
92     % useful signals
93     us = [N(length(N))+1-N(wini):1:samples+N(length(N))+N(wini)];
94     ft = f(:, us);
95     pt = p(:, us);
96     %
97     % Estimating noise
98     %
99     for i=1:times
100         for k=1:samples

```



```

101     % estimations are stored for every sample in the same line
102     tempw = [k:1:k+window(wini)];
103     ptmp = pt(i, tempw);
104     ftmp = ft(i, tempw);
105
106     % V&V (as found in Catherine Morlet's PhD)
107     % F(p) = 1
108     thhat_m0(i, k) = viterbi_s(ptmp, ftmp, 0, M);
109     % F(p) = p^1
110     thhat_m1(i, k) = viterbi_s(ptmp, ftmp, 1, M);
111     % F(p) = p^2
112     thhat_m2(i, k) = viterbi_s(ptmp, ftmp, 2, M);
113     % F(p) = p^4
114     thhat_m4(i, k) = viterbi_s(ptmp, ftmp, 4, M);
115     end
116 end
117 %
118 % Demodulate
119 %
120 ss = s(:, index);
121 ys = y(:, index);
122 ms = m(:, index);
123 % phase noise removal
124 yhat_m0 = ys.*exp(-j*thhat_m0);
125 yhat_m1 = ys.*exp(-j*thhat_m1);
126 yhat_m2 = ys.*exp(-j*thhat_m2);
127 yhat_m4 = ys.*exp(-j*thhat_m4);
128 % find symbols
129 for i=1:times
130     for k=1:samples
131         mw(i,k) = symbol_testing(ys(i,k), sym);
132         mhat_m0(i,k) = symbol_testing(yhat_m0(i,k), sym);
133         mhat_m1(i,k) = symbol_testing(yhat_m1(i,k), sym);
134         mhat_m2(i,k) = symbol_testing(yhat_m2(i,k), sym);
135         mhat_m4(i,k) = symbol_testing(yhat_m4(i,k), sym);
136     end
137 end
138 %
139 % Calculate error
140 %
141 % residual symbol phase
142 Zs = Z(:, index);
143 res_w = (symbol_phase(ys, M) - Zs);
144 res_m0 = (symbol_phase(yhat_m0, M) - Zs);
145 res_m1 = (symbol_phase(yhat_m1, M) - Zs);
146 res_m2 = (symbol_phase(yhat_m2, M) - Zs);
147 res_m4 = (symbol_phase(yhat_m4, M) - Zs);
148 % residual symbol phase MSE
149 resmse_w(wini) = sum(sum(res_w.^2, 1)/times)/samples;
150 resmse_m0(wini) = sum(sum(res_m0.^2, 1)/times)/samples;
151 resmse_m1(wini) = sum(sum(res_m1.^2, 1)/times)/samples;
152 resmse_m2(wini) = sum(sum(res_m2.^2, 1)/times)/samples;
153 resmse_m4(wini) = sum(sum(res_m4.^2, 1)/times)/samples;
154 % nof changed symbols
155 mdiff_w(wini) = sum(100*sum( ((ms-mw)~=0), 1)/times)/samples;
156 mdiff_m0(wini) = sum(100*sum( ((ms-mhat_m0)~=0), 1)/times)/samples;
157 mdiff_m1(wini) = sum(100*sum( ((ms-mhat_m1)~=0), 1)/times)/samples;
158 mdiff_m2(wini) = sum(100*sum( ((ms-mhat_m2)~=0), 1)/times)/samples;
159 mdiff_m4(wini) = sum(100*sum( ((ms-mhat_m4)~=0), 1)/times)/samples;

```

```

160 end
161
162 %
163 % Visualize
164 %
165 t = strcat('SNR=', num2str(SNR), 'dB\Delta', 'f=', num2str(df), ...,
166           'Hz\theta=', num2str(th), '\window_size_variable');
167 % comparison of residual phase error
168 figure;
169 hold on;
170 plot(window, resmse_m0, 'r');
171 plot(window, resmse_m1, 'g');
172 plot(window, resmse_m2, 'b');
173 plot(window, resmse_m4, 'm');
174 ylabel('residual_phase_(MSE)');
175 xlabel('window_size');
176 legend('V&V_F(p)=1', 'V&V_F(p)=p^1', 'V&V_F(p)=p^2', 'V&V_F(p)=p^4', ...
177        'Location', 'BestOutside');
178 title(t);
179 % # of changed symbols
180 figure;
181 hold on;
182 plot(window, mdiff_w, 'k');
183 plot(window, mdiff_m0, 'r');
184 plot(window, mdiff_m1, 'g');
185 plot(window, mdiff_m2, 'b');
186 plot(window, mdiff_m4, 'm');
187 ylabel('percentage_of_missed_symbols');
188 xlabel('window_size');
189 legend('no_V&V', 'V&V_F(p)=1', 'V&V_F(p)=p^1', 'V&V_F(p)=p^2', ...
190        'V&V_F(p)=p^4', 'Location', 'BestOutside');
191 title(t);

```

```

1 clear;
2 close all;
3
4 %
5 % Compares instances of the V&V algorithm with different n and various theta.
6 %
7 % Author(s): I. Papadopoulos
8 % Copyright 2005
9 % Revision: 1.1 Date: 2005/09/15
10
11 %
12 % Simulation parameters
13 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
14
15 % frequency uncertainty
16 df = 10000;
17 % phase error
18 th = [0:pi/32:pi/4];
19 % signal-to-noise ratio
20 SNR = 0;
21 % number of symbols used (N preceeding and N following)
22 N = 16;
23
24 %

```

```

25 % System constants
26 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
27
28 % number of symbols - 4 is QPSK (4-PSK)
29 M = 4;
30 % frequency
31 fc = 30*10^(9); % 30GHz
32 % nc
33 nc = 498;
34 % bit energy
35 Eb = 1;
36 % symbol energy
37 Es = (M/2)*Eb;
38 % symbol period
39 T = nc/fc; % about 166us
40 % Pulse width
41 A = sqrt(2*Es/T);
42 % symbol phase
43 sym = [0:(2*pi)/M:2*pi-0.01];
44
45 %
46 % Simulation
47 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
48
49 %
50 % Simulation constants
51 %
52 % nof samples
53 samples = 500;
54 % nof times
55 times = 200;
56 % Delta
57 D = (2*pi*df*T);
58 % variance
59 sigma2 = (A^2)./(10.^(SNR/10));
60 % window
61 window = N*2;
62
63 %
64 % Generate signals
65 %
66 % we must have samples at both ends
67 ns = window + samples;
68 % symbols
69 m = round((M-1)*rand(times, ns));
70 % symbol phase
71 Z = m*( (2*pi)/M );
72 % frequency uncertainty
73 tmp = D*[(-N):1:(samples+N-1)];
74 for i=1:times
75     DF(i, :) = tmp;
76 end
77 % complex AWGN noise
78 w = sqrt(sigma2/2)*(randn(times, ns) + j*randn(times, ns));
79 % transmitted signal
80 s = A*exp( j*Z );
81 for thi=1:length(th)
82     % combined phase error and frequency offset
83     Terr = DF + th(thi);

```

```

84     % received signal
85     y = s.*exp( j*Terr ) + w;
86     % signal power
87     p = abs(y);
88     % symbol phase
89     f = angle(y);
90
91     %%
92     %% Estimating noise
93     %%
94     for i=1:times
95         for k=1:samples
96             % estimations are stored for every sample in the same line
97             tempw = [k:1:k+window];
98             ptmp = p(i, tempw);
99             ftmp = f(i, tempw);
100            % V&V
101            % F(p) = 1
102            thhat_m0(i, k) = viterbi_s(ptmp, ftmp, 0, M);
103            % F(p) = p^1
104            thhat_m1(i, k) = viterbi_s(ptmp, ftmp, 1, M);
105            % F(p) = p^2
106            thhat_m2(i, k) = viterbi_s(ptmp, ftmp, 2, M);
107            % F(p) = p^4
108            thhat_m4(i, k) = viterbi_s(ptmp, ftmp, 4, M);
109        end
110    end
111
112    %%
113    % Demodulate
114    %
115    % useful index - everything else is garbage
116    index = [N+1:1:ns-N];
117    ss = s(:, index);
118    ys = y(:, index);
119    ms = m(:, index);
120    % phase noise removal
121    yhat_m0 = ys.*exp(-j*thhat_m0);
122    yhat_m1 = ys.*exp(-j*thhat_m1);
123    yhat_m2 = ys.*exp(-j*thhat_m2);
124    yhat_m4 = ys.*exp(-j*thhat_m4);
125    % find symbols
126    for i=1:times
127        for k=1:samples
128            mw(i,k) = symbol_testing(ys(i,k), sym);
129            mhat_m0(i,k) = symbol_testing(yhat_m0(i,k), sym);
130            mhat_m1(i,k) = symbol_testing(yhat_m1(i,k), sym);
131            mhat_m2(i,k) = symbol_testing(yhat_m2(i,k), sym);
132            mhat_m4(i,k) = symbol_testing(yhat_m4(i,k), sym);
133        end
134    end
135
136    %%
137    % Calculate error
138    %
139    % residual symbol phase
140    Zs = Z(:, index);
141    res_w = (symbol_phase(ys, M) - Zs);
142    res_m0 = (symbol_phase(yhat_m0, M) - Zs);

```

```

143     res_m1 = (symbol_phase(yhat_m1, M) - Zs);
144     res_m2 = (symbol_phase(yhat_m2, M) - Zs);
145     res_m4 = (symbol_phase(yhat_m4, M) - Zs);
146     % residual symbol phase MSE
147     resmse_w(thi) = sum(sum(res_w.^2, 1)/times)/samples;
148     resmse_m0(thi) = sum(sum(res_m0.^2, 1)/times)/samples;
149     resmse_m1(thi) = sum(sum(res_m1.^2, 1)/times)/samples;
150     resmse_m2(thi) = sum(sum(res_m2.^2, 1)/times)/samples;
151     resmse_m4(thi) = sum(sum(res_m4.^2, 1)/times)/samples;
152     % nof changed symbols
153     mdiff_w(thi) = sum(100*sum( (ms-mw)~=0), 1)/times)/samples;
154     mdiff_m0(thi) = sum(100*sum( (ms-mhat_m0)~=0), 1)/times)/samples;
155     mdiff_m1(thi) = sum(100*sum( (ms-mhat_m1)~=0), 1)/times)/samples;
156     mdiff_m2(thi) = sum(100*sum( (ms-mhat_m2)~=0), 1)/times)/samples;
157     mdiff_m4(thi) = sum(100*sum( (ms-mhat_m4)~=0), 1)/times)/samples;
158 end
159
160 %
161 % Visualize
162 %
163 t = strcat('SNR=', num2str(SNR), 'dB\Delta', 'f=', num2str(df), 'Hz', ...
164     '\variable\theta', '\window_size=', num2str(window));
165 % comparison of residual phase error
166 figure;
167 hold on;
168 plot(th, resmse_m0, 'r');
169 plot(th, resmse_m1, 'g');
170 plot(th, resmse_m2, 'b');
171 plot(th, resmse_m4, 'm');
172 ylabel('residual_phase_(MSE)');
173 xlabel('\theta_(radians)');
174 legend('V&V_LF(p)=1', 'V&V_LF(p)=p^1', 'V&V_LF(p)=p^2', 'V&V_LF(p)=p^4', ...
175     'Location', 'BestOutside');
176 title(t);
177 % # of changed symbols
178 figure;
179 hold on;
180 plot(th, mdiff_w, 'k');
181 plot(th, mdiff_m0, 'r');
182 plot(th, mdiff_m1, 'g');
183 plot(th, mdiff_m2, 'b');
184 plot(th, mdiff_m4, 'm');
185 ylabel('percentage_of_missed_symbols');
186 xlabel('\theta_(radians)');
187 legend('no_V&V', 'V&V_LF(p)=1', 'V&V_LF(p)=p^1', 'V&V_LF(p)=p^2', ...
188     'V&V_LF(p)=p^4', 'Location', 'BestOutside');
189 title(t);

```

B.4.2 Κώδικας Ενότητας 5.3.2

```

1 clear;
2 close all;
3
4 %
5 % Compares V&V, optimal V&V and Multipass V&V.
6 %

```

```

7 % Author(s): I. Papadopoulos
8 % Copyright 2005
9 % Revision : 1.1 Date : 2005/09/01
10
11 %
12 % Simulation parameters
13 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
14
15 % frequency uncertainty
16 df = 100;
17 % phase error
18 th = pi/10;
19 % signal-to-noise ratio
20 SNR = 0;
21 % number of symbols used (N preceding and N following)
22 N = 16;
23
24 %
25 % System constants
26 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
27
28 % number of symbols - 4 is QPSK (4-PSK)
29 M = 4;
30 % frequency
31 fc = 30*10^(9); % 30GHz
32 % nc
33 nc = 498;
34 % bit energy
35 Eb = 1;
36 % symbol energy
37 Es = (M/2)*Eb;
38 % symbol period
39 T = nc/fc; % about 166us
40 % Pulse width
41 A = sqrt(2*Es/T);
42 % symbol phase
43 sym = [0:(2*pi)/M:2*pi-0.01];
44
45 %
46 % Simulation
47 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
48
49 %
50 % Simulation constants
51 %
52 % nof samples
53 samples = 300;
54 % nof times
55 times = 200;
56 % Delta
57 D = (2*pi*df*T);
58 % variance
59 sigma2 = (A^2)/(10^(SNR/10));
60 % window
61 window = N*2;
62
63 %
64 % Generate signals
65 %

```

```

66
67 % we must have samples at both ends - double of them for the Multipass V&V
68 ns = window + samples + window;
69 % symbols
70 m = round((M-1)*randn(times, ns));
71 % symbol phase
72 Z = m*( 2*pi)/M );
73 % complex AWGN noise
74 w = sqrt(sigma2/2)*(randn(times, ns) + j*randn(times, ns));
75 % frequency uncertainty
76 tmp = D*[(-window):1:(samples+window-1)];
77 for i=1:times
78     DF(i, :) = tmp;
79 end
80 % combined phase error and frequency offset
81 Terr = DF + th;
82 % transmitted signal
83 s = A*exp( j*Z );
84 % received signal
85 y = s.*exp( j*Terr ) + w;
86 % signal power
87 p = abs(y);
88 % symbol phase
89 f = angle(y);
90
91 %
92 % Estimating noise
93 %
94 all_samples = samples+window;
95 for i=1:times
96     for k=1:all_samples
97         % estimations are stored for every sample in the same line
98         tempw = [k:1:k+window];
99         ptmp = p(i, tempw);
100        ftmp = f(i, tempw);
101
102        % Optimal V&V estimation
103        thhat_o(i, k) = viterbio(y(i,tempw), Z(i,tempw));
104        % V&V F(p) = p^2
105        thhat_m2(i, k) = viterbis(ptmp, ftmp, 2, M);
106    end
107 end
108
109 %
110 % Demodulate
111 %
112 % useful index
113 index = [N+1:1:ns-N];
114 ss = s(:, index);
115 ys = y(:, index);
116 ms = m(:, index);
117 % phase noise removal
118 yhat_o = ys.*exp(-j*thhat_o);
119 yhat_m2 = ys.*exp(-j*thhat_m2);
120 % find symbols
121 for i=1:times
122     for k=1:all_samples
123         mw(i,k) = symbol_testing(ys(i,k), sym);
124         mhat_o(i,k) = symbol_testing(yhat_o(i,k), sym);

```

```

125         mhat_m2(i,k) = symbol_testing(yhat_m2(i,k), sym);
126     end
127 end
128
129 %
130 % Run multipass phase estimation - 1 pass
131 %
132 Zest = mhat_m2*( (2*pi)/M ); % using V&V  $F(p) = p^2$ 
133 for i=1:times
134     for k=1:samples
135         tempw = [k:1:k+window];
136         thhat_mpl(i,k) = viterbi_o(ys(i,tempw), Zest(i,tempw));
137     end
138 end
139 % find symbols
140 rindex = [N+1:1:samples+N];
141 yr = ys(:, rindex);
142 yhat_mpl = yr.*exp(-j*thhat_mpl);
143 for i=1:times
144     for k=1:samples
145         mhat_mpl(i,k) = symbol_testing(yhat_mpl(i,k), sym);
146     end
147 end
148
149 %
150 % Calculate error - using some samples, the rest is garbage and more symbols
151 % for multi-phase estimation
152 %
153 % residual symbol phase
154 Zs = Z(:, [window+1:1:ns-window]);
155 res_w = (symbol_phase(ys(:, rindex), M) - Zs);
156 res_o = (symbol_phase(yhat_o(:, rindex), M) - Zs);
157 res_m2 = (symbol_phase(yhat_m2(:, rindex), M) - Zs);
158 res_mpl = (symbol_phase(yhat_mpl, M) - Zs);
159 % residual symbol phase MSE
160 resmse_w = sum(res_w.^2, 1)/times;
161 resmse_o = sum(res_o.^2, 1)/times;
162 resmse_m2 = sum(res_m2.^2, 1)/times;
163 resmse_mpl = sum(res_mpl.^2, 1)/times;
164 % percentage of missed symbols
165 mdiff_w = 100*sum( ((ms(:,rindex)-mw(:,rindex))≠0), 1)/times;
166 mdiff_o = 100*sum( ((ms(:,rindex)-mhat_o(:,rindex))≠0), 1)/times;
167 mdiff_m2 = 100*sum( ((ms(:,rindex)-mhat_m2(:,rindex))≠0), 1)/times;
168 mdiff_mpl = 100*sum( ((ms(:,rindex)-mhat_mpl)≠0), 1)/times;
169
170 %
171 % Print MSE
172 %
173 fprintf('Samples_%d\tTimes_%d\n', samples, times);
174 fprintf('SNR:_%2.2fdb\tPhase_error:_%2.2f\tDf:_%2.2fHz\n', ...
175     SNR, rad2deg(th), df);
176 fprintf('Residual_phase_MSE_without_V&V: %f\n', ...
177     sum(resmse_w)/length(resmse_w));
178 fprintf('Residual_phase_MSE_of_optimal_V&V: %f\n', ...
179     sum(resmse_o)/length(resmse_o));
180 fprintf('Residual_phase_MSE_of_V&V_F(p) = p^2: %f\n', ...
181     sum(resmse_m2)/length(resmse_m2));
182 fprintf('Residual_phase_MSE_of_Multipass_V&V_F(p) = p^2: %f\n', ...
183     sum(resmse_mpl)/length(resmse_mpl));

```



```

184 fprintf('Missed_symbols_without_V&V: %2.3f%%\n', ...
185         sum(mdiff_w)/length(mdiff_w));
186 fprintf('Missed_symbols_of_optimal_V&V: %2.3f%%\n', ...
187         sum(mdiff_o)/length(mdiff_o));
188 fprintf('Missed_symbols_of_V&V_LF(p)=p^2: %2.3f%%\n', ...
189         sum(mdiff_m2)/length(mdiff_m2));
190 fprintf('Missed_symbols_of_Multipass_V&V_LF(p)=p^2: %2.3f%%\n', ...
191         sum(mdiff_mp1)/length(mdiff_mp1));
192
193 %
194 % Visualize
195 %
196 si = [0:1:samples-1];
197 t = strcat('SNR=', num2str(SNR), 'dB \Delta', 'f=', num2str(df), ...
198         'Hz \theta=', num2str(th), ' \window_size=', num2str(window));
199 % residual phase error
200 figure;
201 hold on;
202 plot(si, resmse_w, 'k');
203 plot(si, resmse_m2, 'b');
204 plot(si, resmse_mp1, 'c');
205 plot(si, resmse_o, 'r');
206 ylabel('residual_phase_(MSE)');
207 xlabel('n');
208 legend('no_V&V', 'V&V_LF(p)=p^2', 'Multipass_V&V(1iteration)', ...
209        'optimal_V&V', 'Location', 'BestOutside');
210 title(t);
211 % # of changed symbols
212 figure;
213 hold on;
214 plot(si, mdiff_w, 'k');
215 plot(si, mdiff_m2, 'b');
216 plot(si, mdiff_mp1, 'c');
217 plot(si, mdiff_o, 'r');
218 ylabel('percentage_of_missed_symbols');
219 xlabel('n');
220 legend('no_V&V', 'V&V_LF(p)=p^2', 'Multipass_V&V(1iteration)', ...
221        'optimal_V&V', 'Location', 'BestOutside');
222 title(t);

```

```

1 clear;
2 close all;
3
4 %
5 % Compares instances of the V&V, optimal V&V and Multipass V&V and various SNRs.
6 %
7 % Author(s): I. Papadopoulos
8 % Copyright 2005
9 % Revision: 1.1 Date: 2005/08/01
10
11 %
12 % Simulation parameters
13 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
14
15 % frequency uncertainty
16 df = 10000;
17 % phase error

```

```

18 th = pi/10;
19 % signal-to-noise ratio
20 SNR = [0:1:15];
21 % number of symbols used (N preceeding and N following)
22 N = 16;
23
24 %
25 % System constants
26 %%%%%%%%%%%
27
28 % number of symbols - 4 is QPSK (4-PSK)
29 M = 4;
30 % frequency
31 fc = 30*10^(9); % 30GHz
32 % nc
33 nc = 498;
34 % bit energy
35 Eb = 1;
36 % symbol energy
37 Es = (M/2)*Eb;
38 % symbol period
39 T = nc/fc; % about 166us
40 % Pulse width
41 A = sqrt(2*Es/T);
42 % symbol phase
43 sym = [0:(2*pi)/M:2*pi-0.01];
44
45 %
46 % Simulation
47 %%%%%%%%%%%
48
49 %
50 % Simulation constants
51 %
52 % nof samples
53 samples = 500;
54 % nof times
55 times = 200;
56 % Delta
57 D = (2*pi*df*T);
58 % variance
59 sigma2 = (A^2)./(10.^(SNR/10));
60 % window
61 window = N*2;
62
63 %
64 % Generate signals
65 %
66 % we must have samples at both ends - double of them for the Multipass V&V
67 ns = window + samples + window;
68 % symbols
69 m = round((M-1)*rand(times, ns));
70 % symbol phase
71 Z = m*( (2*pi)/M );
72 % frequency uncertainty
73 tmp = D*[(-window):1:(samples+window-1)];
74 for i=1:times
75     DF(i, :) = tmp;
76 end

```

```

77 % combined phase error and frequency offset
78 Terr = DF + th;
79 % transmitted signal
80 s = A*exp( j*Z );
81 all_samples = samples>window;
82 for snri=1:length(SNR)
83     % complex AWGN noise
84     w = sqrt(sigma2(snri)/2)*(randn(times, ns) + j*randn(times, ns));
85     % received signal
86     y = s.*exp( j*Terr ) + w;
87     % signal power
88     p = abs(y);
89     % symbol phase
90     f = angle(y);
91
92     %
93     % Estimating noise
94     %
95     for i=1:times
96         for k=1:all_samples
97             % estimations are stored for every sample in the same line
98             tempw = [k:1:k>window];
99             ptmp = p(i, tempw);
100            ftmp = f(i, tempw);
101            % Optimal V&V estimation
102            thhat_o(i, k) = viterbi_o(y(i,tempw), Z(i,tempw));
103            % V&V F(p) = p^2
104            thhat_m2(i, k) = viterbi_s(ptmp, ftmp, 2, M);
105        end
106    end
107
108    %
109    % Demodulate
110    %
111    % useful index - everything else is garbage
112    index = [N+1:1:ns-N];
113    ss = s(:, index);
114    ys = y(:, index);
115    ms = m(:, index);
116    % phase noise removal
117    yhat_o = ys.*exp(-j*thhat_o);
118    yhat_m2 = ys.*exp(-j*thhat_m2);
119    % find symbols
120    for i=1:times
121        for k=1:all_samples
122            mw(i,k) = symbol_testing(ys(i,k), sym);
123            mhat_o(i,k) = symbol_testing(yhat_o(i,k), sym);
124            mhat_m2(i,k) = symbol_testing(yhat_m2(i,k), sym);
125        end
126    end
127
128    %
129    % Run multi step phase estimation - 1st iteration
130    %
131    Zest = mhat_m2*( (2*pi)/M ); % using V&V F(p) = p^2
132    for i=1:times
133        for k=1:samples
134            tempw = [k:1:k>window];
135            thhat_mp1(i,k) = viterbi_o(ys(i,tempw), Zest(i,tempw));

```

```

136     end
137     end
138     % find symbols
139     rindex = [N+1:1:samples+N];
140     yr = ys(:, rindex);
141     yhat_mpl = yr.*exp(-j*thhat_mpl);
142     for i=1:times
143         for k=1:samples
144             mhat_mpl(i,k) = symbol_testing(yhat_mpl(i,k), sym);
145         end
146     end
147
148     %
149     % Calculate error
150     %
151     % residual symbol phase
152     Zs = Z(:, [window+1:1:ns-window]);
153     res_w = (symbol_phase(ys(:, rindex), M) - Zs);
154     res_o = (symbol_phase(yhat_o(:, rindex), M) - Zs);
155     res_m2 = (symbol_phase(yhat_m2(:, rindex), M) - Zs);
156     res_mpl = (symbol_phase(yhat_mpl, M) - Zs);
157     % residual symbol phase MSE
158     resmse_w(snri) = sum(sum(res_w.^2, 1)/times)/samples;
159     resmse_o(snri) = sum(sum(res_o.^2, 1)/times)/samples;
160     resmse_m2(snri) = sum(sum(res_m2.^2, 1)/times)/samples;
161     resmse_mpl(snri) = sum(sum(res_mpl.^2, 1)/times)/samples;
162     % nof changed symbols
163     mdiff_w(snri) = ...
164         sum(100*sum( ((ms(:,rindex)-mw(:,rindex))≠0), 1)/times)/samples;
165     mdiff_o(snri) = ...
166         sum(100*sum( ((ms(:,rindex)-mhat_o(:,rindex))≠0), 1)/times)/samples;
167     mdiff_m2(snri) = ...
168         sum(100*sum( ((ms(:,rindex)-mhat_m2(:,rindex))≠0), 1)/times)/samples;
169     mdiff_mpl(snri) = ...
170         sum(100*sum( ((ms(:,rindex)-mhat_mpl)≠0), 1)/times)/samples;
171 end
172
173 %
174 % Visualize
175 %
176 %
177 t = strcat('variable_SNR_\Delta', 'f=', num2str(df), 'Hz_\theta=', ...,
178     num2str(th), '_window_size=', num2str(window));
179 % residual phase error
180 figure;
181 hold on;
182 plot(SNR, resmse_m2, 'b');
183 plot(SNR, resmse_mpl, 'c');
184 plot(SNR, resmse_o, 'r');
185 ylabel('residual_phase_(MSE)');
186 xlabel('SNR_(dB)');
187 legend('V&V_LF(p)=p^2', 'Multipass_V&V_L(1iteration)', 'optimal_V&V', ...
188     'Location', 'BestOutside');
189 title(t);
190 % # of changed symbols
191 figure;
192 hold on;
193 plot(SNR, mdiff_w, 'm');
194 plot(SNR, mdiff_m2, 'b');

```

```

195 plot(SNR, mdiff_mpl, 'c');
196 plot(SNR, mdiff_o, 'r:');
197 ylabel('percentage_of_missed_symbols');
198 xlabel('SNR_(dB)');
199 legend('no_V&V', 'V&V_F(p)=p^2', 'Multipass_V&V_(1iteration)', ...
200       'optimal_V&V', 'Location', 'BestOutside');
201 title(t);

```

```

1 clear;
2 close all;
3
4 %
5 % Compares instances of the V&V, optimal V&V and Multipass V&V and
6 % various theta.
7 %
8 % Author(s): I. Papadopoulos
9 % Copyright 2005
10 % Revision: 1.1 Date: 2005/08/01
11
12 %
13 % Simulation parameters
14 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
15
16 % frequency uncertainty
17 df = 100;
18 % phase error
19 th = [0:pi/32:pi/4];
20 % signal-to-noise ratio
21 SNR = 2;
22 % number of symbols used (N preceding and N following)
23 N = 16;
24
25 %
26 % System constants
27 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
28
29 % number of symbols - 4 is QPSK (4-PSK)
30 M = 4;
31 % frequency
32 fc = 30*10^(9); % 30GHz
33 % nc
34 nc = 498;
35 % bit energy
36 Eb = 1;
37 % symbol energy
38 Es = (M/2)*Eb;
39 % symbol period
40 T = nc/fc; % about 166us
41 % Pulse width
42 A = sqrt(2*Es/T);
43 % symbol phase
44 sym = [0:(2*pi)/M:2*pi-0.01];
45
46 %
47 % Simulation
48 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
49

```

```

50 %
51 % Simulation constants
52 %
53 % nof samples
54 samples = 500;
55 % nof times
56 times = 200;
57 % Delta
58 D = (2*pi*df*T);
59 % variance
60 sigma2 = (A^2)./(10.^(SNR/10));
61 % window
62 window = N*2;
63
64 %
65 % Generate signals
66 %
67 % we must have samples at both ends - double of them for the Multipass V&V
68 ns = window + samples + window;
69 % symbols
70 m = round((M-1)*rand(times, ns));
71 % symbol phase
72 Z = m*(2*pi)/M;
73 % frequency uncertainty
74 tmp = D*[(-window):1:(samples+window-1)];
75 for i=1:times
76     DF(i, :) = tmp;
77 end
78 % complex AWGN noise
79 w = sqrt(sigma2/2)*(randn(times, ns) + j*randn(times, ns));
80 % transmitted signal
81 s = A*exp(j*Z);
82 all_samples = samples+window;
83 for thi=1:length(th)
84     % combined phase error and frequency offset
85     Terr = DF + th(thi);
86     % received signal
87     y = s.*exp(j*Terr) + w;
88     % signal power
89     p = abs(y);
90     % symbol phase
91     f = angle(y);
92
93     %
94     % Estimating noise
95     %
96     for i=1:times
97         for k=1:all_samples
98             % estimations are stored for every sample in the same line
99             tempw = [k:1:k+window];
100             ptmp = p(i, tempw);
101             ftmp = f(i, tempw);
102             % Optimal V&V estimation
103             thhat_o(i, k) = viterbi_o(y(i,tempw), Z(i,tempw));
104             % V&V F(p) = p^2
105             thhat_m2(i, k) = viterbi_s(ptmp, ftmp, 2, M);
106         end
107     end
108

```

```

109     %
110     % Demodulate
111     %
112     % useful index - everything else is garbage
113     index = [N+1:1:ns-N];
114     ss = s(:, index);
115     ys = y(:, index);
116     ms = m(:, index);
117     % phase noise removal
118     yhat_o = ys.*exp(-j*thhat_o);
119     yhat_m2 = ys.*exp(-j*thhat_m2);
120     % find symbols
121     for i=1:times
122         for k=1:all_samples
123             mw(i,k) = symbol_testing(ys(i,k), sym);
124             mhat_o(i,k) = symbol_testing(yhat_o(i,k), sym);
125             mhat_m2(i,k) = symbol_testing(yhat_m2(i,k), sym);
126         end
127     end
128
129     %
130     % Run multipass phase estimation - 1st iteration
131     %
132     Zest = mhat_m2*( (2*pi)/M ); % using V&V F(p) = p^2
133     for i=1:times
134         for k=1:samples
135             tempw = [k:1:k+window];
136             thhat_mp1(i,k) = viterbi_o(ys(i,tempw), Zest(i,tempw));
137         end
138     end
139     % find symbols
140     rindex = [N+1:1:samples+N];
141     yr = ys(:, rindex);
142     yhat_mp1 = yr.*exp(-j*thhat_mp1);
143     for i=1:times
144         for k=1:samples
145             mhat_mp1(i,k) = symbol_testing(yhat_mp1(i,k), sym);
146         end
147     end
148
149     %
150     % Calculate error
151     %
152     % residual symbol phase
153     Zs = Z(:, [window+1:1:ns-window]);
154     res_w = (symbol_phase(ys(:, rindex), M) - Zs);
155     res_o = (symbol_phase(yhat_o(:, rindex), M) - Zs);
156     res_m2 = (symbol_phase(yhat_m2(:, rindex), M) - Zs);
157     res_mp1 = (symbol_phase(yhat_mp1, M) - Zs);
158     % residual symbol phase MSE
159     resmse_w(thi) = sum(sum(res_w.^2, 1)/times)/samples;
160     resmse_o(thi) = sum(sum(res_o.^2, 1)/times)/samples;
161     resmse_m2(thi) = sum(sum(res_m2.^2, 1)/times)/samples;
162     resmse_mp1(thi) = sum(sum(res_mp1.^2, 1)/times)/samples;
163     % nof changed symbols
164     mdiff_w(thi) = ...
165         sum(100*sum( ((ms(:,rindex)-mw(:,rindex))≠0), 1)/times)/samples;
166     mdiff_o(thi) = ...
167         sum(100*sum( ((ms(:,rindex)-mhat_o(:,rindex))≠0), 1)/times)/samples;

```

```

168     mdiff_m2(thi) = ...
169         sum(100*sum( ((ms(:,rindex)-mhat_m2(:,rindex))≠0), 1)/times)/samples;
170     mdiff_mpl(thi) = ...
171         sum(100*sum( ((ms(:,rindex)-mhat_mpl)≠0), 1)/times)/samples;
172 end
173
174 %
175 % Visualize
176 %
177 t = strcat('SNR=', num2str(SNR),'dB\Delta', 'f=', num2str(df), 'Hz', ...
178     '\variable\theta', '\window_size=', num2str(window));
179 % residual phase error
180 figure;
181 hold on;
182 plot(th, resmse_m2, 'b');
183 plot(th, resmse_mpl, 'c');
184 plot(th, resmse_o, 'r:');
185 ylabel('residual_phase_(MSE)');
186 xlabel('\theta_(radians)');
187 legend('V&VLF(p)=p^2', 'Multipass_V&V(1iteration)', 'optimal_V&V', ...
188     'Location', 'BestOutside');
189 title(t);
190 % # of changed symbols
191 figure;
192 hold on;
193 plot(th, mdiff_w, 'm');
194 plot(th, mdiff_m2, 'b');
195 plot(th, mdiff_mpl, 'c');
196 plot(th, mdiff_o, 'r:');
197 ylabel('percentage_of_missed_symbols');
198 xlabel('\theta_(radians)');
199 legend('no_V&V', 'V&VLF(p)=p^2', 'Multipass_V&V(1iteration)', ...
200     'optimal_V&V', 'Location', 'BestOutside');
201 title(t);

```

```

1 clear;
2 close all;
3
4 %
5 % Compares instances optimal V&V and Multipass V&V and various SNRs.
6 %
7 % Author(s): I. Papadopoulos
8 % Copyright 2005
9 % Revision : 1.1 Date : 2005/08/01
10
11 %
12 % Simulation parameters
13 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
14
15 % frequency uncertainty
16 df = 100;
17 % phase error
18 th = pi/10;
19 % signal-to-noise ratio
20 SNR = [0:1:15];
21 % number of symbols used (N preceeding and N following)
22 N = 16;

```



```

23
24 %
25 % System constants
26 %%%%%%%%%%%
27
28 % number of symbols - 4 is QPSK (4-PSK)
29 M = 4;
30 % frequency
31 fc = 30*10^(9); % 30GHz
32 % nc
33 nc = 498;
34 % bit energy
35 Eb = 1;
36 % symbol energy
37 Es = (M/2)*Eb;
38 % symbol period
39 T = nc/fc; % about 166us
40 % Pulse width
41 A = sqrt(2*Es/T);
42 % symbol phase
43 sym = [0:(2*pi)/M:2*pi-0.01];
44
45 %
46 % Simulation
47 %%%%%%%%%%%
48
49 %
50 % Simulation constants
51 %
52 % nof samples
53 samples = 500;
54 % nof times
55 times = 200;
56 % Delta
57 D = (2*pi*df*T);
58 % variance
59 sigma2 = (A^2)./(10.^(SNR/10));
60 % window
61 window = 2*N;
62
63 %
64 % Generate signals
65 %
66 % we must have samples at both ends - in need of 2 passes
67 ns = 3*N + samples + 3*N;
68 % symbols
69 m = round((M-1)*rand(times, ns));
70 % symbol phase
71 Z = m*( (2*pi)/M );
72 % frequency uncertainty
73 tmp = D*[(-3*N):1:(samples+3*N-1)];
74 for i=1:times
75     DF(i, :) = tmp;
76 end
77 % combined phase error and frequency offset
78 Terr = DF + th;
79 % transmitted signal
80 s = A*exp( j*Z );
81 all.samples = samples + 4*N;

```

```

82 all_samples2 = samples + 2*N;
83 i1 = [2*N+1:1:samples+2*N];
84 i2 = [N+1:1:samples+N];
85 for snri=1:length(SNR)
86     % complex AWGN noise
87     w = sqrt(sigma2(snri)/2)*(randn(times, ns) + j*randn(times, ns));
88     % received signal
89     y = s.*exp( j*Terr ) + w;
90     % signal power
91     p = abs(y);
92     % symbol phase
93     f = angle(y);
94
95     %
96     % Estimating noise
97     %
98     for i=1:times
99         for k=1:all_samples
100             % estimations are stored for every sample in the same line
101             tempw = [k:1:k+window];
102             ptmp = p(i, tempw);
103             ftmp = f(i, tempw);
104
105             % Optimal V&V estimation
106             thhat_o(i, k) = viterbi_o(y(i,tempw), Z(i,tempw));
107             % V&V F(p) = 1
108             thhat_m0(i, k) = viterbi_s(ptmp, ftmp, 0, M);
109             % V&V F(p) = p^2
110             thhat_m2(i, k) = viterbi_s(ptmp, ftmp, 2, M);
111         end
112     end
113
114     %
115     % Demodulate
116     %
117     % useful index – everything else is garbage
118     index = [N+1:1:ns-N];
119     ss = s(:, index);
120     ys = y(:, index);
121     ms = m(:, index);
122     % phase noise removal
123     yhat_o = ys.*exp(-j*thhat_o);
124     yhat_m0 = ys.*exp(-j*thhat_m0);
125     yhat_m2 = ys.*exp(-j*thhat_m2);
126     % find symbols
127     for i=1:times
128         for k=1:all_samples
129             mw(i,k) = symbol_testing(ys(i,k), sym);
130             mhat_o(i,k) = symbol_testing(yhat_o(i,k), sym);
131             mhat_m0(i,k) = symbol_testing(yhat_m0(i,k), sym);
132             mhat_m2(i,k) = symbol_testing(yhat_m2(i,k), sym);
133         end
134     end
135
136     %
137     % Run multi step phase estimation – 1st iteration
138     %
139     Zest = mhat_m2*( 2*pi/M ); % using V&V F(p) = p^2
140     for i=1:times

```

```

141     for k=1:all_samples2
142         tempw = [k:1:k+window];
143         thhat_mp1(i,k) = viterbi_o(ys(i,tempw), Zest(i,tempw));
144     end
145 end
146 % find symbols
147 rindex = [N+1:1:length(index)-N];
148 yr = ys(:, rindex);
149 yhat_mp1 = yr.*exp(-j*thhat_mp1);
150 for i=1:times
151     for k=1:samples + 2*N
152         mhat_mp1(i,k) = symbol_testing(yhat_mp1(i,k), sym);
153     end
154 end
155
156 %
157 % Run multi step phase estimation - 1st iteration for V&V F() = 1
158 %
159 Zest = mhat_m0*( (2*pi)/M ); % using V&V F(p) = 1
160 for i=1:times
161     for k=1:all_samples2
162         tempw = [k:1:k+window];
163         thhat_mps1(i,k) = viterbi_o(ys(i,tempw), Zest(i,tempw));
164     end
165 end
166
167 % find symbols
168 yhat_mps1 = yr.*exp(-j*thhat_mps1);
169 for i=1:times
170     for k=1:samples + 2*N
171         mhat_mps1(i,k) = symbol_testing(yhat_mps1(i,k), sym);
172     end
173 end
174
175 %
176 % Run multi step phase estimation - 2nd iteration
177 %
178 Zest = mhat_mp1*( (2*pi)/M ); % using previous run
179 for i=1:times
180     for k=1:samples
181         tempw = [k:1:k+window];
182         thhat_mp2(i,k) = viterbi_o(yr(i,tempw), Zest(i,tempw));
183     end
184 end
185 % find symbols
186 tindex = [N+1:1:length(rindex)-N];
187 yt = yr(:, tindex);
188 yhat_mp2 = yt.*exp(-j*thhat_mp2);
189 for i=1:times
190     for k=1:samples
191         mhat_mp2(i,k) = symbol_testing(yhat_mp2(i,k), sym);
192     end
193 end
194
195 %
196 % Calculate error
197 %
198 % residual symbol phase
199 Zs = Z(:, [3*N+1:1:samples+3*N]);

```

```

200     res_w = (symbol_phase(ys(:, i1), M) - Zs);
201     res_o = (symbol_phase(yhat_o(:, i1), M) - Zs);
202     res_m2 = (symbol_phase(yhat_m2(:, i1), M) - Zs);
203     res_mp1 = (symbol_phase(yhat_mp1(:, i2), M) - Zs);
204     res_mps1 = (symbol_phase(yhat_mps1(:, i2), M) - Zs);
205     res_mp2 = (symbol_phase(yhat_mp2, M) - Zs);
206     % residual symbol phase MSE
207     resmse_w(snri) = sum(sum(res_w.^2, 1)/times)/samples;
208     resmse_o(snri) = sum(sum(res_o.^2, 1)/times)/samples;
209     resmse_m2(snri) = sum(sum(res_m2.^2, 1)/times)/samples;
210     resmse_mp1(snri) = sum(sum(res_mp1.^2, 1)/times)/samples;
211     resmse_mps1(snri) = sum(sum(res_mps1.^2, 1)/times)/samples;
212     resmse_mp2(snri) = sum(sum(res_mp2.^2, 1)/times)/samples;
213     % nof changed symbols
214     tms = ms(:, i1);
215     mdiff_w(snri) = ...
216         sum(100*sum((tms-mw(:,i1))≠0), 1)/times)/samples;
217     mdiff_o(snri) = ...
218         sum(100*sum((tms-mhat_o(:,i1))≠0), 1)/times)/samples;
219     mdiff_m2(snri) = ...
220         sum(100*sum((tms-mhat_m2(:,i1))≠0), 1)/times)/samples;
221     mdiff_mp1(snri) = ...
222         sum(100*sum((tms-mhat_mp1(:,i2))≠0), 1)/times)/samples;
223     mdiff_mps1(snri) = ...
224         sum(100*sum((tms-mhat_mps1(:,i2))≠0), 1)/times)/samples;
225     mdiff_mp2(snri) = ...
226         sum(100*sum((tms-mhat_mp2)≠0), 1)/times)/samples;
227 end
228
229 %
230 % Visualize
231 %
232 t = strcat('variable_SNR_\Delta', 'f=', num2str(df), 'Hz_\theta=', ...,
233     num2str(th), '_window_size=', num2str(window));
234 % residual phase error
235 figure;
236 hold on;
237 plot(SNR, resmse_m2, 'b');
238 plot(SNR, resmse_mp1, 'c');
239 plot(SNR, resmse_mp2, 'y');
240 plot(SNR, resmse_o, 'r');
241 ylabel('residual_phase_(MSE)');
242 xlabel('SNR_(dB)');
243 legend('V&V_F(p)=p^2', 'Multipass_V&V_(1iteration)', ...
244     'Multipass_V&V_(2iterations)', 'optimal_V&V', ...
245     'Location', 'BestOutside');
246 title(t);
247 % residual phase error
248 figure;
249 hold on;
250 plot(SNR, resmse_mp1, 'c');
251 plot(SNR, resmse_mps1, 'k');
252 ylabel('residual_phase_(MSE)');
253 xlabel('SNR_(dB)');
254 legend('Multipass_V&V_F(p)=p^2', 'Multipass_V&V_F(p)=1', ...
255     'Location', 'BestOutside');
256 title(t);
257 % # of changed symbols
258 figure;

```

```

259 hold on;
260 plot(SNR, mdiff_w, 'm');
261 plot(SNR, mdiff_m2, 'b');
262 plot(SNR, mdiff_mp1, 'c');
263 plot(SNR, mdiff_mp2, 'y');
264 plot(SNR, mdiff_o, 'r:');
265 ylabel('percentage_of_missed_symbols');
266 xlabel('SNR_(dB)');
267 legend('no_V&V', 'V&V_F(p)=p^2', 'Multipass_V&V(1iteration)', ...
268       'Multipass_V&V(2iterations)', 'optimal_V&V', ...
269       'Location', 'BestOutside');
270 title(t);
271 % # of changed symbols
272 figure;
273 hold on;
274 plot(SNR, mdiff_mp1, 'c');
275 plot(SNR, mdiff_mps1, 'k');
276 ylabel('percentage_of_missed_symbols');
277 xlabel('SNR_(dB)');
278 legend('Multipass_V&V_F(p)=p^2', 'Multipass_V&V_F(p)=1', ...
279       'Location', 'BestOutside');
280 title(t);

```

Β.5 Κώδικας προσομοιώσεων εκτιμητών βαρυκέντρου

```

1 clear;
2 close all;
3
4 %
5 % Compares instances of the BAR algorithm with V&V and various SNRs.
6 %
7 % Author(s): I. Papadopoulos
8 % Copyright 2005
9 % Revision: 1.1 Date: 2005/09/27
10
11 %
12 % Simulation parameters
13 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
14
15 % frequency uncertainty
16 df = 0;
17 % phase error
18 th = pi/10;
19 % signal-to-noise ratio
20 SNR = [0:1:15];
21 % number of symbols used (N preceding and N following)
22 N = 16;
23 % EBAR threshold
24 thresh = 3*pi/8;
25
26 %
27 % System constants
28 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
29
30 % number of symbols -- 4 is QPSK (4-PSK)

```

```

31 M = 4;
32 % frequency
33 fc = 30*10^(9); % 30GHz
34 % nc
35 nc = 498;
36 % bit energy
37 Eb = 1;
38 % symbol energy
39 Es = (M/2)*Eb;
40 % symbol period
41 T = nc/fc; % about 166us
42 % Pulse width
43 A = sqrt(2*Es/T);
44 % symbol phase
45 sym = [0:(2*pi)/M:2*pi-0.01];
46
47 %
48 % Simulation
49 %%%%%%%%%%%
50
51 %
52 % Simulation constants
53 %
54
55 % nof samples
56 samples = 200;
57 % nof times
58 times = 200;
59 % Delta
60 D = (2*pi*df*T);
61 % variance
62 sigma2 = (A^2)./(10.^(SNR/10));
63 % window
64 window = N*2;
65
66 %
67 % Generate signals
68 %
69 % we must have samples at both ends - V&V estimates the central symbol
70 % (window is 2N+1) whereas BAR estimates the start of the window
71 ns = N + samples + 2*N - 1;
72 % symbols
73 m = round((M-1)*rand(times, ns));
74 % symbol phase
75 Z = m*( (2*pi)/M );
76 % frequency uncertainty
77 tmp = D*[-N:1:samples+2*N-2];
78 for i=1:times
79     DF(i, :) = tmp;
80 end
81 % combined phase error and frequency offset
82 Terr = DF + th;
83 % transmitted signal
84 s = A*exp( j*Z );
85 for snri=1:length(SNR)
86     % complex AWGN noise
87     w = sqrt(sigma2(snri)/2)*(randn(times, ns) + j*randn(times, ns));
88     % received signal
89     y = s.*exp( j*Terr ) + w;

```

```

90     % signal power
91     p = abs(y);
92     % symbol phase
93     f = angle(y);
94     % removed modulation using FOLD function
95     fd = FOLD(f, M);
96
97     %
98     % Estimating noise
99     %
100    for i=1:times
101        for k=1:samples
102            % estimations are stored for every sample in the same line
103            tempw = [k:1:k>window];
104            ptmp = p(i, tempw);
105            ftmp = f(i, tempw);
106            % V&V F(p) = p^2
107            thhat_m2(i, k) = viterbi_s(ptmp, ftmp, 2, M);
108            % BAR
109            tempw = [k+N:1:k+N>window-1]; % get 2*N samples
110            ftmp = fd(i, tempw); % using demodulated signal phase
111            thhat_bar(i,k) = MEMARB(ftmp, M);
112            thhat_ebar(i,k) = MEMARB_EBAR(ftmp, thresh, M);
113            thhat_lbar(i,k) = MEMARB_LBAR(ftmp, M);
114            thhat_qlbar(i,k) = MEMARB_QLBAR(ftmp);
115        end
116    end
117
118    %
119    % Demodulate
120    %
121    % useful index - everything else is garbage
122    index = [N+1:1:samples+N];
123    ss = s(:, index);
124    ys = y(:, index);
125    ms = m(:, index);
126    % phase noise removal
127    yhat_m2 = ys.*exp(-j*thhat_m2);
128    yhat_bar = ys.*exp(-j*thhat_bar);
129    yhat_ebar = ys.*exp(-j*thhat_ebar);
130    yhat_lbar = ys.*exp(-j*thhat_lbar);
131    yhat_qlbar = ys.*exp(-j*thhat_qlbar);
132    % find symbols
133    for i=1:times
134        for k=1:samples
135            mw(i,k) = symbol_testing(ys(i,k), sym);
136            mhat_m2(i,k) = symbol_testing(yhat_m2(i,k), sym);
137            mhat_bar(i,k) = symbol_testing(yhat_bar(i,k), sym);
138            mhat_ebar(i,k) = symbol_testing(yhat_ebar(i,k), sym);
139            mhat_lbar(i,k) = symbol_testing(yhat_lbar(i,k), sym);
140            mhat_qlbar(i,k) = symbol_testing(yhat_qlbar(i,k), sym);
141        end
142    end
143
144    %
145    % Calculate error
146    %
147
148    % residual symbol phase

```

```

149     Zs = Z(:, index);
150     res_w      = (symbol_phase(ys, M) - Zs);
151     res_m2     = (symbol_phase(yhat_m2, M) - Zs);
152     res_bar    = (symbol_phase(yhat_bar, M) - Zs);
153     res_ebar   = (symbol_phase(yhat_ebar, M) - Zs);
154     res_lbar   = (symbol_phase(yhat_lbar, M) - Zs);
155     res_qlbar  = (symbol_phase(yhat_qlbar, M) - Zs);
156     % residual symbol phase MSE
157     resmse_w(snri) = sum(sum(res_w.^2, 1)/times)/samples;
158     resmse_m2(snri) = sum(sum(res_m2.^2, 1)/times)/samples;
159     resmse_bar(snri) = sum(sum(res_bar.^2, 1)/times)/samples;
160     resmse_ebar(snri) = sum(sum(res_ebar.^2, 1)/times)/samples;
161     resmse_lbar(snri) = sum(sum(res_lbar.^2, 1)/times)/samples;
162     resmse_qlbar(snri) = sum(sum(res_qlbar.^2, 1)/times)/samples;
163     % nof changed symbols
164     mdiff_w(snri) = sum(100*sum(((ms-mw)≠0), 1)/times)/samples;
165     mdiff_m2(snri) = sum(100*sum(((ms-mhat_m2)≠0), 1)/times)/samples;
166     mdiff_bar(snri) = sum(100*sum(((ms-mhat_bar)≠0), 1)/times)/samples;
167     mdiff_ebar(snri) = sum(100*sum(((ms-mhat_ebar)≠0), 1)/times)/samples;
168     mdiff_lbar(snri) = sum(100*sum(((ms-mhat_lbar)≠0), 1)/times)/samples;
169     mdiff_qlbar(snri) = sum(100*sum(((ms-mhat_qlbar)≠0), 1)/times)/samples;
170 end
171
172 %
173 % Visualize
174 %
175 t = strcat('variable_SNR_L\Delta', 'f=', num2str(df), 'Hz_L\theta=', ...,
176           num2str(th), '_Lwindow_size=', num2str(window));
177 % comparison of residual phase error
178 figure;
179 hold on;
180 plot(SNR, resmse_m2, 'b');
181 plot(SNR, resmse_bar, 'm');
182 plot(SNR, resmse_ebar, 'g');
183 plot(SNR, resmse_lbar, 'r');
184 ylabel('residual_phase_(MSE)');
185 xlabel('SNR_(dB)');
186 legend('V&V_LF(p)=p^2', 'BAR', 'EBAR', 'LBAR', 'Location', 'BestOutside');
187 title(t);
188 % # of changed symbols
189 figure;
190 hold on;
191 plot(SNR, mdiff_w, 'k:');
192 plot(SNR, mdiff_m2, 'b');
193 plot(SNR, mdiff_bar, 'm');
194 plot(SNR, mdiff_ebar, 'g');
195 plot(SNR, mdiff_lbar, 'r');
196 ylabel('percentage_of_missed_symbols');
197 xlabel('SNR_(dB)');
198 legend('no_V&V', 'V&V_LF(p)=p^2', 'BAR', 'EBAR', 'LBAR', 'Location', ...
199       'BestOutside');
200 title(t);
201 % # of changed symbols
202 figure;
203 hold on;
204 plot(SNR, mdiff_lbar, 'r');
205 plot(SNR, mdiff_qlbar, 'c');
206 ylabel('percentage_of_missed_symbols');
207 xlabel('SNR_(dB)');

```



```

208 legend('LBAR', 'QLBAR', 'Location', 'BestOutside');
209 title(t);

```

```

1 clear;
2 close all;
3
4 %
5 % Compares instances of the BAR algorithm with V&V and various theta.
6 %
7 % Author(s): I. Papadopoulos
8 % Copyright 2005
9 % Revision : 1.1 Date : 2005/08/01
10
11 %
12 % Simulation parameters
13 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
14
15 % frequency uncertainty
16 df = 100;
17 % phase error
18 th = [0:pi/32:pi/4];
19 % signal-to-noise ratio
20 SNR = 0;
21 % number of symbols used (N preceding and N following)
22 N = 16;
23 % EBAR threshold
24 thresh = 3*pi/8;
25
26 %
27 % System constants
28 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
29
30 % number of symbols - 4 is QPSK (4-PSK)
31 M = 4;
32 % frequency
33 fc = 30*10^(9); % 30GHz
34 % nc
35 nc = 498;
36 % bit energy
37 Eb = 1;
38 % symbol energy
39 Es = (M/2)*Eb;
40 % symbol period
41 T = nc/fc; % about 166us
42 % Pulse width
43 A = sqrt(2*Es/T);
44 % symbol phase
45 sym = [0:(2*pi)/M:2*pi-0.01];
46
47 %
48 % Simulation
49 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
50
51 %
52 % Simulation constants
53 %
54 % nof samples

```

```

55 samples = 200;
56 % nof times
57 times = 200;
58 % Delta
59 D = (2*pi*df*T);
60 % variance
61 sigma2 = (A^2)./(10.^(SNR/10));
62 % window
63 window = N*2;
64
65 %
66 % Generate signals
67 %
68 % we must have samples at both ends - V&V estimates the central symbol
69 % (window is 2N+1) whereas BAR estimates the start of the window
70 ns = N + samples + 2*N - 1;
71 % symbols
72 m = round((M-1)*rand(times, ns));
73 % symbol phase
74 Z = m*(2*pi)/M;
75 % frequency uncertainty
76 tmp = D*[-N:1:samples+2*N-2];
77 for i=1:times
78     DF(i, :) = tmp;
79 end
80 % complex AWGN noise
81 w = sqrt(sigma2/2)*(randn(times, ns) + j*randn(times, ns));
82 % transmitted signal
83 s = A*exp(j*Z);
84 for thi=1:length(th)
85     % combined phase error and frequency offset
86     Terr = DF + th(thi);
87     % received signal
88     y = s.*exp(j*Terr) + w;
89     % signal power
90     p = abs(y);
91     % symbol phase
92     f = angle(y);
93     % removed modulation using FOLD function
94     fd = FOLD(f, M);
95
96     %
97     % Estimating noise
98     %
99     for i=1:times
100         for k=1:samples
101             % estimations are stored for every sample in the same line
102             tempw = [k:1:k+window];
103             ptmp = p(i, tempw);
104             ftmp = f(i, tempw);
105             % V&V  $F(p) = p^2$ 
106             thhat_m2(i, k) = viterbis(ptmp, ftmp, 2, M);
107             % BAR
108             tempw = [k+N:1:k+N+window-1]; % get 2*N samples
109             ftmp = fd(i, tempw); % using demodulated signal phase
110             thhat_bar(i,k) = MEMARB(ftmp, M);
111             thhat_lbar(i,k) = MEMARB_EBAR(ftmp, thresh, M);
112             thhat_lbar(i,k) = MEMARB_LBAR(ftmp, M);
113             thhat_qlbar(i,k) = MEMARB_QLBAR(ftmp);

```

```

114     end
115 end
116
117 %
118 % Demodulate
119 %
120 % useful index - everything else is garbage
121 index = [N+1:1:samples+N];
122 ss = s(:, index);
123 ys = y(:, index);
124 ms = m(:, index);
125 % phase noise removal
126 yhat_m2 = ys.*exp(-j*thhat_m2);
127 yhat_bar = ys.*exp(-j*thhat_bar);
128 yhat_ebar = ys.*exp(-j*thhat_ebar);
129 yhat_lbar = ys.*exp(-j*thhat_lbar);
130 yhat_qlbar = ys.*exp(-j*thhat_qlbar);
131 % find symbols
132 for i=1:times
133     for k=1:samples
134         mw(i,k) = symbol_testing(ys(i,k), sym);
135         mhat_m2(i,k) = symbol_testing(yhat_m2(i,k), sym);
136         mhat_bar(i,k) = symbol_testing(yhat_bar(i,k), sym);
137         mhat_ebar(i,k) = symbol_testing(yhat_ebar(i,k), sym);
138         mhat_lbar(i,k) = symbol_testing(yhat_lbar(i,k), sym);
139         mhat_qlbar(i,k) = symbol_testing(yhat_qlbar(i,k), sym);
140     end
141 end
142
143 %
144 % Calculate error
145 %
146 % residual symbol phase
147 Zs = Z(:, index);
148 res_w = (symbol_phase(ys, M) - Zs);
149 res_m2 = (symbol_phase(yhat_m2, M) - Zs);
150 res_bar = (symbol_phase(yhat_bar, M) - Zs);
151 res_ebar = (symbol_phase(yhat_ebar, M) - Zs);
152 res_lbar = (symbol_phase(yhat_lbar, M) - Zs);
153 res_qlbar = (symbol_phase(yhat_qlbar, M) - Zs);
154 % residual symbol phase MSE
155 resmse_w(thi) = sum(sum(res_w.^2, 1)/times)/samples;
156 resmse_m2(thi) = sum(sum(res_m2.^2, 1)/times)/samples;
157 resmse_bar(thi) = sum(sum(res_bar.^2, 1)/times)/samples;
158 resmse_ebar(thi) = sum(sum(res_ebar.^2, 1)/times)/samples;
159 resmse_lbar(thi) = sum(sum(res_lbar.^2, 1)/times)/samples;
160 resmse_qlbar(thi) = sum(sum(res_qlbar.^2, 1)/times)/samples;
161 % nof changed symbols
162 mdiff_w(thi) = sum(100*sum((ms-mw)~=0, 1)/times)/samples;
163 mdiff_m2(thi) = sum(100*sum((ms-mhat_m2)~=0, 1)/times)/samples;
164 mdiff_bar(thi) = sum(100*sum((ms-mhat_bar)~=0, 1)/times)/samples;
165 mdiff_ebar(thi) = sum(100*sum((ms-mhat_ebar)~=0, 1)/times)/samples;
166 mdiff_lbar(thi) = sum(100*sum((ms-mhat_lbar)~=0, 1)/times)/samples;
167 mdiff_qlbar(thi) = sum(100*sum((ms-mhat_qlbar)~=0, 1)/times)/samples;
168 end
169
170 %
171 % Visualize
172 %

```

```

173 t = strcat('SNR=', num2str(SNR), 'dB\Delta', 'f=', num2str(df), 'Hz', ...
174     '\variable\theta', '\window_size=', num2str(window));
175 % comparison of residual phase error
176 figure;
177 hold on;
178 plot(th, resmse_m2, 'b');
179 plot(th, resmse_bar, 'm');
180 plot(th, resmse_ebar, 'g');
181 plot(th, resmse_lbar, 'r');
182 xlabel('\theta (radians)');
183 ylabel('residual phase (MSE)');
184 legend('V&V\F(p)=p^2', 'BAR', 'EBAR', 'LBAR', 'Location', 'BestOutside');
185 title(t);
186 % # of changed symbols
187 figure;
188 hold on;
189 plot(th, mdiff_w, 'k:');
190 plot(th, mdiff_m2, 'b');
191 plot(th, mdiff_bar, 'm');
192 plot(th, mdiff_ebar, 'g');
193 plot(th, mdiff_lbar, 'r');
194 ylabel('percentage of missed symbols');
195 xlabel('\theta (radians)');
196 legend('no_V&V', 'V&V\F(p)=p^2', 'BAR', 'EBAR', 'LBAR', 'Location', ...
197     'BestOutside');
198 title(t);
199 % # of changed symbols
200 figure;
201 hold on;
202 plot(th, mdiff_lbar, 'r');
203 plot(th, mdiff_qlbar, 'c');
204 ylabel('percentage of missed symbols');
205 xlabel('\theta (radians)');
206 legend('LBAR', 'QLBAR', 'Location', 'BestOutside');
207 title(t);

```

```

1 clear;
2 close all;
3
4 %
5 % Compares instances of the BAR algorithm with V&V and various windows.
6 %
7 % Author(s): I. Papadopoulos
8 % Copyright 2005
9 % Revision: 1.1 Date: 2005/08/01
10
11 %
12 % Simulation parameters
13 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
14
15 % frequency uncertainty
16 df = 100;
17 % phase error
18 th = pi/10;
19 % signal-to-noise ratio
20 SNR = 2;
21 % number of symbols used (N preceding and N following)

```

```

22 N = [2,4,8,16,32,64,128];
23
24 %
25 % System constants
26 %%%%%%%%%%%
27
28 % number of symbols - 4 is QPSK (4-PSK)
29 M = 4;
30 % frequency
31 fc = 30*10^(9); % 30GHz
32 % nc
33 nc = 498;
34 % bit energy
35 Eb = 1;
36 % symbol energy
37 Es = (M/2)*Eb;
38 % symbol period
39 T = nc/fc; % about 166us
40 % Pulse width
41 A = sqrt(2*Es/T);
42 % symbol phase
43 sym = [0:(2*pi)/M:2*pi-0.01];
44 % EBAR threshold
45 thresh = 3*pi/8;
46
47 %
48 % Simulation
49 %%%%%%%%%%%
50
51 %
52 % Simulation constants
53 %
54
55 % nof samples
56 samples = 300;
57 % nof times
58 times = 200;
59 % Delta
60 D = (2*pi*df*T);
61 % variance
62 sigma2 = (A^2)./(10.^(SNR/10));
63 % window
64 window = N*2;
65
66 %
67 % Generate signals
68 %
69 % maximum samples
70 ns = N(length(N)) + samples + 2*N(length(N))-1;
71 % symbols
72 m = round((M-1)*rand(times, ns));
73 % symbol phase
74 Z = m*(2*pi)/M;
75 % complex AWGN noise
76 w = sqrt(sigma2/2)*(randn(times, ns) + j*randn(times, ns));
77 % frequency uncertainty
78 tmp = D*[(-N(length(N))):1:samples+2*N(length(N))-2];
79 for i=1:times
80     DF(i, :) = tmp;

```

```

81 end
82 % combined phase error and frequency offset
83 Terr = DF + th;
84 % transmitted signal
85 s = A*exp( j*Z );
86 % received signal
87 y = s.*exp( j*Terr ) + w;
88 % signal power
89 p = abs(y);
90 % symbol phase
91 f = angle(y);
92 % removed modulation using FOLD function
93 fd = FOLD(f, M);
94 time = toc;
95 fprintf('Generating_%i_signals:_%4.2f_secs\n', ns, time);
96 % useful index - everything else is garbage
97 index = [N(length(N))+1:1:samples+N(length(N))];
98 for wini=1:length(window)
99     % useful signals
100     us = [N(length(N))+1-N(wini):1:samples+N(length(N))+N(wini)];
101     ft = f(:, us);
102     pt = p(:, us);
103
104     %
105     % Estimating noise
106     %
107     for i=1:times
108         for k=1:samples
109             % estimations are stored for every sample in the same line
110             tempw = [k:1:k+window(wini)];
111             ptmp = pt(i, tempw);
112             ftmp = ft(i, tempw);
113             % V&V F(p) = p^2
114             thhat_m2(i, k) = viterbi_s(ptmp, ftmp, 2, M);
115             % BAR
116             tempw = [k+N(wini):1:k+N(wini)+window(wini)-1]; % get 2*N samples
117             ftmp = fd(i, tempw); % using demodulated signal phase
118             thhat_bar(i,k) = MEMARB(ftmp, M);
119             thhat_ebar(i,k) = MEMARB_EBAR(ftmp, thresh, M);
120             thhat_lbar(i,k) = MEMARB_LBAR(ftmp, M);
121             thhat_qlbar(i,k) = MEMARB_QLBAR(ftmp);
122         end
123     end
124
125     %
126     % Demodulate
127     %
128     ss = s(:, index);
129     ys = y(:, index);
130     ms = m(:, index);
131     % phase noise removal
132     yhat_m2 = ys.*exp(-j*thhat_m2);
133     yhat_bar = ys.*exp(-j*thhat_bar);
134     yhat_ebar = ys.*exp(-j*thhat_ebar);
135     yhat_lbar = ys.*exp(-j*thhat_lbar);
136     yhat_qlbar = ys.*exp(-j*thhat_qlbar);
137     % find symbols
138     for i=1:times
139         for k=1:samples

```

```

140         mw(i,k)           = symbol_testing(ys(i,k), sym);
141         mhat_m2(i,k)      = symbol_testing(yhat_m2(i,k), sym);
142         mhat_bar(i,k)     = symbol_testing(yhat_bar(i,k), sym);
143         mhat_ebar(i,k)    = symbol_testing(yhat_ebar(i,k), sym);
144         mhat_lbar(i,k)    = symbol_testing(yhat_lbar(i,k), sym);
145         mhat_qlbar(i,k)   = symbol_testing(yhat_qlbar(i,k), sym);
146     end
147 end
148
149 %
150 % Calculate error
151 %
152 % residual symbol phase
153 Zs = Z(:, index);
154 res_w      = (symbol_phase(ys, M) - Zs);
155 res_m2     = (symbol_phase(yhat_m2, M) - Zs);
156 res_bar    = (symbol_phase(yhat_bar, M) - Zs);
157 res_ebar   = (symbol_phase(yhat_ebar, M) - Zs);
158 res_lbar   = (symbol_phase(yhat_lbar, M) - Zs);
159 res_qlbar  = (symbol_phase(yhat_qlbar, M) - Zs);
160 % residual symbol phase MSE
161 resmse_w(wini) = sum(sum(res_w.^2, 1)/times)/samples;
162 resmse_m2(wini) = sum(sum(res_m2.^2, 1)/times)/samples;
163 resmse_bar(wini) = sum(sum(res_bar.^2, 1)/times)/samples;
164 resmse_ebar(wini) = sum(sum(res_ebar.^2, 1)/times)/samples;
165 resmse_lbar(wini) = sum(sum(res_lbar.^2, 1)/times)/samples;
166 resmse_qlbar(wini) = sum(sum(res_qlbar.^2, 1)/times)/samples;
167 % nof changed symbols
168 mdiff_w(wini) = sum(100*sum((ms-mw)~=0, 1)/times)/samples;
169 mdiff_m2(wini) = sum(100*sum((ms-mhat_m2)~=0, 1)/times)/samples;
170 mdiff_bar(wini) = sum(100*sum((ms-mhat_bar)~=0, 1)/times)/samples;
171 mdiff_ebar(wini) = sum(100*sum((ms-mhat_ebar)~=0, 1)/times)/samples;
172 mdiff_lbar(wini) = sum(100*sum((ms-mhat_lbar)~=0, 1)/times)/samples;
173 mdiff_qlbar(wini) = sum(100*sum((ms-mhat_qlbar)~=0, 1)/times)/samples;
174 end
175
176 %
177 % Visualize
178 %
179 t = strcat('SNR=', num2str(SNR), 'dB\Delta', 'f=', num2str(df), '...',
180           'Hz\theta=', num2str(th), '\window_size_variable');
181 % comparison of residual phase error
182 figure;
183 hold on;
184 plot(window, resmse_m2, 'b');
185 plot(window, resmse_bar, 'm');
186 plot(window, resmse_ebar, 'g');
187 plot(window, resmse_lbar, 'r');
188 xlabel('window_size');
189 ylabel('residual_phase_(MSE)');
190 legend('V&V_F(p)=p^2', 'BAR', 'EBAR', 'LBAR', 'Location', 'BestOutside');
191 title(t);
192 % # of changed symbols
193 figure;
194 hold on;
195 plot(window, mdiff_w, 'k:');
196 plot(window, mdiff_m2, 'b');
197 plot(window, mdiff_bar, 'm');
198 plot(window, mdiff_ebar, 'g');

```

```
199 plot(window, mdiff_lbar, 'r');
200 ylabel('percentage_of_missed_symbols');
201 xlabel('window_size');
202 legend('no_V&V', 'V&V_LF(p)=p^2', 'BAR', 'EBAR', 'LBAR','Location', ...
203       'BestOutside');
204 title(t);
205 % # of changed symbols
206 figure;
207 hold on;
208 plot(window, mdiff_lbar, 'r');
209 plot(window, mdiff_qlbar, 'c');
210 ylabel('percentage_of_missed_symbols');
211 xlabel('window_size');
212 legend('LBAR', 'QLBAR', 'Location', 'BestOutside');
213 title(t);
```


Βιβλιογραφία

- [1] Catherine Morlet, "*Démodulateur embarqué multiporteuses pour services multimédia par satellites*", Ph.D. thesis, Ecole Nationale Supérieure des Télécommunications, 2000.
- [2] A.J. Viterbi, A.M. Viterbi, "*Nonlinear Estimation of PSK-modulated Carrier Phase with Applications to Burst Digital Transmission*", IEEE Transactions on Information Theory vol. IT-29, pp543-551, July 1983.
- [3] Marie-Laure Boucheret, Ivar Mortensens, Henri Favaro, Eric Belis, "*A new Algorithm for Nonlinear Estimation of PSK-Modulated Carrier Phase*", ENST TELECOM Paris, Alcatel Space Industries.
- [4] Catherine Morlet, Jungpil Yu, Marie-Laure Boucheret, Gilles Mesnager, "*Synchronization Algorithms for Multimedia Satellite Communications Payloads*", Alcatel Space Industries.
- [5] Catherine Morlet, Marie-Laure Boucheret, "*Impact of Phase Noise on Carrier Phase Estimation at Ka Band*", Alcatel Space Industries, ENST.
- [6] Yan Wang, Erchin Serpedin, Philippe Ciblat, "*Optimal Blind Carrier Recovery for MPSK Burst Transmissions*", IEEE Transactions on Communications vol. 51, No. 9, September 2003.
- [7] D. Giancristofaro, R. Ardoino, "*Synchronization Architectures, Algorithms and Demodulation Performances for the EuroSkyWay Payload based on Symbol-Synchronous MF-TDMA*", Alenia Aerospazio.
- [8] Γεώργιος Μουστακίδης, '*Εισαγωγή στην Θεωρία Εκτίμησης και Ανίχνευσης*', Πανεπιστήμιο Πατρών.
- [9] Simon Haykin, '*Συστήματα Επικοινωνίας*', εκδόσεις Παπασωτηρίου, ISBN 960-7510-18-6, Αθήνα, 1995.
- [10] Wikipedia, <http://wikipedia.org>.



ΠΑΝΕΠΙΣΤΗΜΙΟ
ΘΕΣΣΑΛΙΑΣ



004000074802