

ΠΑΝΕΠΙΣΤΗΜΙΟ ΚΡΗΤΗΣ
ΣΧΟΛΗ ΘΕΤΙΚΩΝ ΕΠΙΣΤΗΜΩΝ
ΤΜΗΜΑ ΕΠΙΣΤΗΜΗΣ ΥΠΟΛΟΓΙΣΤΩΝ

**ΣΧΕΔΙΑΣΜΟΣ ΚΑΙ ΥΛΟΠΟΙΗΣΗ
ΜΗΧΑΝΙΣΜΩΝ ΒΕΛΤΙΩΣΗΣ ΤΗΣ ΠΟΙΟΤΗΤΑΣ
ΥΠΗΡΕΣΙΑΣ ΓΙΑ ΤΗΝ ΜΕΤΑΔΟΣΗ ΦΩΝΗΣ
ΜΕΣΩ ΤΟΥ ΔΙΑΔΙΚΤΥΟΥ**

ΠΑΓΩΜΕΝΟΣ Α. ΓΕΩΡΓΙΟΣ

Μεταπτυχιακή Εργασία

ΗΡΑΚΛΕΙΟ ΟΚΤΩΒΡΙΟΣ 2001

Πανεπιστήμιο Κρήτης
Σχολή Θετικών Επιστημών
Τμήμα Επιστήμης Υπολογιστών

**Σχεδιασμός και Υλοποίηση μηχανισμών βελτίωσης της
ποιότητας υπηρεσίας για την μετάδοση φωνής μέσω του
διαδικτύου**

Παγωμένος Α. Γεώργιος

Μεταπτυχιακή Εργασία

Ηράκλειο, Οκτώβριος 2001

Πανεπιστήμιο Κρήτης
Σχολή Θετικών Επιστημών
Τμήμα Επιστήμης Υπολογιστών

**Σχεδιασμός και Υλοποίηση μηχανισμών βελτίωσης της ποιότητας
υπηρεσίας για την μετάδοση φωνής μέσω του διαδικτύου**

Εργασία που υποβλήθηκε από τον
Γεώργιο Α. Παγωμένο
ως μερική εκπλήρωση των απαιτήσεων για την απόκτηση
ΜΕΤΑΠΤΥΧΙΑΚΟΥ ΔΙΠΛΩΜΑΤΟΣ ΕΙΔΙΚΕΥΣΗΣ

Συγγραφέας:

Γεώργιος Α. Παγωμένος
Τμήμα Επιστήμης Υπολογιστών
Πανεπιστήμιο Κρήτης

Εισηγητική Επιτροπή:

Απόστολος Τραγανίτης
Αναπληρωτής Καθηγητής, Επόπτης

Ευάγγελος Μαρκάτος
Αναπληρωτής Καθηγητής, Μέλος

Γεώργιος Γεωργακόπουλος
Επίκουρος Καθηγητής, Μέλος

Δεκτή:

Πάνος Κωνσταντόπουλος, Καθηγητής
Πρόεδρος Επιτροπής Μεταπτυχιακών Σπουδών

Ηράκλειο, Οκτώβριος 2001

Στην οικογένεια μου

Σχεδιασμός και Υλοποίηση μηχανισμών βελτίωσης της ποιότητας υπηρεσίας για την μετάδοση φωνής μέσω του διαδικτύου

Γεώργιος Α. Παγωμένος

Μεταπτυχιακή Εργασία

Τμήμα Επιστήμης Υπολογιστών
Πανεπιστήμιο Κρήτης

Περίληψη

Επιχειρήσεις και οργανισμοί σε όλο τον κόσμο αναζητούν ένα τρόπο μείωσης των διαρκώς αυξανόμενων τηλεπικοινωνιακών τους δαπανών. Η χρήση κοινών τηλεπικοινωνιακών πόρων για την μετάδοση της φωνής και των δεδομένων, τους παρέχει ένα τρόπο εξοικονόμησης εσόδων.

Αν και τα οφέλη, που προκύπτουν από την χρήση εφαρμογών μετάδοσης φωνής μέσω του διαδικτύου είναι προφανή και παράλληλα υπάρχουν αρκετά εμπορικά προϊόντα στην αγορά μόνο ένα μικρό ποσοστό χρηστών του διαδικτύου χρησιμοποιούν αυτή την τεχνολογία. Τον βασικότερο λόγο, αποτελεί η ποιότητα της παρεχόμενης υπηρεσίας η οποία δεν μπορεί να συγκριθεί με την αντίστοιχη του τηλεφωνικού δικτύου.

Στα πλαίσια αυτής της εργασίας, μελετήθηκαν αλλά και προτάθηκαν νέες τεχνικές βελτίωσης της παρεχόμενης ποιότητας υπηρεσίας των εφαρμογών μετάδοσης φωνής μέσω του διαδικτύου. Ουσιαστικά η επιδιωκόμενη βελτίωση επιτυγχάνεται αντιμετωπίζοντας δυο βασικά χαρακτηριστικά του διαδικτύου: την διακύμανση της καθυστέρησης και τις απώλειες πακέτων.

Επόπτης: Απόστολος Τραγανίτης
Αναπληρωτής Καθηγητής Επιστήμης Υπολογιστών
Πανεπιστήμιο Κρήτης

Designing and implementing mechanisms for improving the Quality of Service for Voice over IP applications

George A. Pagomenos
Master of Science Thesis

Computer Science Department
University of Crete

Abstract

Organizations and enterprises around the world seek to reduce rising communications costs. The consolidation of separate voice and data resources offers an opportunity for significant savings.

Although, the benefits from using Voice over IP technology are obvious and there are a lot of applications in the market only a small fraction of users utilized it. The basic reason consists the low quality of voice provided in comparison to the Public Switched Telephone Network.

Scope of this thesis, was to study and propose new techniques in order to improve the quality of service of VoIP technology. More specifically, we tried to develop mechanisms that attempt to eliminate or at least minimize the impact of packet loss and delay jitter on the quality of the audio delivered to the destinations.

Supervisor: Apostolos Traganitis
Associate Professor of Computer Science
University of Crete

Ευχαριστίες

Στο σημείο αυτό θα ήθελα να ευχαριστήσω θερμά τον επόπτη της εργασίας μου κ. Απόστολο Τραγανίτη για την πολύτιμη βοήθεια αλλά και καθοδήγηση που μου παρείχε σε όλη την διάρκεια της παρούσας εργασίας.

Οφείλω επίσης ένα θερμό ευχαριστώ στα μέλη της εισηγητικής επιτροπής κ. Ευάγγελο Μαρκάτο και κ. Γεώργιο Γεωργακόπουλο για τις εποικοδομητικές τους παρατηρήσεις.

Το μεγαλύτερο όμως ευχαριστώ το οφείλω στους γονείς μου, Ανδρέα και Αθηνά για τους κόπους, τις θυσίες και την εμπιστοσύνη που μου έδειξαν καθώς και στα αδέρφια μου Δήμο και Γιάννη, για την ψυχολογική υποστήριξη που μου παρείχαν χωρίς την οποία δεν θα ήμουν σε θέση να ολοκληρώσω αυτήν την εργασία.

Κατάλογος Περιεχομένων

ΚΑΤΑΛΟΓΟΣ ΠΕΡΙΕΧΟΜΕΝΩΝ	15
ΚΑΤΑΛΟΓΟΣ ΕΙΚΟΝΩΝ	17
ΚΑΤΑΛΟΓΟΣ ΠΙΝΑΚΩΝ	19
ΚΑΤΑΛΟΓΟΣ ΠΛΑΙΣΙΩΝ	21
ΚΕΦΑΛΑΙΟ 1	23
ΕΙΣΑΓΩΓΗ	23
1.1 ΟΡΙΣΜΟΣ	23
1.2 ΠΑΡΟΥΣΙΑΣΗ ΤΟΥ ΠΡΟΒΛΗΜΑΤΟΣ	23
1.3 ΠΑΡΟΥΣΙΑΣΗ ΠΑΡΟΝΤΩΝ ΠΡΩΤΟΚΟΛΛΩΝ ΕΠΙΚΟΙΝΩΝΙΑΣ	25
1.4 ΔΟΜΗ ΤΗΣ ΕΡΓΑΣΙΑΣ.....	27
ΚΕΦΑΛΑΙΟ 2	29
ΜΗΧΑΝΙΣΜΟΙ ΕΞΑΛΕΙΨΗΣ ΔΙΑΚΥΜΑΝΣΕΩΝ ΣΤΗΝ ΚΑΘΥΣΤΕΡΗΣΗ	29
2.1 ΕΙΣΑΓΩΓΗ.....	29
2.2 ΧΑΡΑΚΤΗΡΙΣΤΙΚΑ ΚΑΘΥΣΤΕΡΗΣΕΩΝ ΤΟΥ ΔΙΚΤΥΟΥ ΑΠΟ ΑΚΡΗ-ΣΕ-ΑΚΡΗ	31
2.3 ΑΠΟΔΟΣΗ ΠΡΟΣΑΡΜΟΣΤΙΚΩΝ ΑΛΓΟΡΙΘΜΩΝ ΡΥΘΜΙΣΗΣ ΧΡΟΝΟΥ ΑΝΑΠΑΡΑΓΩΓΗΣ.....	32
2.4 ΠΡΟΣΑΡΜΟΣΤΙΚΟΙ ΑΛΓΟΡΙΘΜΟΙ ΥΠΟΛΟΓΙΣΜΟΥ ΧΡΟΝΟΥ ΑΝΑΠΑΡΑΓΩΓΗΣ.....	34
2.4.1 <i>Αλγόριθμος 1 (NeVoT 1.4)</i>	36
2.4.2 <i>Αλγόριθμος 2 (NeVoT 3.28)</i>	37
2.4.3 <i>Αλγόριθμος 3 (Αλγόριθμος Ανίχνευσης Στιγμαίων Αυξήσεων στις καθυστερήσεις του δικτύου)</i>	37
2.4.4 <i>Αλγόριθμος 4 (Μέθοδος Υπολογισμού «Κενών»)</i>	41
2.4.5 <i>Αλγόριθμος 5 (NLMS)</i>	45
2.4.6 <i>Αλγόριθμος ανίχνευσης «στιγμαίων αυξήσεων» με παράλληλη χρήση πρόβλεψης (Hybrid)</i>	47
2.4.7 <i>Πειραματικά αποτελέσματα</i>	48
ΚΕΦΑΛΑΙΟ 3	61
ΤΕΧΝΙΚΕΣ ΑΝΑΚΤΗΣΗΣ ΧΑΜΕΝΩΝ ΠΑΚΕΤΩΝ ΦΩΝΗΣ	61
3.1 ΕΙΣΑΓΩΓΗ.....	61
3.2 ΕΠΑΝΟΡΘΩΣΗ ΒΑΣΙΣΜΕΝΗ ΣΤΟΝ ΑΠΟΣΤΟΛΕΑ	61
3.2.1 <i>Forward Error Correction (FEC)</i>	62
3.2.1.1 <i>Media Independent FEC</i>	62
3.2.1.2 <i>Media Specific FEC</i>	64
3.2.2 <i>Διαφύλλωση</i>	65
3.2.3 <i>Επαναμετάδοση</i>	67
3.3 ΕΠΑΝΟΡΘΩΣΗ ΒΑΣΙΣΜΕΝΗ ΣΤΟΝ ΠΑΡΑΛΗΠΤΗ	67
3.3.1 <i>Insertion</i>	68
3.3.1.1 <i>Splicing</i>	69
3.3.1.2 <i>Silence Substitution</i>	69
3.3.1.3 <i>Noise Substitution</i>	69
3.3.1.4 <i>Packet Repetition</i>	69
3.3.2 <i>Interpolation</i>	70
3.3.2.1 <i>Waveform Substitution</i>	70
3.3.2.2 <i>Pitch Waveform Replication</i>	70
3.3.2.3 <i>Time Scale Modification</i>	70

3.3.3	<i>Regeneration</i>	71
3.3.3.1	Interpolation of Transmitted State.....	71
3.3.3.2	Model -Based Recovery	71
3.4	ΑΝΑΚΑΤΑΣΚΕΥΗ ΜΕΣΟΥ ΟΡΟΥ ΜΕ ΧΡΗΣΗ ΜΕΤΑΣΧΗΜΑΤΙΣΜΩΝ ΚΑΙ ΔΙΑΦΥΛΛΩΣΗΣ	72
3.4.1	<i>Ανακατασκευή μέσου όρου με χρήση μετασχηματισμού και διαφύλλωσης ενός επιπέδου</i>	73
3.4.1.1	Απώλεια ενός πακέτου σε ένα ζεύγος διαφύλλωσης.....	74
3.4.1.2	Πλήρης παράδοση ζεύγους διαφύλλωσης.....	76
3.4.2	<i>Ανακατασκευή μέσου όρου με χρήση μετασχηματισμού και διαφύλλωσης δυο επιπέδων</i>	76
3.4.3	<i>Πειραματικά αποτελέσματα</i>	78
3.5	ΑΝΑΚΑΤΑΣΚΕΥΗ GAUSS ΜΕ ΧΡΗΣΗ ΜΕΤΑΣΧΗΜΑΤΙΣΜΩΝ ΚΑΙ ΔΙΑΦΥΛΛΩΣΗΣ	92
3.5.1	<i>Ανακατασκευή Gauss με χρήση μετασχηματισμών και διαφύλλωσης ενός επιπέδου</i>	92
3.5.1.1	Απώλεια ενός πακέτου σε ένα ζεύγος διαφύλλωσης.....	92
3.5.1.2	Πλήρης παράδοση ζεύγους διαφύλλωσης.....	95
3.5.2	<i>Πειραματικά αποτελέσματα</i>	95
3.6	ΑΝΑΚΑΤΑΣΚΕΥΗ ΜΕΣΟΥ ΟΡΟΥ ΜΕ ΧΡΗΣΗ ΔΙΑΦΥΛΛΩΣΗΣ ΣΕ ΤΡΙΑ ΜΕΡΗ.....	100
3.6.1	<i>Απώλεια ενός πακέτου τριάδας διαφύλλωσης</i>	100
3.6.2	<i>Ανακατασκευή μέσου όρου με χρήση equalizer</i>	101
3.6.3	<i>Πειραματικά αποτελέσματα</i>	103
3.7	ΣΥΜΠΕΡΑΣΜΑΤΑ-ΠΡΟΤΑΣΕΙΣ.....	111
	ΚΕΦΑΛΑΙΟ 4.....	115
	ΕΠΙΣΚΟΠΗΣΗ ΕΡΓΑΣΙΑΣ – ΜΕΛΛΟΝΤΙΚΕΣ ΕΠΕΚΤΑΣΕΙΣ.....	115
	ΠΑΡΑΡΤΗΜΑ Α	117
	ΠΑΡΑΡΤΗΜΑ Β	119
	ΑΝΑΦΟΡΕΣ.....	121

Κατάλογος Εικόνων

Εικόνα 2.4.1 Εκτιμήσεις καθυστέρησης αναπαραγωγής για τους αλγορίθμους 1 & 2...	38
Εικόνα 2.4.2 Εκτιμήσεις καθυστέρησης αναπαραγωγής για τους αλγορίθμους 1, 2, SDA	40
Εικόνα 2.4.3 Βέλτιστο «κενό» για ένα πιο αργό παίξιμο	42
Εικόνα 2.4.4 Βέλτιστο «κενό» για ένα πιο γρήγορο παίξιμο	42
Εικόνα 2.4.5 Προσαρμοστικό φίλτρο ενός βήματος.....	46
Εικόνα 2.4.6 Συμπεριφορά του SDA και του υβριδικού.....	47
Εικόνα 2.4.7 Σύγκριση αλγορίθμων 1,2 & SDA «ίχνος 1».....	49
Εικόνα 2.4.8 Σύγκριση αλγορίθμων 1,2 & SDA «ίχνος 2».....	49
Εικόνα 2.4.9 Σύγκριση αλγορίθμων 1,2 & SDA «ίχνος 3».....	50
Εικόνα 2.4.10 Σύγκριση αλγορίθμων 1,2 & SDA «ίχνος 4».....	50
Εικόνα 2.4.11 Σύγκριση αλγορίθμων 1,2 & SDA «ίχνος 5».....	51
Εικόνα 2.4.12 Σύγκριση αλγορίθμων 1,2 & SDA «ίχνος 6».....	51
Εικόνα 2.4.13 Σύγκριση αλγορίθμων 1,GAP & NLMS «ίχνος 1»	53
Εικόνα 2.4.14 Σύγκριση αλγορίθμων 1,GAP & NLMS «ίχνος 2»	53
Εικόνα 2.4.15 Σύγκριση αλγορίθμων 1,GAP & NLMS «ίχνος 3»	54
Εικόνα 2.4.16 Σύγκριση αλγορίθμων 1,GAP & NLMS «ίχνος 4»	54
Εικόνα 2.4.17 Σύγκριση αλγορίθμων 1,GAP & NLMS «ίχνος 5»	55
Εικόνα 2.4.18 Σύγκριση αλγορίθμων 1,GAP & NLMS «ίχνος 6»	55
Εικόνα 2.4.19 Σύγκριση αλγορίθμων NLMS & Hybrid «ίχνος 1».....	57
Εικόνα 2.4.20 Σύγκριση αλγορίθμων NLMS & Hybrid «ίχνος 2».....	57
Εικόνα 2.4.21 Σύγκριση αλγορίθμων NLMS & Hybrid «ίχνος 3».....	58
Εικόνα 2.4.22 Σύγκριση αλγορίθμων NLMS & Hybrid «ίχνος 4».....	58
Εικόνα 2.4.23 Σύγκριση αλγορίθμων NLMS & Hybrid «ίχνος 5».....	59
Εικόνα 2.4.24 Σύγκριση αλγορίθμων NLMS & Hybrid «ίχνος 6».....	59

Κατάλογος Πινάκων

Πίνακας 2.4.1 Πληροφορίες για τα «ίχνη» που χρησιμοποιήθηκαν	35
Πίνακας 2.4.2 Παράμετροι του προσαρμοστικού αλγορίθμου	46
Πίνακας 3.4.1 Χαρακτηριστικά αρχείων ήχου	79

Κατάλογος Πλαισίων

Πλαίσιο 2.4.1 Αλγόριθμος Ανίχνευσης Αιχμών (<i>SDA</i>)	39
Πλαίσιο 2.4.2 Υπολογισμός καθυστέρησης αναπαραγωγής για τον <i>SDA</i>	40
Πλαίσιο 2.4.3 Αλγόριθμος υπολογισμού κενών (<i>Gap</i>)	44
Πλαίσιο 2.4.4 Υπολογισμός Βέλτιστου Κενού για τον αλγόριθμο <i>Gap</i>	45
Πλαίσιο 2.4.5 Αλγόριθμος <i>Hybrid (NLMS & SDA)</i>	48

ΚΕΦΑΛΑΙΟ 1

Εισαγωγή

1.1 Ορισμός

Παραδοσιακά η μετάδοση φωνής λάμβανε χώρα μέσω του Δημοσίου Δικτύου Μεταγωγής Τηλεφωνημάτων (Public Switched Telephone Network). Όμως στις μέρες μας αφενός λόγω της ραγδαίας εξάπλωσης και χρήσης του διαδικτύου και αφετέρου της παράλληλης αύξηση της ισχύος των υπολογιστικών συστημάτων, η διαδικασία μετάδοσης της φωνής άρχισε να πραγματοποιείται και μέσω του διαδικτύου (Internet).

Επιχειρήσεις και οργανισμοί σε όλο τον κόσμο, αναζητούν ένα τρόπο να μειώσουν τις διαρκώς αυξανόμενες τηλεπικοινωνιακές δαπάνες. Η χρησιμοποίηση κοινών τηλεπικοινωνιακών πόρων, για την μετάδοση της φωνής και των δεδομένων, τους παρέχει ένα τρόπο εξοικονόμησης εσόδων. Παράλληλα, η παροχή αυτού του είδους ολοκληρωμένων υπηρεσιών αποτελεί ανερχόμενη προτεραιότητα και για τους διαχειριστές δικτύων. Επιχειρήσεις και οργανισμοί επιδιώκουν λύσεις που θα τους επιτρέπουν αφενός να εκμεταλλευτούν την χωρητικότητα των ευρυζώνιων δικτύων για την μεταφορά φωνής και δεδομένων και επιπροσθέτως να χρησιμοποιούν το διαδίκτυο και τα εταιρικά δίκτυα (Intranets) σαν εναλλακτικές οικονομικότερες λύσεις.

Αν και τα οφέλη, που προκύπτουν από την χρήση εφαρμογών μετάδοσης φωνής μέσω του διαδικτύου είναι προφανή και ενώ υπάρχουν αρκετά εμπορικά προϊόντα στην αγορά, μόνο ένα μικρό ποσοστό χρηστών του διαδικτύου χρησιμοποιούν αυτή την τεχνολογία [49]. Τον βασικότερο λόγο, αποτελεί η ποιότητα της παρεχόμενης υπηρεσίας η οποία δεν μπορεί να συγκριθεί με την αντίστοιχη του τηλεφωνικού δικτύου.

1.2 Παρουσίαση του προβλήματος

Μια τυπική εμπορική εφαρμογή μετάδοσης φωνής, ακολουθεί τα εξής βασικά στάδια: αρχικά ψηφιοποιεί το αναλογικό σήμα φωνής, το συσκευάζει σε πακέτα και τέλος το μεταδίδει μέσω του διαδικτύου.

Όμως το Internet σχεδιάστηκε να υποστηρίζει μεταδόσεις δεδομένων μη πραγματικού χρόνου όπως για παράδειγμα (e-mail, ftp, ...). Για αυτό τον λόγο το πρωτόκολλο στο

επίπεδο δικτύου του Internet (σύμφωνα με τα 7 επίπεδα του μοντέλου OSI), για την μετάδοση των πακέτων, παρέχει την υπηρεσία βέλτιστης προσπάθειας (best effort). Αυτή η υπηρεσία δεν δίνει καμία εγγύηση, αφενός για την συνολική καθυστέρηση παράδοσης ενός μεμονωμένου πακέτου και αφετέρου ούτε για το εύρος της διακύμανσης των καθυστερήσεων (jitter) αλλά και της ακεραιότητας (packet loss) μιας ροής πακέτων.

Ουσιαστικά λοιπόν η υπηρεσία βέλτιστης προσπάθειας, παρέχει ένα κανάλι μετάδοσης του οποίου χαρακτηριστικά όπως η καθυστέρηση και ο ρυθμός εμφάνισης σφαλμάτων είναι χρονικά μεταβαλλόμενα [1,2]. Τα χαρακτηριστικά αυτά δεν είναι δυνατόν να είναι γνωστά εκ των προτέρων, από την στιγμή που εξαρτώνται άμεσα από την συμπεριφορά (απρίορι άγνωστη) των άλλων καναλιών μετάδοσης του δικτύου. Το γεγονός αυτό καθιστά αδύνατη την παροχή εγγυήσεων όπως ελάχιστου ρυθμού εμφάνισης σφαλμάτων ή μέγιστης καθυστέρησης. Όμως οι εφαρμογές πραγματικού χρόνου, όπως αυτές της μετάδοσης φωνής μέσω του διαδικτύου χρειάζονται ένα κάτω όριο εγγυήσεων για να αποδίδουν ικανοποιητικά. Το δίκτυο λοιπόν και πιο συγκεκριμένοι ο βαθμός συμφόρησης του, προκαλεί τα πιο έντονα προβλήματα σχετικά με την παρεχόμενη ποιότητα υπηρεσίας στην ροή των πακέτων φωνής. Για την αντιμετώπιση του προβλήματος υπάρχουν δυο προσεγγίσεις:

Η πρώτη προσέγγιση αφορά την επέκταση των παρόντων πρωτοκόλλων και αλλαγή των αντίστοιχων αλγορίθμων χρονοπρογραμματισμού (scheduling) με τέτοιο τρόπο ώστε να παρέχονται οι επιθυμητές εγγυήσεις. Βέβαια για να επιτευχθεί κάτι τέτοιο, θα πρέπει οι μηχανισμοί ελέγχου αποδοχής (admission control), δέσμευσης αλλά και χρονοπρογραμματισμού να υλοποιούνται από το δίκτυο [3] (π.χ-RSVP).

Εναλλακτική προσέγγιση, αποτελεί η προσαρμογή των εφαρμογών στην ήδη υπάρχουσα υπηρεσία που παρέχεται από το δίκτυο. Πρακτικά αυτό σημαίνει ότι οι εφαρμογές οφείλουν να προσαρμόζονται στα χρονικά μεταβαλλόμενα χαρακτηριστικά της σύνδεσης μέσω της οποίας μεταδίδονται τα πακέτα των πολυμεσικών δεδομένων, κατά τρόπο τέτοιο ώστε να μεγιστοποιείται η ποιότητα της παρεχόμενης υπηρεσίας.

Πειραματικά αποτελέσματα αποδεικνύουν ότι η ακουστική ποιότητα των δεδομένων που λαμβάνει ο παραλήπτης, εξαρτάται τόσο από τον αριθμό των χαμένων πακέτων, όσο και από την διακύμανση της καθυστέρησης (jitter) μεταξύ διαδοχικών πακέτων που έχουν παραδοθεί. Άρα, συνοψίζοντας τα πιο σημαντικά χαρακτηριστικά του δικτύου που αφορούν εφαρμογές μετάδοσης φωνής μέσω του διαδικτύου, είναι η διακύμανση της καθυστέρησης (jitter) και η κατανομή εμφάνισης σφαλμάτων στο κανάλι. Επιπροσθέτως για αλληλεπιδραστικές εφαρμογές πραγματικού χρόνου, όπως

στην περίπτωση μιας τηλεσυνδιάσκεψης, βασικό ρόλο παίζει και η μέση συνολική καθυστέρηση η οποία οφείλει να είναι μικρή ώστε να επιτυγχάνεται η αλληλεπίδραση μεταξύ των συμμετεχόντων. Σκοπός αυτής της προσέγγισης είναι η ανάπτυξη μηχανισμών που προσπαθούν να εξαλείψουν ή τουλάχιστον να ελαχιστοποιήσουν τα αποτελέσματα αφενός της απώλειας πακέτων και αφετέρου των διακυμάνσεων στην καθυστέρηση, στην ακουστική ποιότητα που λαμβάνει ο παραλήπτης.

Είναι φανερό ότι η πρώτη προσέγγιση, είναι αρκετά να δύσκολο να υλοποιηθεί στο άμεσο μέλλον. Εταιρίες αλλά και οργανισμοί δεν είναι διατεθειμένοι, κυρίως από οικονομικής άποψης, να αλλάξουν άμεσα ολόκληρη την τηλεπικοινωνιακή τους υποδομή (δρομολογητές, μεταγωγείς, κτλ). Συνήθως η αναβάθμιση των αντιστοίχων τηλεπικοινωνιακών συσκευών γίνεται σταδιακά. Από την άλλη μεριά, η δεύτερη προσέγγιση φαντάζει πιο ρεαλιστική για άμεση εφαρμογή. Αποτελεί σαφώς ευκολότερο να αλλάξουμε μια εφαρμογή τηλεσυνδιάσκεψης στον προσωπικό μας υπολογιστή παρά να αναβαθμίσουμε την υπάρχουσα υποδομή του δικτύου. Βέβαια, δεν πρέπει να ξεχνάμε ότι μια εφαρμογή όσο καλή και να είναι δεν μπορεί σε καμία περίπτωση να παρέχει την ποιότητα που θα απέδιδε μια πιθανή αναβάθμιση του δικτύου. Έτσι, έως ότου το δίκτυο να αποκτήσει την δυνατότητα παροχής εγγυήσεων σε θέματα καθυστερήσεων και απωλειών, η δεύτερη προσέγγιση θα χρησιμοποιείται.

Στα πλαίσια αυτής της εργασίας ασχοληθήκαμε σε θεωρητικό αλλά και σε πειραματικό επίπεδο με την δεύτερη προσέγγιση.

1.3 Παρουσίαση παρόντων πρωτοκόλλων επικοινωνίας

Όπως προαναφέραμε το πρωτόκολλο επιπέδου δικτύου υποστηρίζει μόνο την υπηρεσία βέλτιστης προσπάθειας. Η υπηρεσία αυτή καθιστά πολύ δύσκολη την χρήση πρωτοκόλλων επιπέδου μεταφοράς για την πραγματοποίηση μεταδόσεων πραγματικού χρόνου. Το Transmission Control Protocol – TCP [50] και το User Datagram Protocol - UDP [51], αποτελούν τα πρωτόκολλα επιπέδου μεταφοράς του διαδικτύου. Το TCP είναι ένα πρωτόκολλο με σύνδεση (connection-oriented) και χαρακτηρίζεται από την χωρίς λάθη και σε σωστή σειρά, παράδοση των πακέτων. Αυτό επιτυγχάνεται με μηχανισμούς ανίχνευσης λαθών και επαναμετάδοσης (σε περίπτωση σφάλματος) των δεδομένων. Όμως ακριβώς αυτή του η ιδιότητα καθιστά απαγορευτική την χρήση του για την μετάδοση δεδομένων πραγματικού χρόνου, λόγω των μεγάλων καθυστερήσεων που υπεισέρχονται από τις επαναμεταδόσεις. Από την άλλη μεριά, το UDP είναι ένα πρωτόκολλο χωρίς σύνδεση (connectionless) και δεν παρέχει καμία εγγύηση για την παράδοση ενός πακέτου. Επίσης, βασικό του χαρακτηριστικό είναι η πιθανότητα παράδοσης των πακέτων εκτός σειράς. Κατά αυτόν τον τρόπο, μια εφαρμογή που χρησιμοποιεί το UDP είναι υπεύθυνη τόσο για

την σωστή αναδιάταξη των πακέτων που έχουν παραληφθεί όσο και για τον τρόπο αντιμετώπισης των λαθών. Αυτή του η ιδιότητα δεν είναι κατά ανάγκη αρνητική, γιατί δίνει στην εκάστοτε εφαρμογή μεγάλη ευελιξία στον τομέα καθορισμού εξειδικευμένων απαιτήσεων.

Οι πιο πρόσφατες εφαρμογές πραγματικού χρόνου μετάδοσης πολυμεσικών δεδομένων μέσω του διαδικτύου, βασίζονται στο UDP και ακολουθούν το πρότυπο IETF Real-time Transport Protocol (RTP) [47,48]. Το RTP δεν είναι ένα πρωτόκολλο επιπέδου μεταφοράς, αλλά πρωτόκολλο επιπέδου εφαρμογής. Αρχικά είχε σχεδιαστεί για multicast μεταδόσεις πολυμεσικών δεδομένων πραγματικού χρόνου, αλλά μπορεί να εφαρμοστεί και σε περιπτώσεις unicast μεταδόσεων.

Βασικές του λειτουργίες αποτελούν η παροχή χρονικών σφραγίδων (timestamping), η διαδοχική αρίθμηση (sequence numbering) και γενικά η παροχή μηχανισμών χρονισμού. Μέσω αυτών των μηχανισμών το RTP, παρέχει μια υπηρεσία μεταφοράς, από άκρη σε άκρη (end-to-end) πολυμεσικών δεδομένων πραγματικού χρόνου.

Τυπικά το RTP, «τρέχει» πάνω από το UDP (βλ. εικόνα 1.3.1) και κατά αυτόν τον τρόπο είναι σε θέση να χρησιμοποιήσει τους μηχανισμούς πολυπλεξίας και ελέγχου αθροίσματος για λάθη (checksum). Η προτίμηση του UDP από το TCP έγινε για δυο λόγους:

Πρώτον γιατί το RTP είχε σχεδιαστεί για multicast μεταδόσεις, όμως η φύση των χαρακτηριστικών του TCP δεν του επιτρέπει να αποδίδει ικανοποιητικά σε τέτοιες καταστάσεις.

Και δεύτερον, για την μετάδοση πολυμεσικών δεδομένων πραγματικού χρόνου η αξιόπιστη παράδοση τους δεν είναι τόσο σημαντική όσο η έγκαιρη παράδοση τους. Επιπροσθέτως η αξιόπιστη παράδοση των δεδομένων που παρέχεται από επαναμεταδόσεις είναι μη επιθυμητή. Για παράδειγμα σε περίπτωση συμφόρησης του δικτύου, κάποια πακέτα μπορεί να χαθούν με αποτέλεσμα την μείωση της ποιότητας αναπαραγωγής ήχου στον παραλήπτη. Αν το πρωτόκολλο επιμένει στην αξιόπιστη μετάδοση των δεδομένων και προκαλέσει επαναμεταδόσεις των πακέτων, θα αυξηθεί αφενός η συνολική καθυστέρηση και αφετέρου θα επιβαρύνει την ήδη βεβαρημένη κατάσταση του δικτύου.



Εικόνα 1.3.1 *Μετάδοση πολυμεσικών δεδομένων μέσω RTP*

Επίσης το RTP σχεδιάστηκε έτσι ώστε να συνυπάρχει με ένα συμπληρωματικό πρωτόκολλο ελέγχου το IETF Real Time Control Protocol – RTCP [47,48]. Κατά την διάρκεια μιας RTP συνόδου (RTP session), οι συμμετέχοντες σε αυτό (αποστολείς και παραλήπτες), ανταλλάσσουν κατά τακτά χρονικά διαστήματα RTCP πακέτα μεταφέροντας αναδραστική πληροφορία που αφορά την ποιότητα παράδοσης των πολυμεσικών δεδομένων.

Αναλυτικότερα, έχουν οριστεί δυο συγκεκριμένα RTCP πακέτα ελέγχου για την μεταφορά αυτής της πληροφορίας:

- *Αναφορά Παραλήπτη (Receiver Report)*: Στέλνονται από τους παραλήπτες και περιέχουν πληροφορία σχετικά με την ποιότητας λήψης των πολυμεσικών δεδομένων, όπως το μέγιστο αριθμό πακέτων που έχουν λάβει, τον αριθμό πακέτων που χάθηκαν, την διακύμανση της καθυστέρησης (jitter) και χρονικές σφραγίδες (timestamps) για τον υπολογισμό της συνολικής καθυστέρησης (round-trip delay).
- *Αναφορά Αποστολέα (Sender Report)*: Στέλνονται από τους αποστολείς και εκτός των δεδομένων που περιέχονται και στην αναφορά του παραλήπτη, περιλαμβάνει πληροφορία εσωτερικού συγχρονισμού των πολυμεσικών δεδομένων. π.χ lip-synchronization σε περίπτωση τηλεσυνδιάσκεψης.

Συνήθως στην πράξη το RTP, ενσωματώνεται στην εφαρμογή με αποτέλεσμα πολλά θέματα όπως η ανάκτηση χαμένων πακέτων και ο έλεγχος του βαθμού συμφόρησης του δικτύου να υλοποιούνται σε επίπεδο εφαρμογής.

1.4 Δομή της εργασίας

Στα πλαίσια αυτής της εργασίας ασχοληθήκαμε με την μελέτη και υλοποίηση μηχανισμών βελτίωσης της παρεχόμενης ποιότητας υπηρεσίας εφαρμογών μετάδοσης φωνής μέσω διαδικτύου, σε πραγματικό χρόνο. Η επιδιωκόμενη βελτίωση της ποιότητας επιτυγχάνεται προτείνοντας νέες τεχνικές προσαρμογής των εφαρμογών στα χρονικά μεταβαλλόμενα χαρακτηριστικά μιας σύνδεσης.

Πιο συγκεκριμένα στο Κεφάλαιο 2, μελετώνται οι πιο γνωστοί αλγόριθμοι εξάλειψης των διακυμάνσεων στην καθυστέρηση (jitter elimination), αναφέρονται τα προβλήματα τους και τέλος προτείνεται ένας καινούργιος αλγόριθμος.

Στο Κεφάλαιο 3, παρουσιάζονται μέθοδοι ανάκαμψης σε περιπτώσεις εμφάνισης σφαλμάτων κατά την μετάδοση των δεδομένων και τέλος προτείνεται ένα καινούργιο προσαρμοστικό σχήμα ανακατασκευής πακέτων που έχουν απολεσθεί.

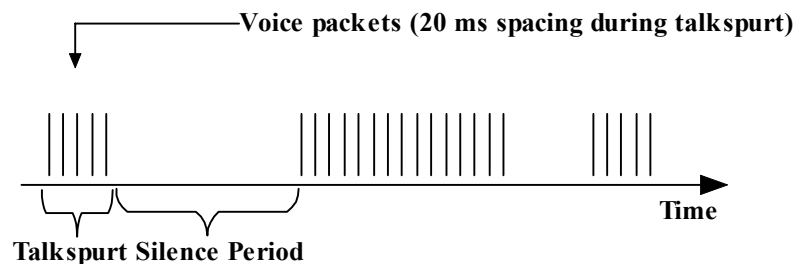
Τέλος στο Κεφάλαιο 4, συνοψίζονται οι παραπάνω τεχνικές και αναφέρονται οι μελλοντικές επεκτάσεις αυτών.

ΚΕΦΑΛΑΙΟ 2

Μηχανισμοί εξάλειψης διακυμάνσεων στην καθυστέρηση

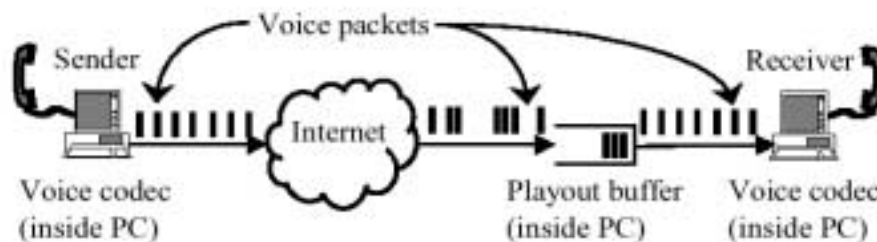
2.1 Εισαγωγή

Οι εφαρμογές μετάδοσης φωνής μέσω του διαδικτύου, λειτουργούν ως εξής: Αρχικά κατά περιοδικά χρονικά διαστήματα συγκεντρώνουν τα δείγματα φωνής που δημιουργούνται στον αποστολέα, τα ψηφιοποιούν και τα συσκευάζουν σε πακέτα. Τέλος, τα παραγόμενα πακέτα μεταδίδονται μέσω του διαδικτύου στους αντίστοιχους παραλήπτες. Για μεγαλύτερη αποδοτικότητα, η διαδικασία παραγωγής ήχου στον αποστολέα χωρίζεται περιόδους ομιλίας (talkspurts) και σε περιόδους σιωπής (βλ. εικόνα 2.1.1). Από την άλλη μεριά, στον παραλήπτη για πετύχουμε πιστή αναπαραγωγή του ήχου τα δεδομένα των πακέτων μιας περιόδου ομιλίας πρέπει να «αναπαραχθούν» με τον ίδιο περιοδικό τρόπο με τον οποίο δημιουργήθηκαν.



Εικόνα 2.1.1 Περίοδοι ομιλίας και σιωπής

Αν το δίκτυο δεν χαρακτηρίζεται από διακυμάνσεις στην καθυστέρηση, ο παραλήπτης είναι σε θέση να αναπαράγει το κάθε πακέτο ακριβώς μόλις το παραλάβει. Όμως στο διαδίκτυο, καταστάσεις οι οποίες χαρακτηρίζονται από έγκαιρη, χωρίς διακυμάνσεις στην καθυστέρηση και σε σωστή σειρά παράδοση των δεδομένων, παρατηρούνται τουλάχιστο σπάνια. Έτσι, για την αντιμετώπιση αυτών των φαινομένων ο παραλήπτης χρησιμοποιεί έναν ενταμιευτή εξομάλυνσης (βλ. εικόνα 2.1.2)



Εικόνα 2.1.2 Χρήση του ενταμιευτή εξομάλυνσης

Τα πακέτα που παραδόθηκαν στον προορισμό τους, αρχικά τοποθετούνται σε έναν ενταμιευτή εξομάλυνσης, έτσι ώστε η περιοδική αναπαραγωγή των πακέτων που ανήκουν στην ίδια περίοδο ομιλίας να καθυστερηθεί για κάποιο χρονικό διάστημα μετά την λήψη του πρώτου πακέτου της συγκεκριμένης περιόδου ομιλίας. Την καθυστέρηση αυτή θα την ονομάζουμε «καθυστέρηση αναπαραγωγής» (playout delay) μιας περιόδου ομιλίας. Είναι εμφανές ότι όσο μεγαλύτερη είναι η παραπάνω καθυστέρηση τόσο νωρίτερα σε σχέση με τον προγραμματισμένο χρόνο αναπαραγωγής (playout time), έχει ληφθεί το συγκεκριμένο πακέτο. Ωστόσο, υπερβολικά μεγάλες καθυστερήσεις μπορούν να επηρεάσουν αρνητικά την ποιότητα της ανθρώπινης συνομιλίας. Έτσι, κατά την περίπτωση που κάποιο πακέτο παραδοθεί στον παραλήπτη μετά τον προγραμματισμένο χρόνο αναπαραγωγής του, τότε αγνοείται και για την εφαρμογή θεωρείται χαμένο. Υπάρχει λοιπόν, μια αντισταθμιστική σχέση μεταξύ της διάρκειας της καθυστέρησης αναπαραγωγής και του εύρους των απωλειών λόγω των αργοπορημένων αφίξεων. Γενικά καθυστερήσεις μεταξύ της δημιουργίας περιόδων ομιλίας και της αναπαραγωγής αυτών στον παραλήπτη, μικρότερες των 400ms [4] και ποσοστά απωλειών μέχρι 5% [5] θεωρούνται ανεκτά για την πραγματοποίηση ανθρώπινων συνομιλιών.

Η διάρκεια καθυστέρησης αναπαραγωγής μιας περιόδου ομιλίας, μπορεί να είναι σταθερή κατά την διάρκεια μιας συνόδου ή να ρυθμίζεται δυναμικά. Όμως στο διαδίκτυο η διάρκεια των συνολικών από άκρη-σε-άκρη καθυστερήσεων κυμαίνεται σημαντικά με αποτέλεσμα, η χρήση σταθερών και μη-προσαρμοστικών καθυστερήσεων αναπαραγωγής να παρέχει χαμηλής ποιότητας ήχου στις αλληλεπιδραστικές εφαρμογές μετάδοσης φωνής.

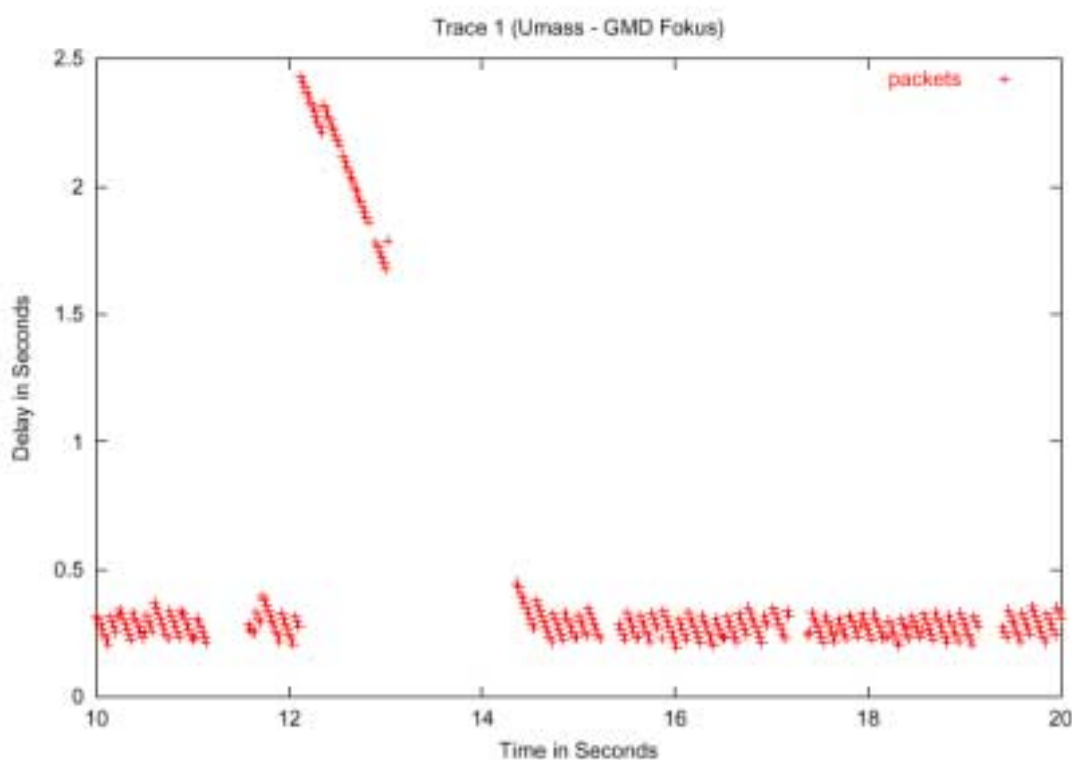
Όσον αφορά την δυναμικά προσαρμοστική ρύθμιση της διάρκειας της καθυστέρησης αναπαραγωγής υπάρχουν δυο προσεγγίσεις: ρύθμιση ανά περίοδο ομιλίας και ρύθμιση ανά πακέτο. Η πρώτη χρησιμοποιεί την ίδια τιμή καθυστέρησης αναπαραγωγής κατά την διάρκεια μιας συγκεκριμένης περιόδου ομιλίας και παράλληλα επιτρέπει την δυναμική ρύθμιση της, μεταξύ διαφορετικών περιόδων ομιλίας. Αποτέλεσμα αυτού, αποτελεί η πιστή αναπαραγωγή της αρχικής περιοδικής φύσης του σήματος φωνής. Βέβαια, η δυναμική ρύθμιση της καθυστέρησης αναπαραγωγής, μεταξύ των περιόδων ομιλίας, προκαλεί τεχνητές αυξομειώσεις στην διάρκεια των περιόδων σιωπής. Αυτό μπορεί να μην γίνει αντιληπτό κατά την συνομιλία αν το μέγεθος τους είναι σχετικά μικρό [6]. Κατά την δεύτερη προσέγγιση, η διάρκεια της καθυστέρησης αναπαραγωγής ποικίλει από πακέτο σε πακέτο. Όμως η ανά πακέτο δυναμική ρύθμιση της καθυστέρησης εισάγει κενά μέσα στις περιόδους ομιλίας τα οποία με τη σειρά τους καταστρέφουν την ποιότητα του ήχου.

Στην συνέχεια αυτού του κεφαλαίου εξετάζονται τα χαρακτηριστικά των συνολικών καθυστερήσεων που παρατηρούνται στο διαδίκτυο (end-to-end delays), καθώς και η

επιρροή τους πάνω στους αλγορίθμους υπολογισμού του χρόνου αναπαραγωγής. Επίσης παρουσιάζονται οι μαθηματικές σχέσεις που χρειάζονται για την αξιολόγηση των αλγορίθμων και τέλος ακολουθεί η ανάλυση αυτών.

2.2 Χαρακτηριστικά των «από άκρη-σε-άκρη» καθυστερήσεων του δικτύου

Προηγούμενες έρευνες [1], έχουν δείξει την ύπαρξη «στιγμαίων αυξήσεων» (spikes) στις συνολικές καθυστερήσεις του διαδικτύου. Ως «στιγμαία αύξηση» ορίζουμε την ξαφνική και μεγάλης κλίμακας αύξηση της συνολικής καθυστέρησης του δικτύου, ακολουθούμενη από μια σειρά πακέτων τα οποία καταφθάνουν σχεδόν ταυτόχρονα οδηγώντας κατά αυτόν τον τρόπο το φαινόμενο στην ολοκλήρωσή του. Στην εικόνα 2.2.1 παρουσιάζεται μια τυπική στιγμαία αύξηση στην τιμή της καθυστέρησης. Κάθε σημείο αναπαριστά ένα πακέτο το οποίο αφικνείται την χρονική στιγμή που υποδεικνύεται από τον άξονα x , και έχει υποστεί καθυστέρηση ίση με την αντίστοιχη τιμή στον άξονα y [8].



Εικόνα 2.2.1 Στιγμαία αύξηση στην τιμή της καθυστέρησης

Αυτή η απότομη αύξηση της καθυστέρησης ακολουθούμενη από μια γραμμική και μονοτονική μείωση, οφείλεται αφενός στην περιοδικότητα δημιουργίας και αποστολής των πακέτων και αφετέρου στον αυξημένο βαθμό συμφόρησης του δικτύου. Πιο συγκεκριμένα, σε κατάσταση συμφόρησης τα πακέτα των πολυμεσικών δεδομένων που ανήκουν στην ίδια ροή, συσσωρεύονται στην ουρά κάποιου δρομολογητή πίσω από

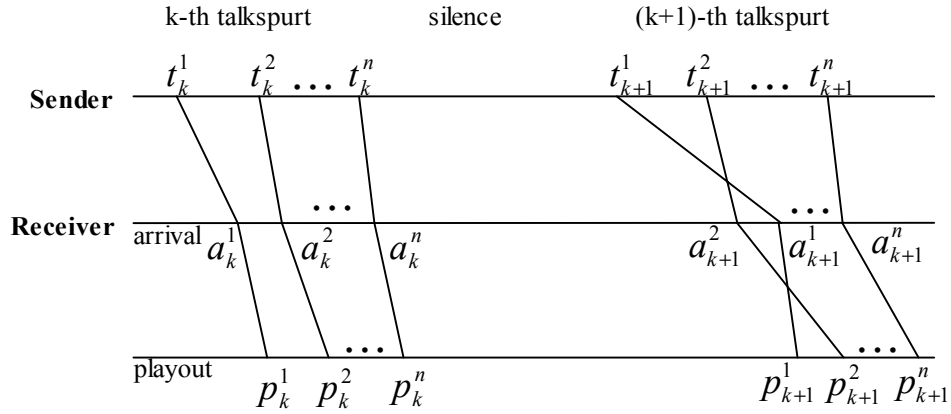
πακέτα άλλων ροών. Έτσι, μόλις εξυπηρετηθούν τα προπορευόμενα πακέτα δρομολογούνται και τα συσσωρευμένα πακέτα φωνής, δημιουργώντας έτσι το φαινόμενο των στιγμιαίων αυξήσεων στις καθυστερήσεις.

Όταν μια στιγμιαία αύξηση «περιέχεται» κατάλληλα μέσα σε μια περίοδο ομιλίας, η επόμενη ευκαιρία να αλλαχθεί η καθυστέρηση αναπαραγωγής (ουσιαστικά στην αρχή της επόμενης περιόδου ομιλίας) παρουσιάζεται αφού αυτή έχει ήδη τελειώσει. Σε αυτή την περίπτωση δεν είναι δυνατό ένας προσαρμοστικός αλγόριθμος αναπαραγωγής να αντιδράσει στο φαινόμενο αυτό, αφού αυτό έχει ήδη τελειώσει μέχρι να ξεκινήσει η επόμενη περίοδος ομιλίας και η συνολική καθυστέρηση έχει επανέλθει στα κανονικά της επίπεδα. Τα πακέτα που καθυστέρησαν υπερβολικά λόγω της στιγμιαία αύξησης της καθυστέρησης, θεωρούνται από τον αλγόριθμο ως χαμένα. Όμως όταν το φαινόμενο αυτό εκτείνεται σε πολλές περιόδους ομιλίας, είναι πολύ βασικό να υπάρξει ταχύτατη αντίδραση, ώστε να ελαχιστοποιηθούν οι απώλειες λόγω αργοπορημένων αφίξεων των πακέτων. Τέλος, πρέπει να σημειωθεί ότι οι τιμές των καθυστερήσεων σε κανονικές καταστάσεις παρουσιάζουν μικρότερη διακύμανση, συγκρίσει αυτών που εμφανίζονται κατά τα φαινόμενα των στιγμιαίων αυξήσεων.

2.3 Απόδοση προσαρμοστικών αλγορίθμων ρύθμισης χρόνου αναπαραγωγής

Η αντισταθμιστική σχέση μεταξύ της μέσης καθυστέρησης αναπαραγωγής και του ποσοστού απωλειών λόγω των καθυστερημένων αφίξεων, μπορεί να χρησιμοποιηθεί ως μέτρο απόδοσης μεταξύ των προσαρμοστικών αλγορίθμων ρύθμισης του χρόνου αναπαραγωγής. Η καθυστέρηση και οι απώλειες, εξετάζονται σε βάση ανά πακέτο και όχι ανά περίοδο ομιλίας για δυο λόγους: Αφενός, γιατί η διάρκεια των περιόδων ομιλίας εξαρτάται άμεσα από τον αλγόριθμο ανίχνευσης σιωπής που χρησιμοποιεί η εκάστοτε εφαρμογή και αφετέρου γιατί διαφορετικές περιόδοι ομιλίας έχουν διαφορετική διάρκεια.

Ως καθυστέρηση αναπαραγωγής (playout delay) ή για να γίνουμε πιο ακριβής από άκρη-σε-άκρη καθυστέρηση, ορίζουμε την διαφορά μεταξύ του χρόνου που θα αναπαραχθεί ένα πακέτο στον παραλήπτη και του χρόνου που δημιουργήθηκε αυτό στον αποστολέα. Στην εικόνα 2.3.1, παρατηρώντας πληροφορίες σχετικά με τις χρονικές εξαρτήσεις πακέτων και περιόδων ομιλίας, είμαστε σε θέση να ορίσουμε την μέση καθυστέρηση αναπαραγωγής [9].



Εικόνα 2.3.1 Χρονικές εξαρτήσεις μεταξύ k -οστής περιόδου ομιλίας & i -οστού πακέτου

Υποθέτοντας ότι έχουμε M περιόδους ομιλίας, μπορούμε να ορίσουμε τις παρακάτω ποσότητες:

- t_k^i : Η χρονική σφραγίδα (timestamp) του αποστολέα για το i -οστό πακέτο της k -οστής περιόδου ομιλίας
- a_k^i : Η χρονική σφραγίδα (timestamp) του παραλήπτη για το i -οστό πακέτο της k -οστής περιόδου ομιλίας
- n_k : Το πλήθος των πακέτων στην k -οστή περίοδο ομιλίας. Αναφερόμαστε σε αυτά που αφίχθησαν στον παραλήπτη
- N : Ο συνολικός αριθμός των πακέτων

$$\text{Άρα ισχύει } N = \sum_{k=1}^M n_k \quad (2.1)$$

Η χρονική στιγμή που θα αναπαραχθεί ένα πακέτο εξαρτάται από τον αλγόριθμο που χρησιμοποιεί ο παραλήπτης για να εκτιμήσει την καθυστέρηση αναπαραγωγής του. Έστω A ο αλγόριθμος υπολογισμού αυτής της χρονικής στιγμής. Τότε το $p_k^i(A)$ αποτελεί την χρονική στιγμή που θα αναπαραχθεί το i -οστό πακέτο της k -οστής περιόδου ομιλίας βάση του αλγορίθμου A . Αν το συγκεκριμένο πακέτο φθάσει αργότερα από την χρονική στιγμή $p_k^i(A)$, δηλαδή $p_k^i(A) < a_k^i$, τότε θεωρείται χαμένο. Διαφορετικά, το πακέτο θα αναπαραχθεί με καθυστέρηση αναπαραγωγής ίση με $p_k^i(A) - t_k^i$. Ορίζουμε το $r_k^i(A)$ ως την ενδεικτική μεταβλητή για το κατά πόσο το i -οστό πακέτο της k -οστής περιόδου ομιλίας λαμβάνεται από τον παραλήπτη πριν από τον προϋπολογισμένο χρόνο αναπαραγωγής, σύμφωνα με τον αλγόριθμο A .

$$\text{Άρα } r_k^i(A) = \begin{cases} 0, & p_k^i(A) < a_k^i \\ 1, & \text{otherwise} \end{cases} \quad (2.2)$$

Έτσι, ο συνολικός αριθμός των πακέτων $N(A)$ που θα αναπαραχθούν βάση του αλγορίθμου A δίνεται από την παρακάτω σχέση:

$$N(A) = \sum_{k=1}^M \sum_{i=1}^{n_k} r_k^i(A) \quad (2.3)$$

Και η μέση καθυστέρηση αναπαραγωγής, αυτών που τελικά αναπαράχθηκαν ορίζεται ως εξής:

$$\frac{1}{N(A)} \sum_{k=1}^M \sum_{i=1}^{n_k} r_k^i(A) (p_k^i(A) - t_k^i) \quad (2.4)$$

Τέλος, αν υπάρχουν N συνολικά πακέτα και από αυτά βάση του αλγορίθμου A αναπαράχθηκαν τα $N(A)$, το ποσοστό των απωλειών l δίνεται από την σχέση:

$$l = \frac{N - N(A)}{N} * 100 \quad (2.5)$$

2.4 Προσαρμοστικοί αλγόριθμοι υπολογισμού χρόνου αναπαραγωγής

Σε αυτό το κεφάλαιο θα παρουσιαστούν και θα συγκριθούν οι πιο γνωστοί αλγόριθμοι ρύθμισης του χρόνου αναπαραγωγής κάθε πακέτου. Στην συνέχεια αφού εντοπιστούν οι αδυναμίες τους θα προταθεί ένας καινούργιος. Για αυτόν τον σκοπό, αναπτύχθηκε ένας προσομοιωτής ο οποίος αναλύει τα «ίχνη» (traces) που παράχθηκαν από μια εφαρμογή μετάδοσης φωνής μέσω του διαδικτύου, ονόματι NeVoT – (Network Voice Terminal) [10]. Ως γλώσσα προγραμματισμού με χρήση της οποίας υλοποιήθηκε ο προσομοιωτής επιλέχθηκε η Java [11]. Πιο αναλυτικά, το πρόγραμμα προσομοίωσης διαβάζει τις χρονικές σφραγίδες (timestamps), των αποστολέα και παραλήπτη, από τα πακέτα φωνής που μεταδόθηκαν από διαφορετικούς δικτυακούς τόπους. Τα ίχνη αυτά [12] είναι αντιπροσωπευτικά των διαφορετικών συνθηκών καθυστέρησης που συναντώνται στο διαδίκτυο.

Στον πίνακα 2.4.1 παρουσιάζονται οι δικτυακοί τόποι αλλά και οι χρονικές στιγμές, κατά τις οποίες συλλέχθηκαν τα παραπάνω ίχνη. Επίσης, πρέπει να επισημάνουμε ότι κατά την διεξαγωγή αυτών των μετρήσεων, χρησιμοποιήθηκε κωδικοποίηση 8 KHz PCM για το σήμα της φωνής και με χρονική μονάδα παραγωγής πακέτων τα 20ms, το οποίο ισοδυναμεί με μήκος πακέτων 160 bytes.

TRACE	ΑΠΟΣΤΟΛΕΑΣ	ΠΑΡΑΛΗΠΤΗΣ	ΧΡΟΝΟΣ ΕΝΑΡΞΗΣ	ΔΙΑΡΚΕΙΑ	MULTICAST
1	Umass	GMD Fokus	08:41pm 6/27/95(Tu)	1348 sec	No
2	Umass	GMD Fokus	09:58am 7/21/95(Fr)	1323 sec	Yes
3	Umass	GMD Fokus	11:05am 7/21/95(Fr)	1040 sec	No
4	INRIA	Umass	09:20pm 8/26/93(Th)	580 sec	No
5	UCI	INRIA	09:00pm 9/18/93(Sa)	1091 sec	No
6	Umass	Osaka University	00:35am 9/24/93(Fr)	649 sec	No

Πίνακας 2.4.1 Πληροφορίες για τα «ίχνη» που χρησιμοποιήθηκαν

Για κάθε ένα πακέτο που έχει ληφθεί, ο προσομοιωτής υπολογίζει την χρονική στιγμή αναπαραγωγής του χρησιμοποιώντας κάποιον από τους αντίστοιχους αλγόριθμους. Αν το πακέτο φθάσει πριν από αυτήν την προγραμματισμένη χρονική στιγμή, τότε τοποθετείται στον ενταμιευτή αναπαραγωγής αλλιώς θεωρείται χαμένο.

Το θέμα συγχρονισμού των ρολογιών μεταξύ των αποστολέα και παραλήπτη αποφεύχθηκε κατά την πραγματοποίηση των πειραμάτων με την έκφραση της χρονικής στιγμής της άφιξης κάθε πακέτου σαν συνάρτηση του χρόνου αποστολής του. Αυτό γίνεται εφικτό με την χρήση της *ελάχιστης καθυστέρησης* που υπολογίζεται για όλο το «ίχνος». Αυτή η ποσότητα αναπαριστά, προσεγγιστικά την καθυστέρηση διάδοσης (propagation delay) ενός πακέτου. Επίσης, η καθυστέρηση διάδοσης εκφράζει το σταθερό μέρος της καθυστέρησης που εισάγει το δίκτυο. Η ελάχιστη καθυστέρηση υπολογίζεται διαβάζοντας ολόκληρη την πληροφορία που περιέχεται στο «ίχνος» και αφαιρώντας από την χρονική σφραγίδα του παραλήπτη αυτή του αποστολέα. Κατά αυτόν τον τρόπο το μεταβλητό μέρος της καθυστέρησης, για κάθε πακέτο υπολογίζεται ως η διαφορά μεταξύ της χρονικής σφραγίδας του παραλήπτη και του αποστολέα, μείον την ελάχιστη καθυστέρηση. Ομοίως και ο χρόνος άφιξης ενός πακέτου μπορεί να εκφραστεί ως το άθροισμα μεταξύ της χρονικής σφραγίδας του αποστολέα και του μεταβλητού μέρους της καθυστέρησης.

Σε αυτό το σημείο θα αναφερθούν οι ορολογίες καθώς και οι ορισμοί των ποσοτήτων που χρησιμοποιούνται από τους αλγορίθμους υπολογισμού της καθυστέρησης αναπαραγωγής.

Ορίζουμε ως \widehat{d}_k^i την καθυστέρηση μεταξύ της δημιουργίας, του i -οστού πακέτου της k -οστής περιόδου ομιλίας, στον αποστολέα και της άφιξης του στον παραλήπτη. Δηλαδή,
$$\widehat{d}_k^i = a_k^i - t_k^i \quad (2.6)$$

Όπως προαναφέρθηκε μπορεί να μην χρειάζεται να υποθέσουμε ότι τα ρολόγια του αποστολέα και του παραλήπτη είναι συγχρονισμένα, όμως δεν πρέπει να παρεκκλίνουν.

Επίσης, η καθυστέρηση αναπαραγωγής όλων των πακέτων που ανήκουν στην ίδια περίοδο ομιλίας οφείλει να είναι η ίδια σύμφωνα, αφενός με την περιοδική φύση της δημιουργίας των πακέτων μέσα σε μια περίοδο ομιλίας στον αποστολέα και αφετέρου με την περιοδικότητα που διακρίνει την αναπαραγωγή τους στον παραλήπτη.

Έτσι δεδομένου ενός αλγορίθμου A , η καθυστέρηση αναπαραγωγής της k -οστής περιόδου ομιλίας θα συμβολίζεται ως $\widehat{p}_k(A)$ και η χρονική στιγμή που θα αναπαραχθεί το i -οστό πακέτου της k -οστής περιόδου ομιλίας ορίζεται ως:

$$p_k^i(A) = t_k^i + \widehat{p}_k(A) \quad (2.7)$$

2.4.1 Αλγόριθμος 1 (NeVoT 1.4)

Πρώτα θα ασχοληθούμε με τον πιο απλό αλγόριθμο υπολογισμού του χρόνου αναπαραγωγής [8]. Η λειτουργία του στηρίζεται στην συνεχή εκτίμηση δυο στατιστικών χαρακτηριστικών του δικτύου: της καθυστέρησης που συνάντησε ένα πακέτο φωνής και την αντίστοιχη διακύμανση της καθυστέρησης (jitter). Αναλυτικότερα, η εκτίμηση της καθυστέρησης ενός πακέτου υπολογίζεται βάση του αλγορίθμου που περιγράφεται στο RFC793 [8]. Όσον αφορά την διακύμανση της καθυστέρησης, αυτή εκτιμάται βάση της τεχνικής που προτάθηκε από τον Van Jacobson [14] και χρησιμοποιείται στον υπολογισμό της εκτίμησης του συνολικής καθυστέρησης από τους χρονικούς μετρητές επαναμετάδοσης του TCP.

Πιο συγκεκριμένα, η εκτίμηση της καθυστέρησης του i -οστού πακέτου φωνής υπολογίζεται ως εξής:

$$\widehat{u}_k^i = a * \widehat{u}_k^{i-1} + (1-a) * \widehat{d}_k^i \quad (2.8)$$

και αντίστοιχα για την διακύμανση της καθυστέρησης:

$$\widehat{v}_k^i = a * \widehat{v}_k^{i-1} + (1-a) * \left| \widehat{u}_k^i - \widehat{d}_k^i \right| \quad (2.9)$$

Οι παραπάνω εκτιμήσεις, επαναυπολογίζονται κάθε φορά που λαμβάνεται ένα πακέτο στον παραλήπτη. Όταν ξεκινάει μια καινούργια περίοδος ομιλίας, η καθυστέρηση αναπαραγωγής \hat{p}_k εκτιμάται βάση της παρακάτω σχέσης:

$$\hat{p}_k = \hat{u}_{k-1}^{n_{k-1}} + \beta * \hat{v}_{k-1}^{n_{k-1}} \quad (2.10)$$

Το β , σε αυτήν την σχέση αποτελεί έναν συντελεστή διακύμανσης ο οποίος παρέχει ένα βαθμό επιβράδυνσης στην καθυστέρηση αναπαραγωγής. Είναι φανερό ότι όσο μεγαλύτερη είναι η τιμή του συντελεστή τόσο περισσότερα πακέτα φωνής θα αναπαραχθούν λόγω της μεγαλύτερης καθυστέρησης αναπαραγωγής. Ουσιαστικά η τιμή του β , μπορεί να χρησιμοποιηθεί στον έλεγχο της αντισταθμιστικής σχέσης μεταξύ των απωλειών και καθυστερήσεων, όπως και γίνεται στην περίπτωση των προσομοιώσεων που διεξήχθησαν.

Ουσιαστικά ο αλγόριθμος αποτελεί ένα γραμμικό αναδρομικό φίλτρο, το οποίο χαρακτηρίζεται από ένα σταθμικό παράγοντα α . Στα πειράματα που διενεργήθηκαν, η τιμή του επιλέχθηκε να είναι ίση με $\alpha = 0.998002$, ώστε να συμμορφώνεται με την υλοποίηση του αλγορίθμου στην εφαρμογή μετάδοσης φωνής μέσω του διαδικτύου ονόματι NeVoT (Network Voice Terminal) και έκδοσης 1.4.

2.4.2 Αλγόριθμος 2 (NeVoT 3.28)

Αυτός ο αλγόριθμος χρησιμοποιείται στην παρούσα έκδοση της εφαρμογής NeVoT 3.28 [10]. Πιο αναλυτικά:

Ως s_i ορίζουμε το σύνολο όλων των πακέτων που αφίχθησαν στον παραλήπτη κατά την διάρκεια της περιόδου ομιλίας, που ξεκίνησε πριν την λήψη του i . Έτσι η εκτίμηση της καθυστέρησης υπολογίζεται ως:

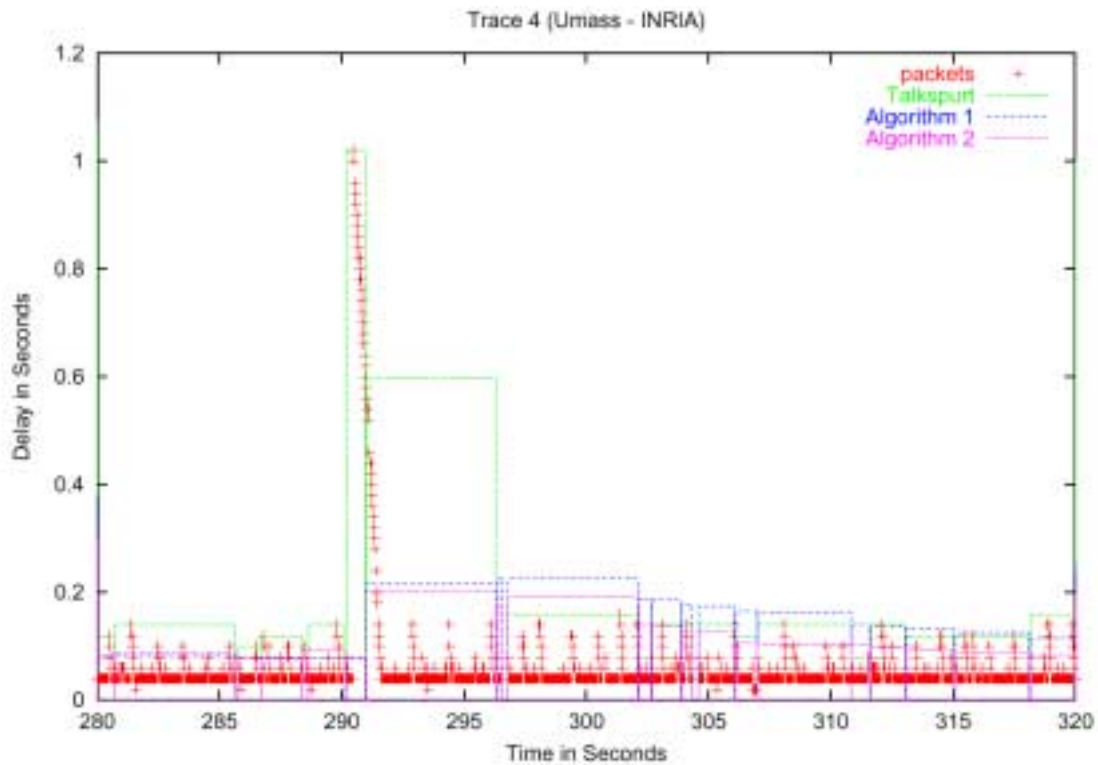
$$\hat{u}_k^i = \min_{j \in s_j} \{ \hat{d}_{k-1}^i \} \quad (2.11)$$

Τέλος ο υπολογισμός της διακύμανσης της καθυστέρησης και της καθυστέρησης αναπαραγωγής γίνεται χρησιμοποιώντας τις σχέσεις (2.9) και (2.10) αντίστοιχα.

2.4.3 Αλγόριθμος 3 (Αλγόριθμος Ανίχνευσης Στιγμιαίων Αυξήσεων στις καθυστερήσεις του δικτύου)

Όπως προαναφέρθηκε οι δυο προηγούμενοι αλγόριθμοι αποτελούν γραμμικά φίλτρα των οποίων βασικό χαρακτηριστικό είναι η μη-έγκαιρη προσαρμογή τους στις περιπτώσεις απότομων αλλαγών στην τιμή της καθυστέρησης. Όμως λειτουργούν ικανοποιητικά στις περιπτώσεις διατήρησης μιας σταθερής τιμής, ειδικά όταν ο

παράγοντας $(1-\alpha)$ του εκτιμητή τεθεί σε χαμηλή τιμή. Αυτό γίνεται εμφανές παρατηρώντας την εικόνα 2.4.1



Εικόνα 2.4.1 Εκτιμήσεις καθυστέρησης αναπαραγωγής για τους αλγόριθμους 1 & 2

Αναλυτικότερα στην εικόνα 2.4.1, εμφανίζεται η καθυστέρηση αναπαραγωγής για τους δυο προαναφερθέντες αλγόριθμους. Στον άξονα x, αναπαριστάται ο χρόνος που έχει περάσει από την αρχή της συνόδου και στον y η από άκρη-σε-άκρη καθυστέρηση που έχει υποστεί ένα πακέτο από το δίκτυο. Με το σύμβολο του σταυρού, υποδεικνύεται η τιμή αυτής της καθυστέρησης την χρονική στιγμή που το πακέτο αφίχθη στον παραλήπτη. Με το πράσινο παραλληλόγραμμα, απεικονίζονται τα όρια των περιόδων ομιλίας ενώ με τα μπλε και μωβ παραλληλόγραμμα σημειώνονται οι καθυστερήσεις αναπαραγωγής υπολογισμένες από τους αλγόριθμους 1 και 2 αντίστοιχα.

Εξαιτίας του γεγονότος της αργής προσαρμογής των προηγούμενων αλγορίθμων στις απότομες αλλαγές της τιμής της καθυστέρησης του δικτύου, προτάθηκε ένας καινούργιος [9]. Βασικό χαρακτηριστικό αυτού του αλγορίθμου αποτελεί η προσπάθεια ανίχνευσης αυτών των στιγμιαίων αυξήσεων (Spike detection Algorithm).

Το φαινόμενο αυτό παρουσιάστηκε στην παράγραφο §2.2. Η ανίχνευση της αρχής του φαινομένου είναι απλή. Το μόνο που χρειάζεται είναι να ελεγχθεί αν η καθυστέρηση μεταξύ διαδοχικών πακέτων που έφθασαν στον παραλήπτη είναι μεγαλύτερη από κάποιο όριο.

Πιο συγκεκριμένα:

$$\text{if}(\text{abs}(\widehat{d}_k^i - \widehat{d}_k^{i-1}) > \text{SPIKE_THRESHOLD}) \Rightarrow \text{Mode} = \text{SPIKE}$$

Όταν ο αλγόριθμος εισέλθει σε κατάσταση «στιγμαϊάς αύξησης» απλά την ακολουθεί και η εκτίμηση της καθυστέρησης γίνεται ως εξής:

$$\text{if}(\text{Mode} == \text{SPIKE}) \Rightarrow \widehat{u}_k^i = \widehat{u}_k^{i-1} + \widehat{d}_k^i - \widehat{d}_k^{i-1}$$

Ο εντοπισμός του τέλους του φαινομένου είναι λίγο πιο πολύπλοκος. Παρατηρώντας την εικόνα 2.2.1, γίνεται εύκολα αντιληπτό το βασικό χαρακτηριστικό που υποδηλώνει το τέλος του φαινομένου. Ουσιαστικά πρόκειται για μια σειρά από πακέτα τα οποία φθάνουν στον παραλήπτη το ένα πίσω από το άλλο σχεδόν ταυτόχρονα. Όμως επειδή τα πακέτα που ανήκουν στην ίδια περίοδο ομιλίας έχουν μεταδοθεί από τον αποστολέα με περιοδικό ρυθμό, οι σχεδόν ταυτόχρονες αφίξεις υποδηλώνουν ότι τα διαδοχικά πακέτα σε μια ομάδα αφίξεων έχουν υποστεί προοδευτικά μικρότερες από άκρη-σε-άκρη καθυστερήσεις. Κατά αυτόν τον τρόπο, κάνοντας χρήση μιας μεταβλητής (*var*) με εκθετικό ρυθμό μείωσης, εντοπίζεται το τέλος φαινομένου. Όταν η τιμή της μεταβλητής γίνει αρκετά μικρή ο αλγόριθμος επανέρχεται ξανά στην κανονική κατάσταση λειτουργίας.

Έτσι συνολικά ο αλγόριθμος παρουσιάζεται στο παρακάτω πλαίσιο.

```

If(Mode==NORMAL)
  If(abs( $\widehat{d}_k^i - \widehat{d}_k^{i-1}$ ) > abs( $\widehat{v}_k^i * 2 + 800$ )){
    Mode = SPIKE
    var = 0
  }
  else{
    var = var/2 + abs((2 *  $\widehat{d}_k^i - \widehat{d}_k^{i-1} - \widehat{d}_k^{i-2}$ )/8);
    if (var <= 63){
      Mode = NORMAL
       $\widehat{d}_k^{i-2} = \widehat{d}_k^{i-1}$ 
       $\widehat{d}_k^{i-1} = \widehat{d}_k^i$ 
    }
  }
}

```

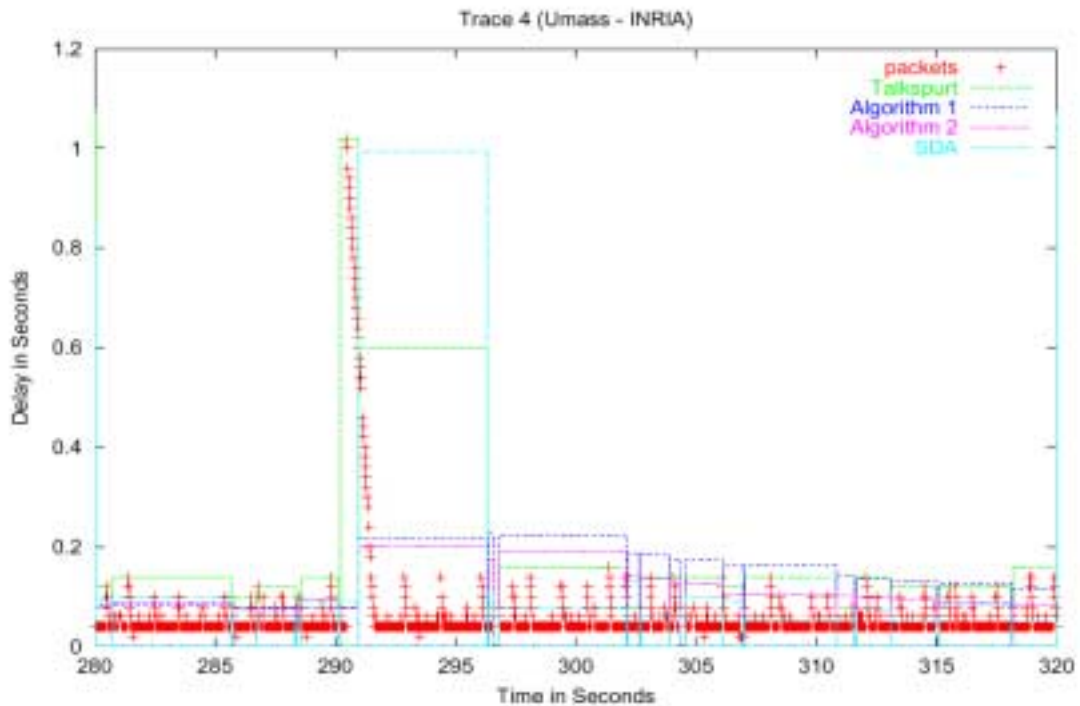
Πλαίσιο 2.4.1 Αλγόριθμος Ανίχνευσης Αιχμών (SDA)

Αντίστοιχα και η καθυστέρηση αναπαραγωγής:

$$\begin{aligned}
 & \text{If } (Mode == NORMAL) \\
 & \quad \hat{u}_k^i = \alpha * \hat{u}_k^{i-1} + (1 - \alpha) * \hat{d}_k^i \\
 & \text{else} \\
 & \quad \hat{u}_k^i = \hat{u}_k^{i-1} + \hat{d}_k^i - \hat{d}_k^{i-1} \\
 & \quad \hat{v}_k^i = \alpha * \hat{v}_k^{i-1} + (1 - \alpha) * |\hat{u}_k^i - \hat{d}_k^i| \\
 & \quad \hat{d}_k^{i-2} = \hat{d}_k^{i-1} \\
 & \quad \hat{d}_k^{i-1} = \hat{d}_k^i \\
 & \quad \hat{p}_k = \hat{u}_{k-1}^{n_{k-1}} + \beta * \hat{v}_{k-1}^{n_{k-1}}
 \end{aligned}$$

Πλαίσιο 2.4.2 Υπολογισμός καθυστέρησης αναπαραγωγής για τον SDA

Παρατηρώντας την εικόνα 2.4.2 γίνεται φανερό ότι ο αλγόριθμος ανίχνευσης «στιγμαίων αυξήσεων» (Spike Detection Algorithm) αντιδρά μεν πολύ γρήγορα και εντοπίζει την εμφάνιση των φαινομένων αυτών, όμως στην αμέσως επόμενη περίοδο ομιλίας η καθυστέρηση αναπαραγωγής έχει υποτιμηθεί με αποτέλεσμα την απώλεια αρκετών πακέτων.



Εικόνα 2.4.2 Εκτιμήσεις καθυστέρησης αναπαραγωγής για τους αλγορίθμους 1, 2, SDA

Ουσιαστικά το πρόβλημα με αυτόν τον αλγόριθμο είναι ότι προσπαθεί να ακολουθήσει τις καθυστερήσεις του δικτύου από πολύ κοντά, με αποτέλεσμα την απώλεια πακέτων όταν η εκτίμηση αυτή είναι μικρή και η περίοδος ομιλίας που ακολουθεί περιέχει πακέτα που έχουν υποστεί ελαφρώς μεγαλύτερες καθυστερήσεις.

2.4.4 Αλγόριθμος 4 (Μέθοδος Υπολογισμού «Κενών»)

Ο βασικός στόχος των προηγούμενων προσαρμοστικών αλγορίθμων [8,9] ήταν να κρατήσουν το ποσοστό απωλειών των πακέτων κάτω του 5% και να διατηρήσουν την από άκρη-σε-άκρη καθυστέρηση όσο μικρότερη γίνεται. Αυτές οι τεχνικές τυπικά αποδίδουν μέσα στα όρια που απαιτούνται από τις εφαρμογές μετάδοσης φωνής ώστε να επιτυγχάνουν ικανοποιητική ποιότητα ήχου. Όμως στερούνται της δυνατότητας να θέτουν επακριβώς ένα διαφορετικό ποσοστό ανοχής απωλειών των πακέτων έτσι ώστε να είναι σε θέση να επιτυγχάνουν διαφορετικά επίπεδα ποιότητας ήχου.

Για αυτόν το λόγο προτάθηκε ένας αλγόριθμος [15], που υπολογίζει την διαφορά μεταξύ του χρόνου αναπαραγωγής ενός πακέτου (playout time) και του χρόνου άφιξης του στον παραλήπτη. Αυτή η διαφορά ονομάζεται «κενό» (gap). Συνεχίζοντας μπορούμε να επεκτείνουμε την έννοια του «κενού» στα πακέτα που παραλήφθηκαν έγκαιρα ή καθυστερημένα [16]. Έτσι, η ποσότητα αυτή θα έχει θετική τιμή όταν ένα πακέτο φθάσει νωρίτερα από την προγραμματισμένη ώρα αναπαραγωγής και αντίστοιχα αρνητική τιμή αν παραληφθεί αργοπορημένο. Έτσι σαν «κενό» i στην περίοδο ομιλίας k ορίζουμε την ποσότητα:

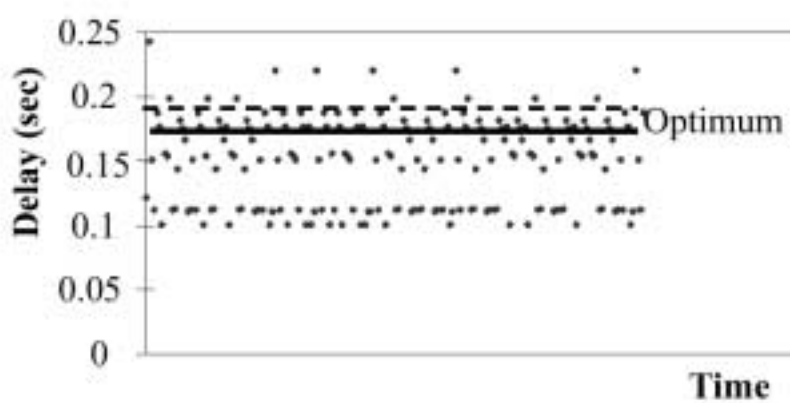
$$Gap_k^i = p_k^i - a_k^i \quad (2.12)$$

Αυτές οι ποσότητες συγκεντρώνονται από όλα τα πακέτα της περιόδου ομιλίας που μόλις ολοκληρώθηκε και χρησιμοποιούνται για την ρύθμιση του χρόνου αναπαραγωγής για την επόμενη περίοδο ομιλίας. Αυτά τα κενά χρησιμοποιούνται για την μέτρηση της απόδοσης της *τρέχουσας καθυστέρησης αναπαραγωγής* σε σχέση με την *βέλτιστη θεωρητική καθυστέρηση αναπαραγωγής*. Η πρώτη αποτελεί την πραγματική καθυστέρηση αναπαραγωγής που υπολογίζεται από τον αλγόριθμο κάθε χρονική στιγμή. Η βέλτιστη θεωρητική καθυστέρηση αναπαραγωγής αποτελεί την ποσότητα ανά περίοδο ομιλίας που ορίζεται ως η ελάχιστη επιθυμητή ποσότητα καθυστέρησης που απαιτείται, έτσι ώστε η παραγόμενη καθυστέρηση αναπαραγωγής να προκαλέσει το αναμενόμενο ποσοστό απωλειών. Όμως αυτή η θεωρητική ποσότητα δεν είναι γνωστή μέχρι να τελειώσει η περίοδος ομιλίας ώστε να υπολογιστούν όλες οι τιμές των κενών.

Ανάλογα, το *βέλτιστο κενό* ορίζεται ως η διαφορά μεταξύ της τρέχουσας και της βέλτιστης καθυστέρησης αναπαραγωγής και μπορεί να χρησιμοποιηθεί για να επιταχύνει

ή να επιβραδύνει τον χρόνο αναπαραγωγής, αντισταθμίζοντας κατά αυτόν τον τρόπο το ποσοστό των λαθών στον χρόνο αναπαραγωγής της επόμενης περιόδου ομιλίας.

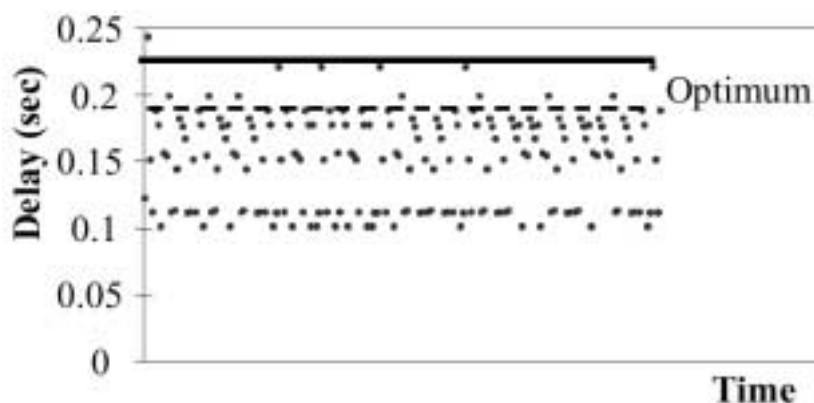
Στις δυο επόμενες εικόνες 2.4.3 και 2.4.4, παρατίθενται οι ποσότητες του βέλτιστου κενού και βέλτιστης καθυστέρησης αναπαραγωγής για δυο υποθετικές περιόδους ομιλίας. Οι τελείες στα σχήματα αναπαριστούν τις καθυστερήσεις που έχουν συναντήσει τα παραληφθέντα πακέτα, η λεπτή διακεκομμένη γραμμή αναφέρεται στην *βέλτιστη καθυστέρησης αναπαραγωγής* και τέλος η συνεχόμενη σκούρα γραμμή ορίζει την *τρέχουσα καθυστέρηση αναπαραγωγής* η οποία έχει υπολογιστεί από τον παρόντα αλγόριθμο. Πακέτα των οποίων οι καθυστερήσεις βρίσκονται πάνω από την σκούρα γραμμή θεωρούνται χαμένα και αντίστοιχα τα πακέτα με καθυστερήσεις κάτω από την σκούρα γραμμή αναπαράγονται κανονικά.



Εικόνα 2.4.3 Βέλτιστο «κενό» για ένα πιο αργό παίξιμο

Στην εικόνα 2.4.3 η γραμμή της βέλτιστης καθυστέρησης βρίσκεται πάνω από την τρέχουσα καθυστέρηση, το οποίο σημαίνει ότι η καθυστέρηση αναπαραγωγής πρέπει να επιβραδυνθεί ώστε να διατηρηθεί το επιθυμητό ποσοστό απωλειών.

Από την άλλη μεριά, στην εικόνα 2.4.4 ο προσαρμοστικός αλγόριθμος έχει υπερεκτιμήσει την καθυστέρηση αναπαραγωγής, η οποία πρέπει να εξισορροπηθεί στην αμέσως επόμενη περίοδο ομιλίας.



Εικόνα 2.4.4 Βέλτιστο «κενό» για ένα πιο γρήγορο παίξιμο

Ο αλγόριθμος που ρυθμίζει την χρονική στιγμή αναπαραγωγής ενός πακέτου, χρησιμοποιώντας τις παραπάνω ποσότητες λειτουργεί σε δυο καταστάσεις: *Κανονική* (NORMAL) και «*Στιγμαία Αύξησης*» (SPIKE). Η τεχνική ανίχνευσης των φαινομένων αυτών (Spike Detection Algorithm) η οποία περιγράφηκε στην προηγούμενη παράγραφο (§2.4.3) χρησιμοποιείται και σε αυτή την περίπτωση για τον εντοπισμό τους. Όταν ανιχνευθεί μια τέτοια στιγμιαία αύξηση στην τιμή της καθυστέρησης, ο αλγόριθμος εισέρχεται σε κατάσταση *Spike*, και ο SDA αναλαμβάνει τον υπολογισμό της καθυστέρησης αναπαραγωγής των πακέτων. Μόλις τελειώσει το φαινόμενο, ο αλγόριθμος επανέρχεται στην κανονική κατάσταση και η διαδικασία υπολογισμού των καθυστερήσεων αναπαραγωγής ξαναρχίζει. Δεδομένου ενός συνόλου ταξινομημένων καθυστερήσεων του δικτύου για την περίοδο ομιλίας $k-1$, το *βέλτιστο κενό* για την περίοδο ομιλίας k υπολογίζεται ως εξής:

$$Opt_{gap} = Gap\left(\sum_{i=0}^{(1-toler)n_{k-1}} freq_i\right) \quad (2.13)$$

Όπου, *toler* είναι η ανοχή στις απώλειες πακέτων που καθορίζεται από τον χρήστη δίνοντας μια τιμή εύρους 0 έως 1. Με *toler* = 0, δηλώνεται μηδενική ανοχή στις απώλειες ενώ με *toler* = 1 δηλώνεται ότι το 100% των απωλειών γίνεται ανεκτό. Ο συνολικός αριθμός των πακέτων που πρέπει να αναπαραχθούν σε μια περίοδο ομιλίας έτσι ώστε να διατηρηθεί το επιθυμητό ποσοστό απωλειών είναι $(1 - toler) * n_{k-1}$.

Στην σχέση (2.13), *Gap()* είναι ο τελεστής που ανακτά την τιμή του «κενού» σχετικά με την εγγραφή i του συνόλου και τέλος $freq_i$ είναι ο αριθμός των πακέτων που σημείωσαν την ίδια καθυστέρηση στο δίκτυο και άρα την ίδια τιμή κενού. Οι εξισώσεις παρακάτω παρουσιάζουν την λειτουργία του αλγορίθμου στην κανονική κατάσταση:

$$\hat{p}_k = \bar{d}_k^i + p \quad (2.14)$$

$$p = p + Opt_{gap}^{k-1} \quad (2.15)$$

Η σχέση (2.14) υπολογίζει την χρονική στιγμή αναπαραγωγής του πρώτου πακέτου στην k -οστή περίοδο ομιλίας, χρησιμοποιώντας την μέση καθυστέρηση \bar{d}_k^i κατά την κανονική κατάσταση λειτουργίας και τον ελάχιστο χρόνο ενταμίευσης p , που χρειάζεται για να τον υπολογισμό του επιθυμητού χρόνου αναπαραγωγής. Αυτός ο χρόνος ενταμίευσης ενημερώνεται με την χρήση του βέλτιστου «κενού» της προηγούμενης περιόδου ομιλίας, έτσι ώστε να ρυθμιστεί ο χρόνος αναπαραγωγής (βλ. σχέση 2.15).

Στο παρακάτω πλαίσιο παρουσιάζεται ο ψευδοκώδικας του αλγορίθμου:

```

Mode = SDA()
If (Mode == SPIKE)
     $\hat{p}_k = \hat{u}_{k-1}^{n_{k-1}} + \beta * \hat{v}_{k-1}^{n_{k-1}}$ 
else
    if (new Talksupt) {
        Calculate  $Opt_{gap}^{k-1}$ 
         $p = p + Opt_{gap}^{k-1}$ 
         $\hat{p}_k = \bar{d}_k^i + p$ 
    }
Record  $Gap_k^i = p_k^i - a_k^i$ 
Update  $\bar{d}_k^i$ 

```

Πλαίσιο 2.4.3 Αλγόριθμος υπολογισμού κενών (Gap)

Συνεχίζοντας θα παρουσιάσουμε τον ψευδοκώδικα υπολογισμού του βέλτιστου «κενού» κάνοντας χρήση μιας δομής δεδομένων που περιέχει, αφενός τις καθυστερήσεις του δικτύου και αφετέρου τα αντίστοιχα υπολογισμένα κενά των πακέτων για την περίοδο ομιλίας που μόλις έχει τελειώσει. DelayLst[x] είναι ο πίνακας των δεδομένων για τα πακέτα της περιόδου ομιλίας που μόλις τελείωσε. Στην γραμμή 2 υπολογίζεται ο αριθμός των πακέτων που πρέπει να αναπαραχθεί στην παρούσα περίοδο ομιλίας ώστε να ικανοποιείται η απαιτούμενη ανοχή σε απώλειες. Στην γραμμή 3, διατρέχεται η ταξινομημένη λίστα των καθυστερήσεων. Οι μεγάλες θετικές τιμές των κενών βρίσκονται στην αρχή της λίστας αφού αντιστοιχούν σε πακέτα που παραλήφθηκαν νωρίς. Όσο διατρέχουμε την λίστα των καθυστερήσεων οι τιμές των κενών μειώνονται μέχρι η καθυστέρηση αναπαραγωγής να γίνει κατά προσέγγιση ίση με την τρέχουσα καθυστέρηση στην λίστα. Αυτό σηματοδοτεί την έναρξη αρνητικών τιμών των κενών στην λίστα, γεγονός που αντιστοιχεί σε αργοπορημένα πακέτα. Οι γραμμές 4,5 και 6 μετρούν τον αριθμό των πακέτων που έχουν συμπεριληφθεί διατρέχοντας την λίστα μέχρι τώρα, έως ότου να φθάσει στο επιθυμητό αριθμό. Έτσι, το βέλτιστο κενό της περιόδου που μόλις τελείωσε είναι η τιμή του κενού στην καταχώρηση i της λίστας, όπου και ικανοποιείται ο βαθμός ανοχής των απωλειών (γραμμή 5).

```

CalcOptimumGap(){
  1. nCurPkts = 0

  /*calculate # of packets that should be played in order to
  meet the tolerance required */
  2. nReqPkts = (1-toler)*nk-1
  /*Loop through the list until reaching # packets required.
  The i-th gap where such condition is met is the optimum
  gap */
  3. for(i=0;i<# in DelayLst[];i++)
  4.   if(nCurPkts >= nReqPkts)
  5.     Optgapk-1 = DelayLst[i].gap
  6.   else
  7.     nCurPkts += DelayLst[i].freq
}

```

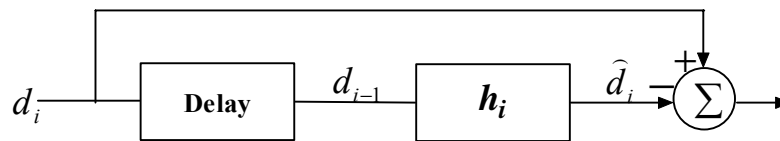
Πλαίσιο 2.4.4 Υπολογισμός Βέλτιστου Κενού για τον αλγόριθμο Gap

2.4.5 Αλγόριθμος 5 (NLMS)

Βασικό χαρακτηριστικό αυτού του αλγορίθμου [17], αποτελεί η προσπάθεια βελτίωσης της εκτίμησης της καθυστέρησης του αλγορίθμου 1, που περιγράφηκε στην παράγραφο §2.4.1. Πιο αναλυτικά, στην προκειμένη περίπτωση η ρύθμιση της καθυστέρησης αναπαραγωγής βασίζεται στην πρόβλεψη της καθυστέρησης που υπεισέρχεται από το δίκτυο, ενώ η μέτρηση της διακύμανσης της καθυστέρησης γίνεται όπως και πρώτα. Φιλοσοφία του αλγορίθμου αποτελεί η άποψη ότι η χρήση μιας ακριβούς πρόβλεψης σε σχέση με την χρήση στατιστικής εκτίμησης, αποδίδει καλύτερα ειδικά στις περιπτώσεις απότομων αλλαγών στην τιμή της καθυστέρησης.

Είναι φανερό ότι η κατάλληλη καθυστέρηση αναπαραγωγής μπορεί να οδηγήσει σε μικρότερες συνολικές από άκρη-σε-άκρη καθυστερήσεις ή σε μικρότερα ποσοστά απωλειών πακέτων, παράγοντες αρκετά σημαντικοί για εφαρμογές μετάδοσης φωνής μέσου διαδικτύου.

Η πρόβλεψη της καθυστέρησης επιτυγχάνεται με την χρήση ενός απλού προσαρμοστικού φίλτρου (Normalized Least Mean Square) [18]. Γενικά η χρήση των προσαρμοστικών φίλτρων στον τομέα των τηλεπικοινωνιών είναι αρκετά διαδεδομένη, όπως για παράδειγμα: *equalizing*, *echo cancellation* και *prediction*. Σε αυτές τις εφαρμογές ο προσαρμοστικός αλγόριθμος προσπαθεί να ελαχιστοποιήσει το αναμενόμενο τετραγωνικό λάθος, μεταξύ των πραγματικών δεδομένων d_i και της αντίστοιχης εκτίμησης \hat{d}_i , ρυθμίζοντας τις παραμέτρους ενός φίλτρου πεπερασμένης απόκρισης (Finite Impulse Response filter) h_i , το οποίο χρησιμοποιείται παράλληλα με το υπόλοιπο σύστημα. Στην εικόνα 2.4.5 παρουσιάζεται ένα τέτοιο προσαρμοστικό φίλτρο ενός βήματος.



Εικόνα 2.4.5 Προσαρμοστικό φίλτρο ενός βήματος

Ένα από τα πιο απλά προσαρμοστικά φίλτρα είναι το Normalized Least Mean Square:

$$h_{i+1} = h_i + \frac{\tilde{\mu}}{d_i^T d_i + \alpha} d_i e_i \quad (2.16)$$

όπου h_i είναι ένας $N \times 1$ πίνακας των συντελεστών του προσαρμοστικού φίλτρου, $\tilde{\mu}$ είναι το μέγεθος του βήματος, d_i είναι ο $N \times 1$ πίνακας που περιέχει τις πιο πρόσφατες N καθυστερήσεις του δικτύου, T είναι ο τελεστής ανάστροφης ενός πίνακα και e_i είναι το λάθος εκτίμησης το οποίο δίνεται από την σχέση:

$$e_i = h_i^T d_i - d_i \quad (2.17)$$

Η εφαρμογή του προσαρμοστικού φίλτρου NLMS στον υπολογισμό της καθυστέρησης αναπαραγωγής \hat{p}_k , πραγματοποιείται εφαρμόζοντας την παρακάτω διαδικασία για κάθε πακέτο:

1. Υπολογισμός της πρόβλεψης της καθυστέρησης του δικτύου \hat{d}_i , δεδομένου N προηγούμενων καθυστερήσεων με χρήση του αλγορίθμου NLMS.
2. Αυτοαναδρομικά υπολογισμός της διακύμανσης της καθυστέρησης όπως στον αλγόριθμο 1 (βλ Σχέση 2.9)
3. Υπολογισμός της καθυστέρησης αναπαραγωγής σύμφωνα με την σχέση 2.10, $\hat{p}_k = \hat{d}_i + \beta * \hat{v}_{k-1}^{n_{k-1}}$

Οι παράμετροι του φίλτρου που χρησιμοποιήθηκαν για τις προσομοιώσεις παρουσιάζονται στον παρακάτω πίνακα:

Παράμετροι	Τιμές
h_0	$[1 \ 0 \ \dots \ 0]^T$
N	20
$\tilde{\mu}$	0.01

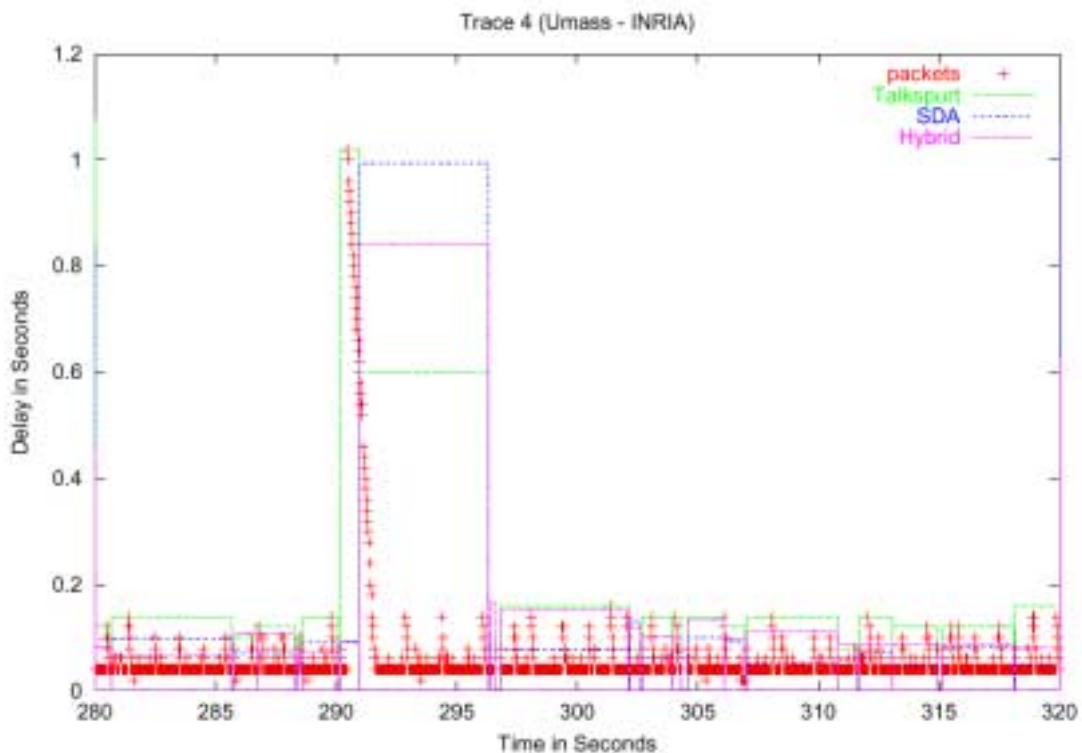
Πίνακας 2.4.2 Παράμετροι του προσαρμοστικού αλγορίθμου

2.4.6 Αλγόριθμος ανίχνευσης «στιγμαίων αυξήσεων» με παράλληλη χρήση πρόβλεψης (Hybrid)

Σε αυτή την παράγραφο, παρουσιάζεται ένας καινούργιος αλγόριθμος που ουσιαστικά αποτελεί συνδυασμό των αλγορίθμων NLMS (§2.4.5) και SDA (§2.4.3). Όπως παρατηρήθηκε ο αλγόριθμος ανίχνευσης στιγμαίων αυξήσεων στις καθυστερήσεις του δικτύου (Spike Detection Algorithm), μπορεί μεν να εντοπίζει τα φαινόμενα αυτά αλλά η εκτίμηση της καθυστέρησης αναπαραγωγής της αμέσως επομένης περιόδου ομιλίας ήταν υποτιμημένη με αποτέλεσμα την αύξηση του ποσοστού απωλειών. Ουσιαστικά το πρόβλημα με αυτόν τον αλγόριθμο ήταν η προσπάθεια του να ακολουθήσει τις καθυστερήσεις του δικτύου από πολύ κοντά, με αποτέλεσμα την αυξημένη απώλεια πακέτων όταν η εκτίμηση αυτή είναι μικρή και η περίοδος ομιλίας, που ακολουθεί το φαινόμενο της στιγμαίας αύξησης, περιέχει πακέτα που έχουν υποστεί ελαφρώς μεγαλύτερες καθυστερήσεις.

Αυτό το πρόβλημα προσπαθήσαμε να λύσουμε με την βοήθεια ενός προσαρμοστικού φίλτρου για την κατάσταση κανονικής λειτουργία. Με αυτόν τον τρόπο πετυχαίνουμε καλύτερη εκτίμηση της καθυστέρησης αποφεύγοντας έτσι τις υπερβολικές απώλειες πακέτων.

Στην εικόνα 2.4.6 παρακολουθούμε αυτήν ακριβώς την διόρθωση του προβλήματος. Στις μετρήσεις αυτός ο αλγόριθμος θα παρουσιάζεται σαν Hybrid, εξαιτίας του γεγονότος ότι αποτελεί συνδυασμό δυο υπαρχόντων αλγορίθμων.



Εικόνα 2.4.6 Συμπεριφορά του SDA και του υβριδικού

Τέλος στο παρακάτω πλαίσιο παρουσιάζεται και ο αντίστοιχος ψευτοκώδικας.

```
Mode=SDA;  
If (Mode=NORMAL)  
    Playout_delay = NLMS;  
else  
    Playout_delay = SDA;
```

Πλαίσιο 2.4.5 Αλγόριθμος Hybrid (NLMS & SDA)

Βασική αιτία της απόδοσης αυτού του αλγορίθμου, αποτελεί το γεγονός ότι σε κανονική κατάσταση η καθυστέρηση του δικτύου παρουσιάζει διακυμάνσεις μικρού εύρους, επιτρέποντας στον αλγόριθμο NLMS να «ακολουθεί» αυτές τις τιμές με αρκετά μεγάλη ακρίβεια. Από την άλλη μεριά, όταν παρουσιαστεί μια κατάσταση στιγμιαίας αύξησης (*SPIKE*) τόσο η διάρκεια της όσο και οι καθυστερήσεις του δικτύου είναι απρόβλεπτες, οπότε απλά ο αλγόριθμος την ακολουθεί έως ότου να τελειώσει.

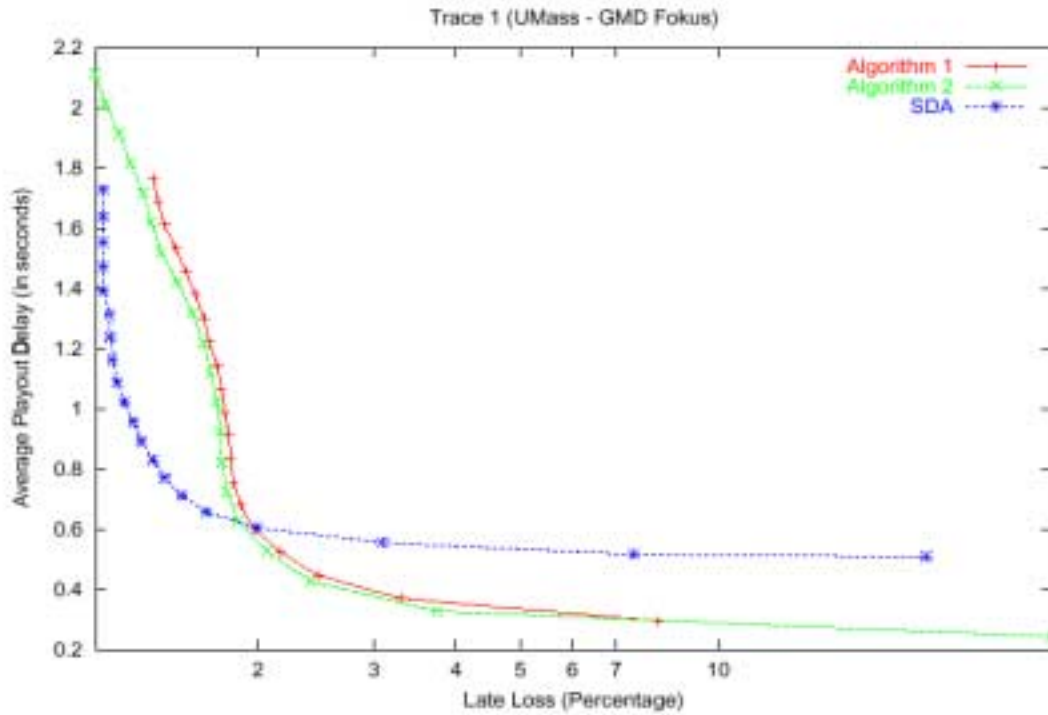
2.4.7 Πειραματικά αποτελέσματα

Για την πραγματοποίηση της σύγκρισης των προαναφερθέντων αλγορίθμων, αναπτύχθηκε ένας παραμετροποιημένος προσομοιωτής. Η γλώσσα προγραμματισμού που επιλέχθηκε για την υλοποίηση του είναι η Java [11]. Πιο συγκεκριμένα, ο χρήστης ως παραμέτρους επιλέγει αφενός έναν από τους παραπάνω αλγορίθμους και αφετέρου το «ίχνος» στο οποίο θέλει να εξετάσει την απόδοσή του. Η αξιολόγηση της απόδοσης των αλγορίθμων, πραγματοποιήθηκε χρησιμοποιώντας την αντισταθμιστική σχέση μεταξύ της μέσης καθυστέρησης αναπαραγωγής και των απωλειών λόγω αργοπορημένων αφίξεων.

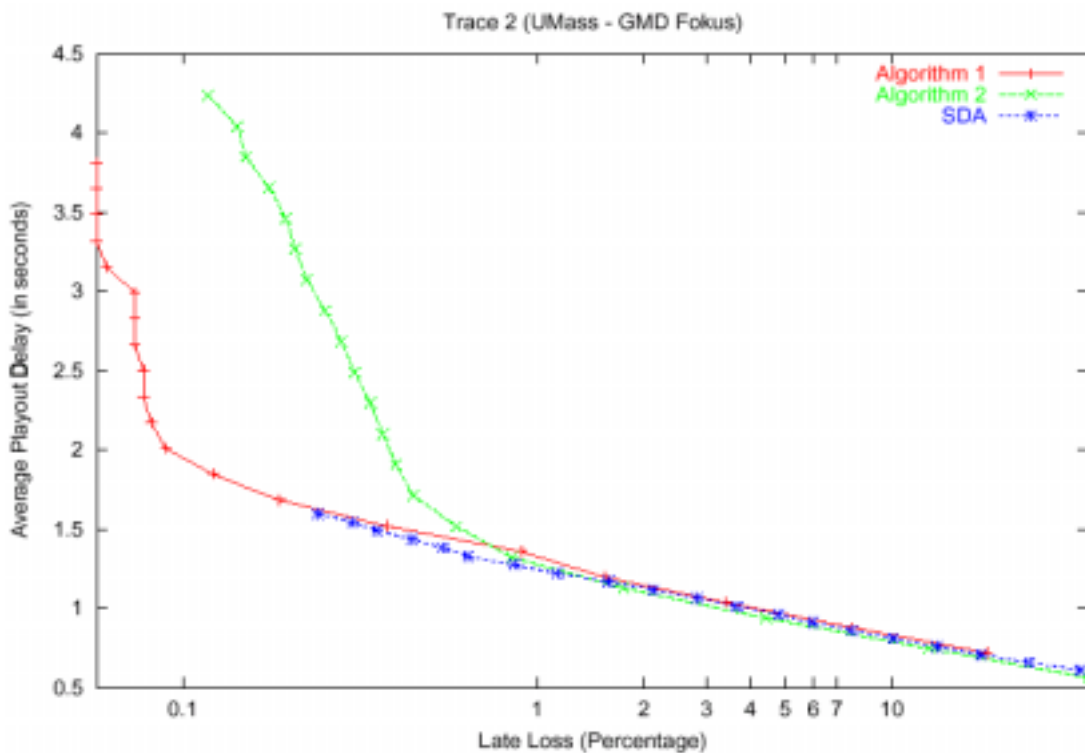
Αναλυτικότερα, όσον αφορά τους αλγορίθμους 1, 2, SDA, NLMS και Hybrid η αξιολόγηση έγινε υπολογίζοντας την καθυστέρηση αναπαραγωγής κυμαίνοντας την τιμή του β από 1 έως 20 στην σχέση 2.10, ενώ για τον αλγόριθμο GAP, οι αντίστοιχες μετρήσεις έλαβαν χώρα μεταβάλλοντας την τιμή της μεταβλητής (*toler*) που ορίζει την ανοχή στις απώλειες από 0% έως και 20%.

Οι μετρήσεις πραγματοποιήθηκαν σε τρία μέρη.

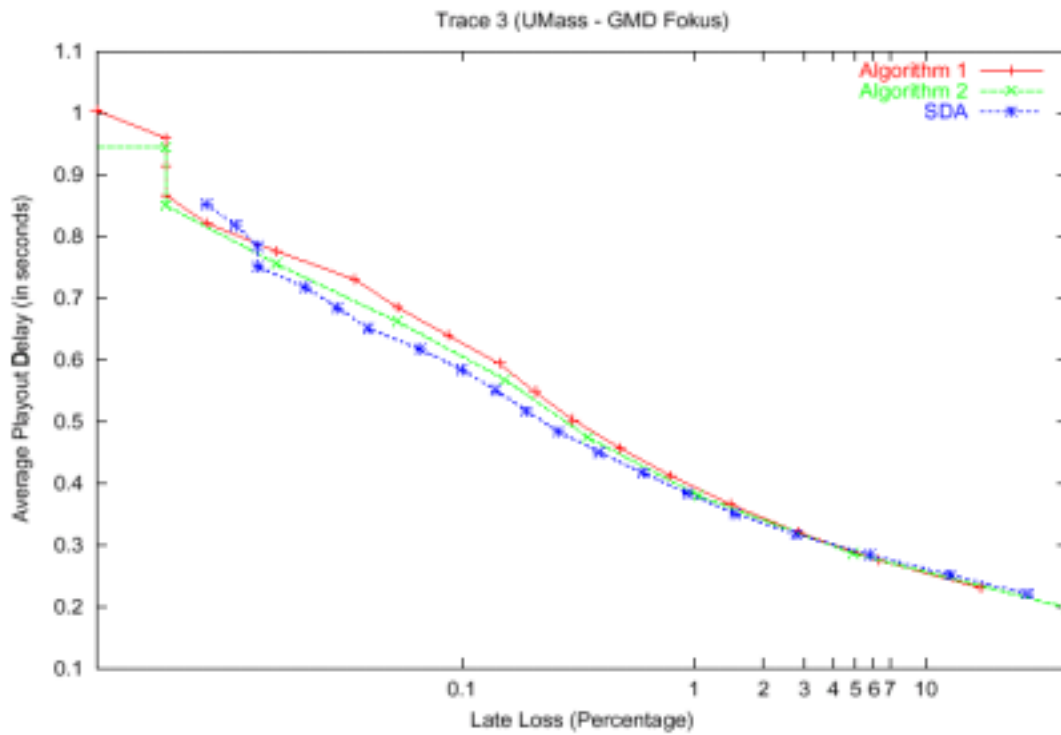
Στο πρώτο μέρος παρατίθενται οι μετρήσεις μεταξύ των αλγορίθμων 1,2 και SDA.



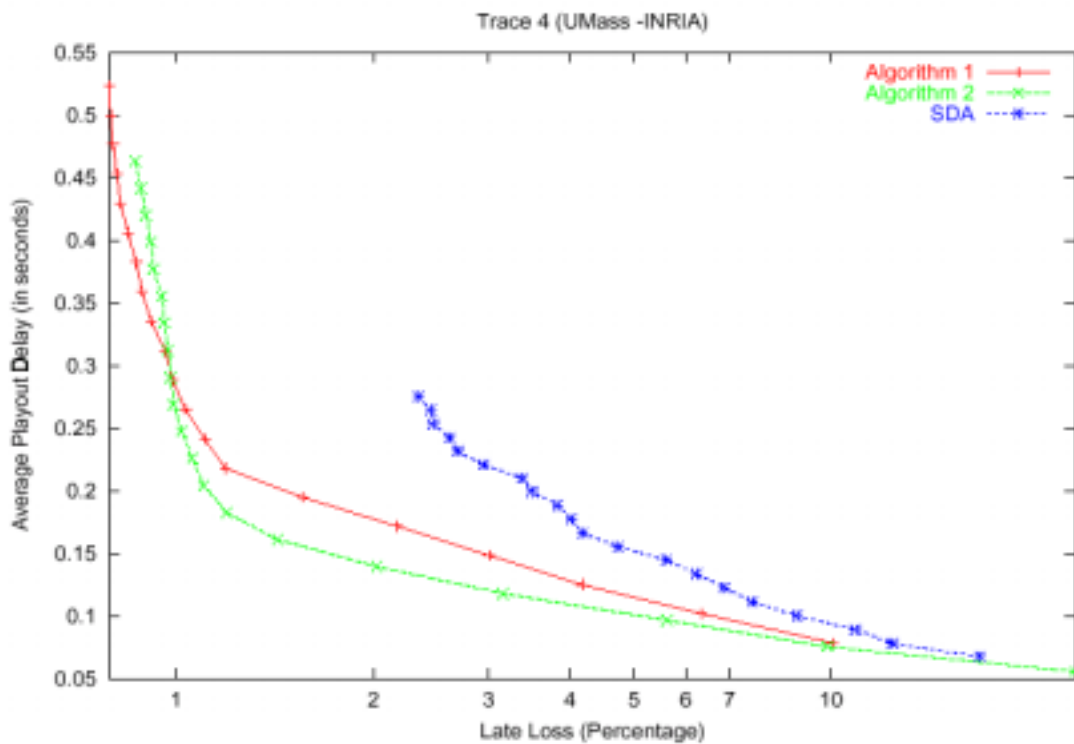
Εικόνα 2.4.7 Σύγκριση αλγορίθμων 1,2 & SDA «ίχνος 1»



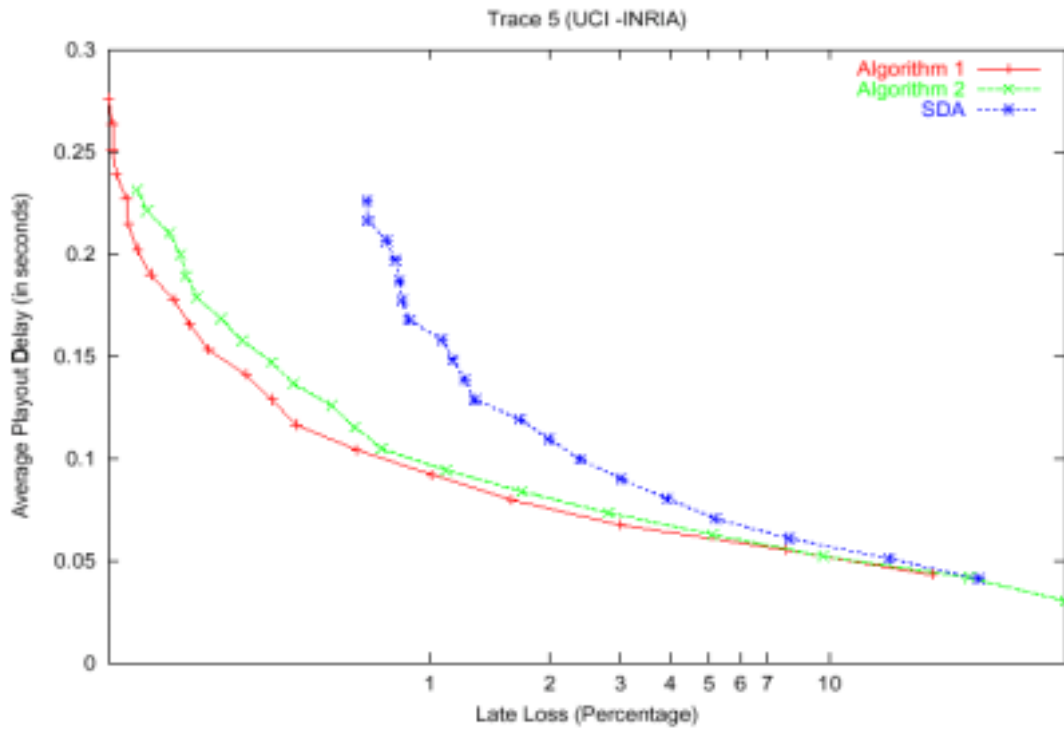
Εικόνα 2.4.8 Σύγκριση αλγορίθμων 1,2 & SDA «ίχνος 2»



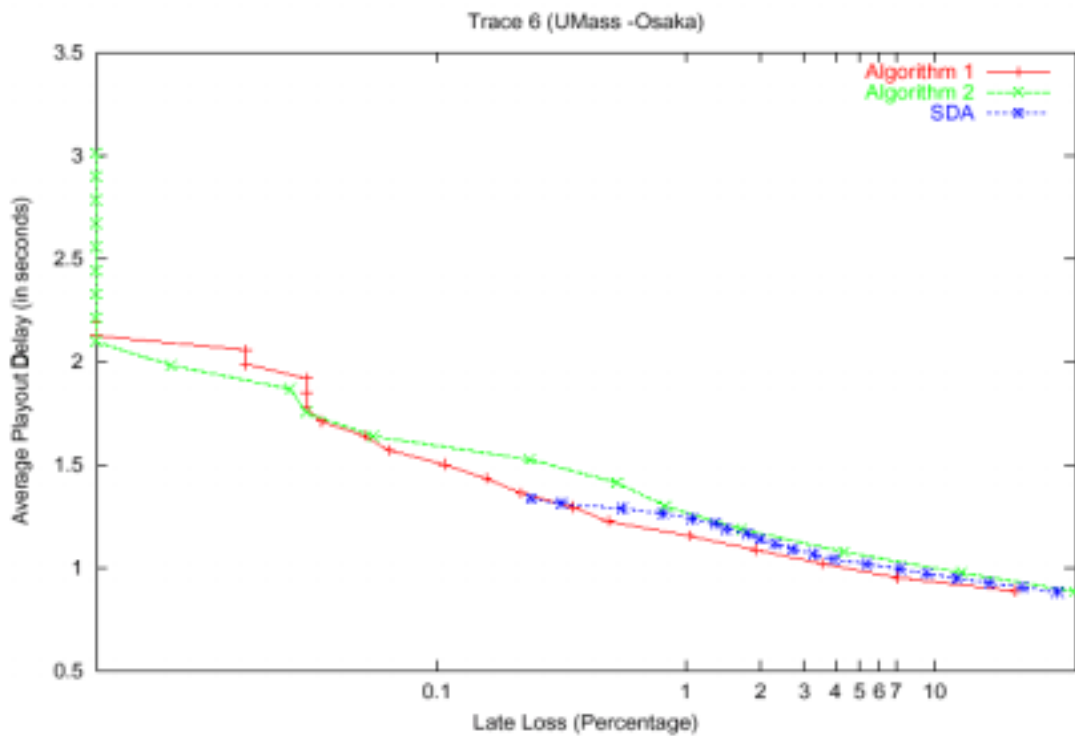
Εικόνα 2.4.9 Σύγκριση αλγορίθμων 1,2 & SDA «ίχνος 3»



Εικόνα 2.4.10 Σύγκριση αλγορίθμων 1,2 & SDA «ίχνος 4»



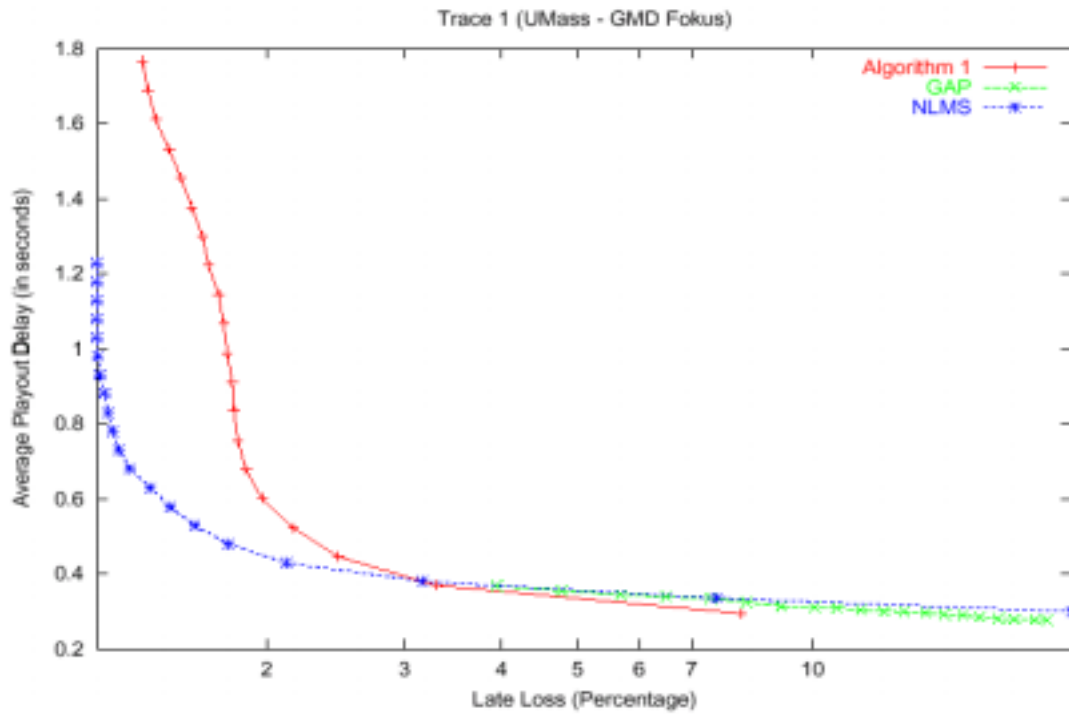
Εικόνα 2.4.11 Σύγκριση αλγορίθμων 1,2 & SDA «ίχνος 5»



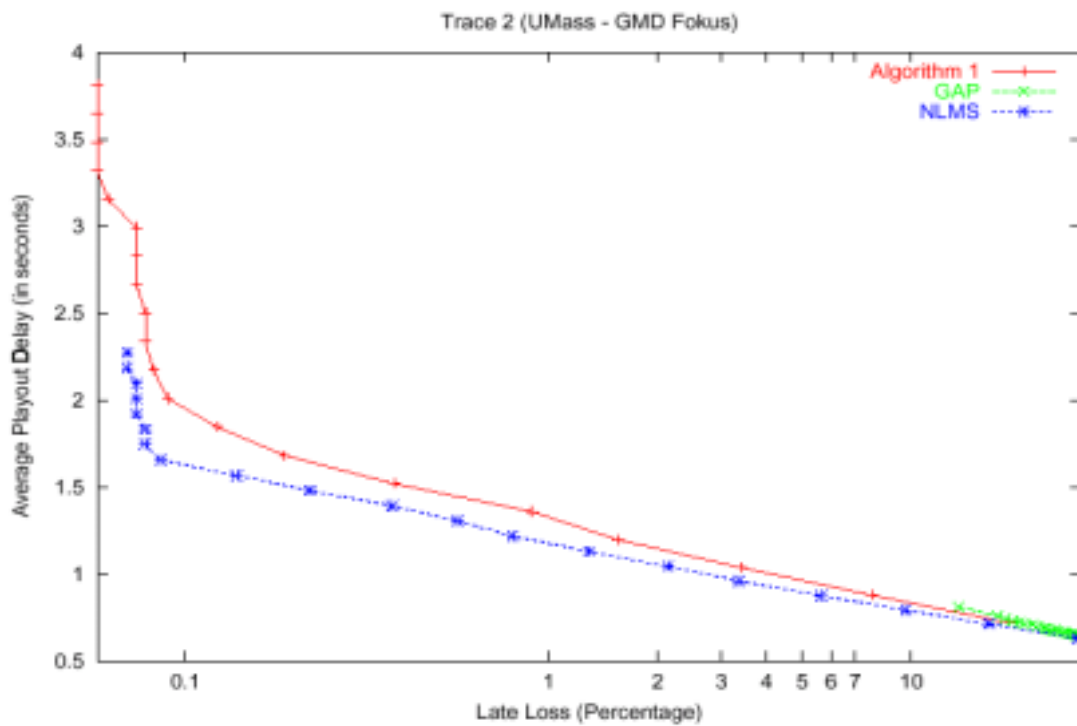
Εικόνα 2.4.12 Σύγκριση αλγορίθμων 1,2 & SDA «ίχνος 6»

Από τις εικόνες 2.4.7 έως και 2.4.12, παρατηρούμε ότι οι Αλγόριθμοι 1 και 2 αποδίδουν σχεδόν ισοδύναμα. Έτσι ουσιαστικά η επιλογή του Αλγορίθμου 2, στην τρέχουσα έκδοση της εφαρμογής μετάδοσης φωνής οφείλεται απλώς στην μικρότερη υπολογιστική πολυπλοκότητα που την χαρακτηρίζει. Από την άλλη μεριά, παρατηρούμε ότι ο αλγόριθμος ανίχνευσης των «στιγμιαίων αυξήσεων» (SDA) αποδίδει στην καλύτερη των περιπτώσεων τουλάχιστον στο ίδιο επίπεδο με τους δυο άλλους (βλ εικόνες 2.4.9 και 2.4.12). Αυτό όμως ήταν αναμενόμενο και οφείλεται στην υποτιμημένη εκτίμηση στην καθυστέρηση αναπαραγωγής της περιόδου ομιλίας που ακολουθεί μια «στιγμιαία αύξηση».

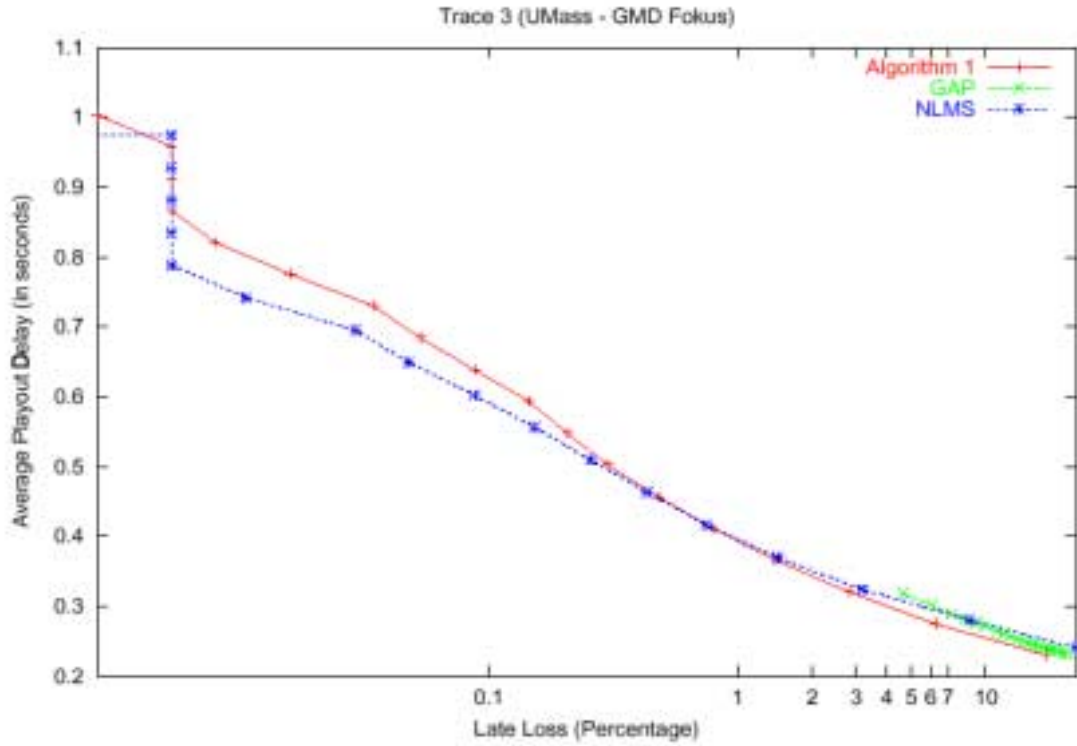
Στο δεύτερο σύνολο μετρήσεων παρατίθενται οι αλγόριθμοι GAP και NLMS, καθώς και ο Αλγόριθμος 1 ως αντιπροσωπευτικό δείγμα της πρώτης ομάδας αλγορίθμων. Παρατηρώντας τις εικόνες 2.4.13 έως και 2.4.18 γίνεται εύκολα αντιληπτή η σαφής υπεροχή του αλγορίθμου NLMS έναντι των υπολοίπων. Όμως ιδιαίτερο ενδιαφέρον παρουσιάζει και η περίπτωση του αλγορίθμου GAP. Σε αυτό το σημείο αξίζει να υπενθυμίσουμε ότι ενώ βασικό του χαρακτηριστικό αποτελούσε η παραμετροποίηση του ως προς το ποσοστό των απωλειών εντούτοις αποτυγχάνει να ανταποκριθεί σε ποσοστά απωλειών κάτω του 3%. Τέλος ούτε η απόδοση του ικανοποιεί αφού είναι το πολύ ίση με αυτή του Αλγορίθμου 1.



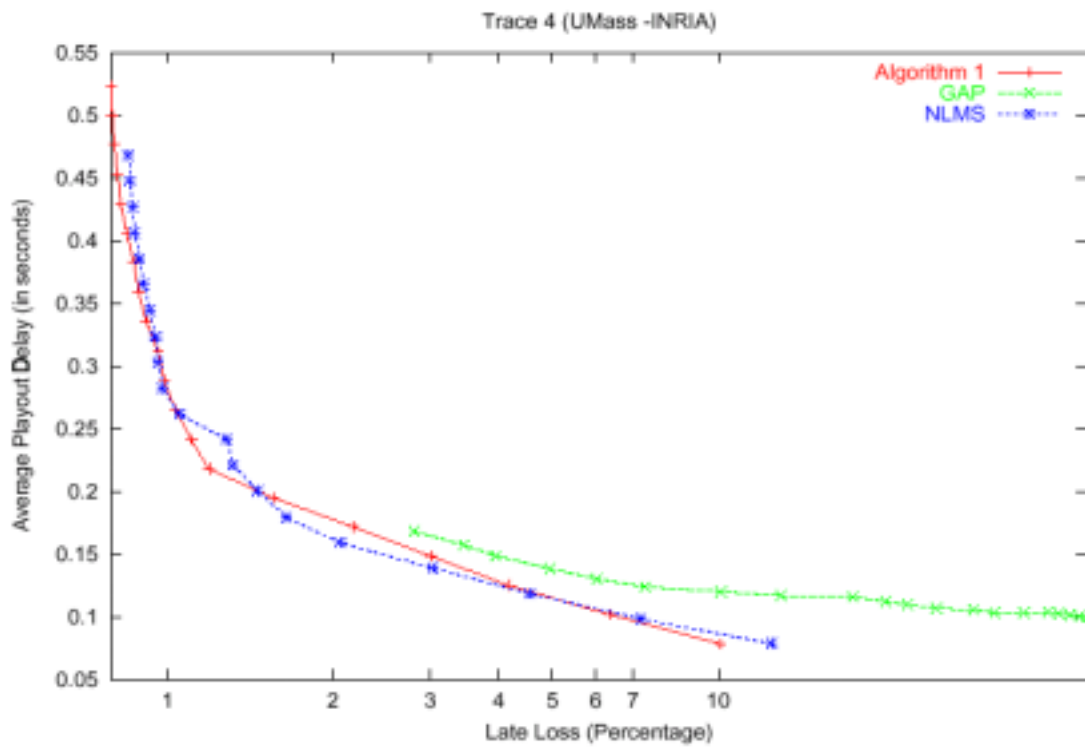
Εικόνα 2.4.13 Σύγκριση αλγορίθμων 1, GAP & NLMS «ίχνος 1»



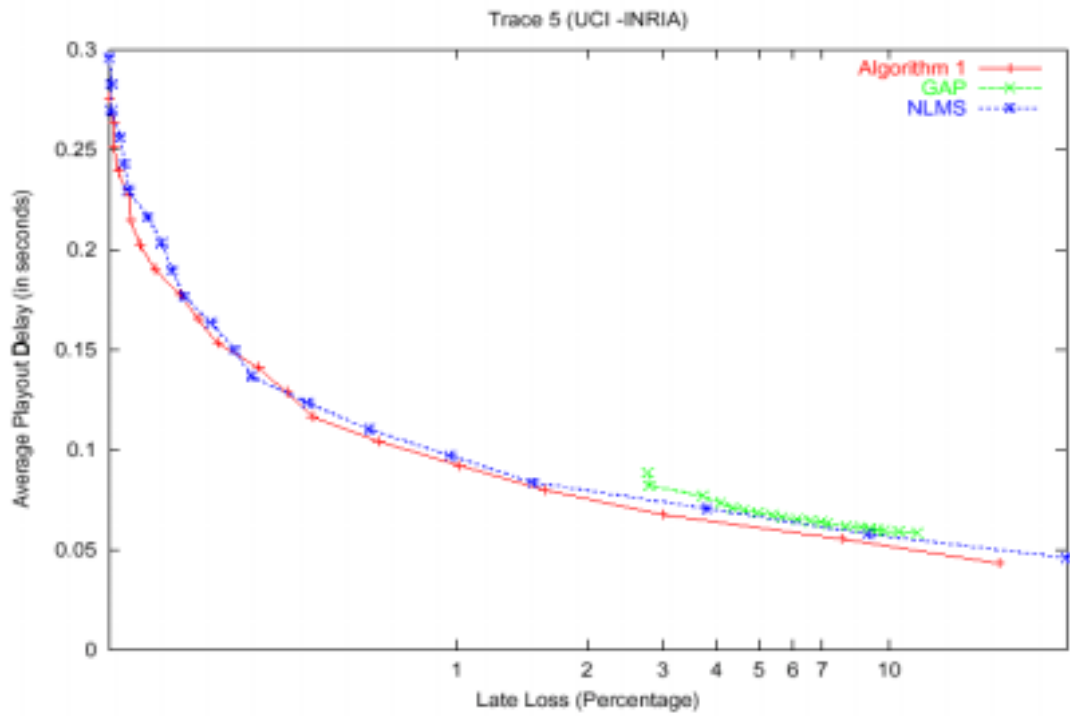
Εικόνα 2.4.14 Σύγκριση αλγορίθμων 1, GAP & NLMS «ίχνος 2»



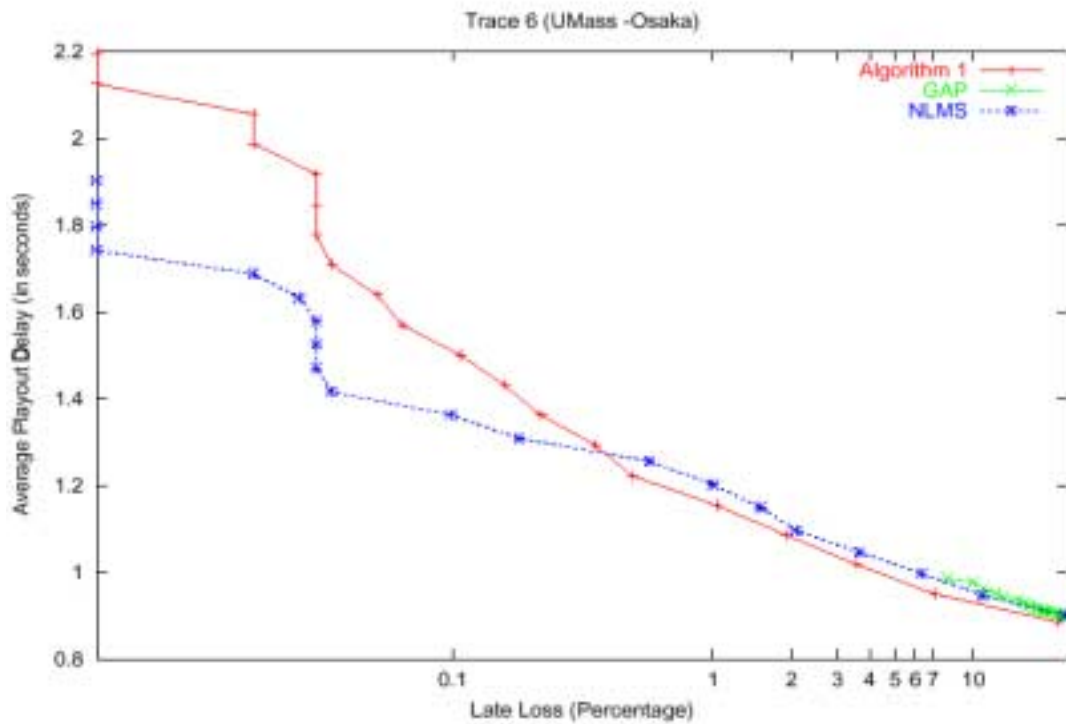
Εικόνα 2.4.15 Σύγκριση αλγορίθμων 1, GAP & NLMS «ίχνος 3»



Εικόνα 2.4.16 Σύγκριση αλγορίθμων 1, GAP & NLMS «ίχνος 4»



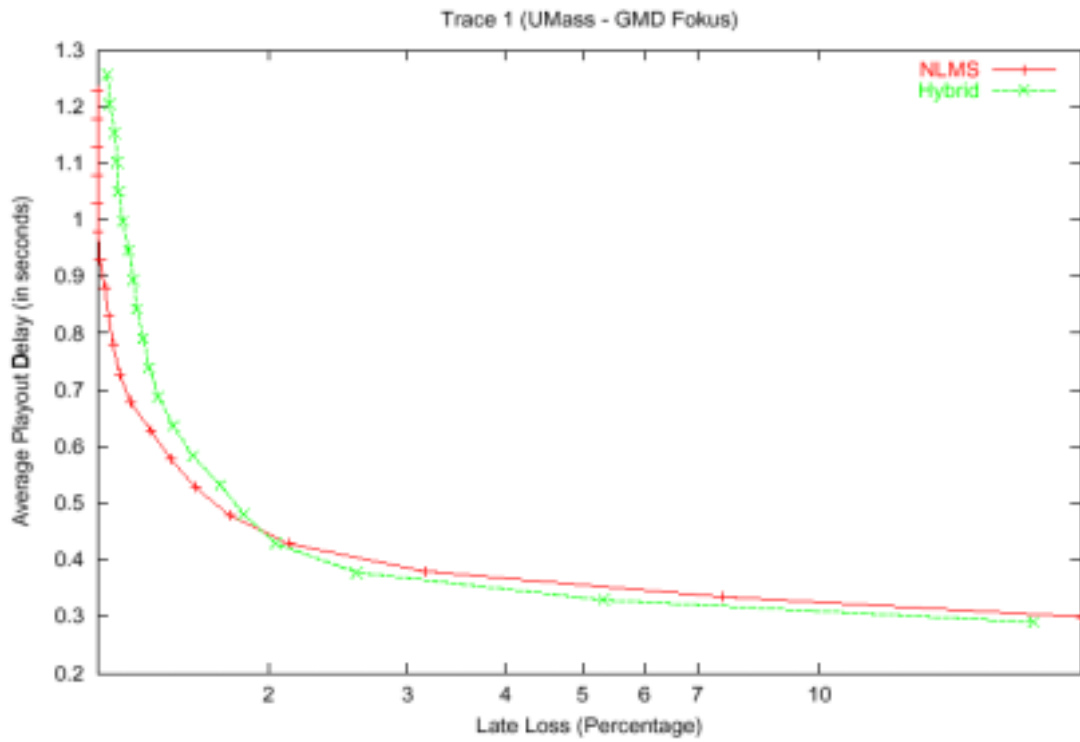
Εικόνα 2.4.17 Σύγκριση αλγορίθμων 1, GAP & NLMS «ίχνος 5»



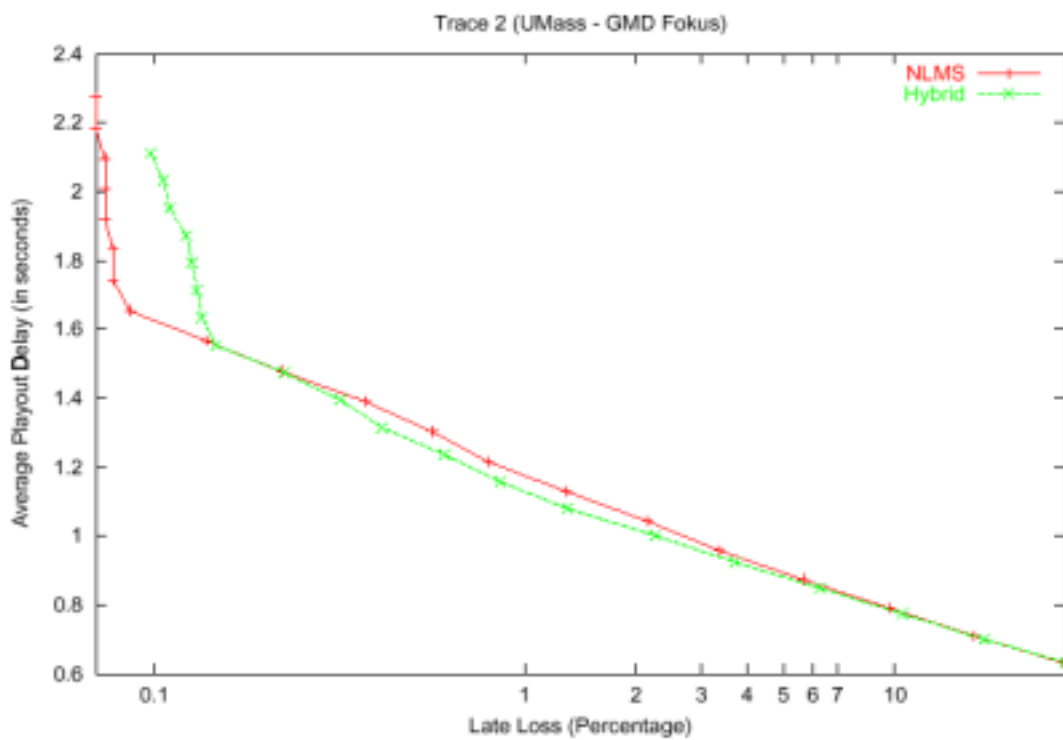
Εικόνα 2.4.18 Σύγκριση αλγορίθμων 1, GAP & NLMS «ίχνος 6»

Τέλος, το κεφάλαιο θα κλείσει με το τρίτο σύνολο μετρήσεων, στο οποίο αντιπαρατίθενται ο NLMS ως η υπάρχουσα βέλτιστη επιλογή και ο καινούργιος αλγόριθμος Hybrid που προτάθηκε μέσα από αυτήν την εργασία. Παρατηρώντας τις εικόνες 2.4.19 έως και 2.4.24 (ιδιαίτερος την εικόνα 2.4.22) γίνεται εύκολα αντιληπτή η ανωτερότητα του αλγορίθμου Hybrid έναντι του NLMS.

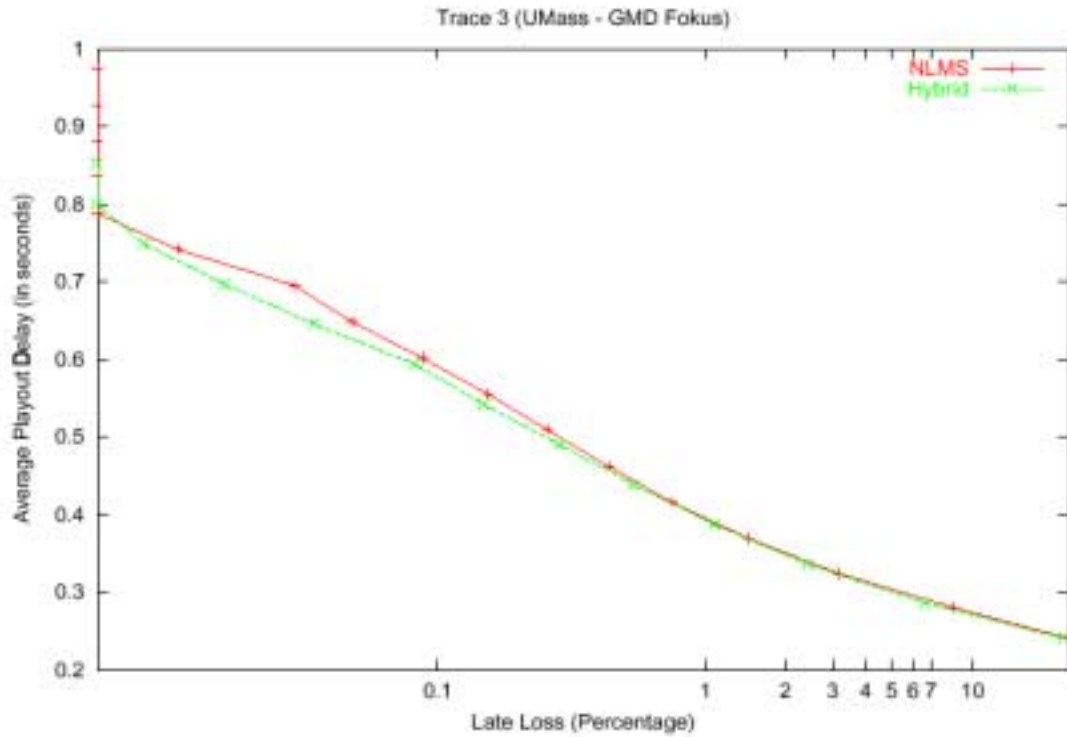
Συνοψίζοντας λοιπόν, παρατηρήσαμε ότι η χρήση ενός προσαρμοστικού φίλτρου πρόβλεψης στον υπολογισμό της καθυστέρησης αναπαραγωγής αποδίδει σαφώς καλύτερα από την αυτοαναδρομική άθροιση μέσου όρου που χρησιμοποιούν οι προηγούμενοι αλγόριθμοι. Επιπροσθέτως, εφοδιάζοντας αυτόν τον αλγόριθμο με την ικανότητα ανίχνευσης των αιχμών και με παράλληλη εφαρμογή του προσαρμοστικού φίλτρου πρόβλεψης στην κανονική κατάσταση λειτουργίας, επιτυγχάνεται καλύτερη εκτίμηση της καθυστέρησης μειώνοντας κατά αυτόν τον τρόπο το ποσοστό απωλειών των πακέτων.



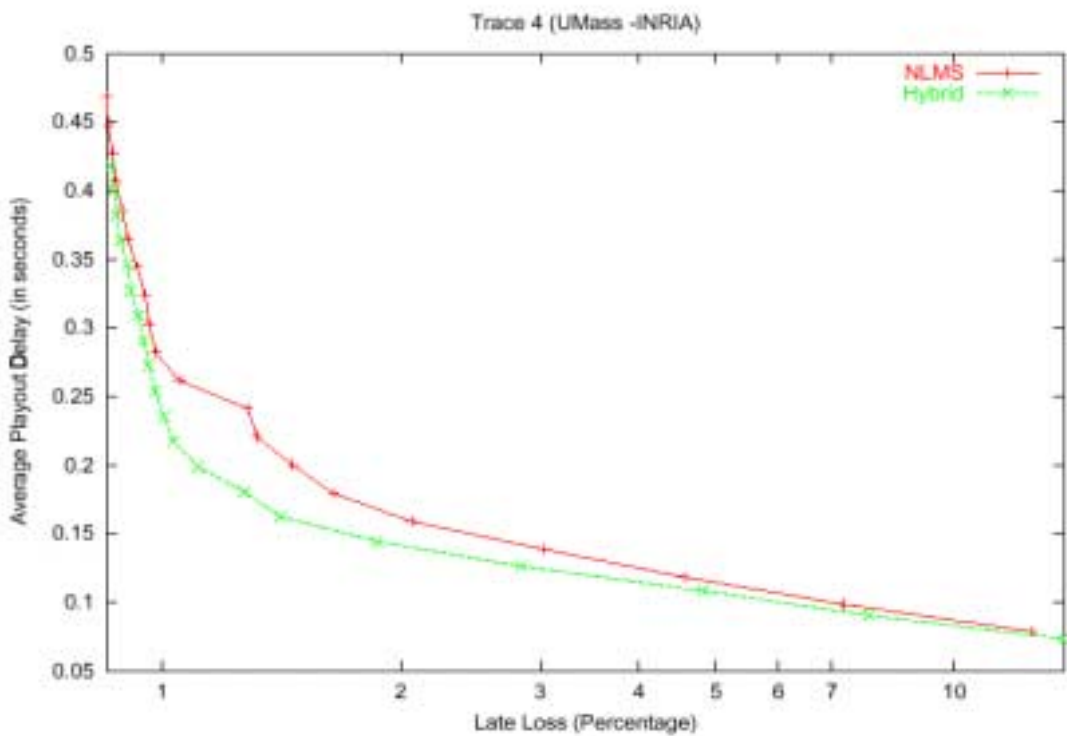
Εικόνα 2.4.19 Σύγκριση αλγορίθμων NLMS & Hybrid «ίχνος 1»



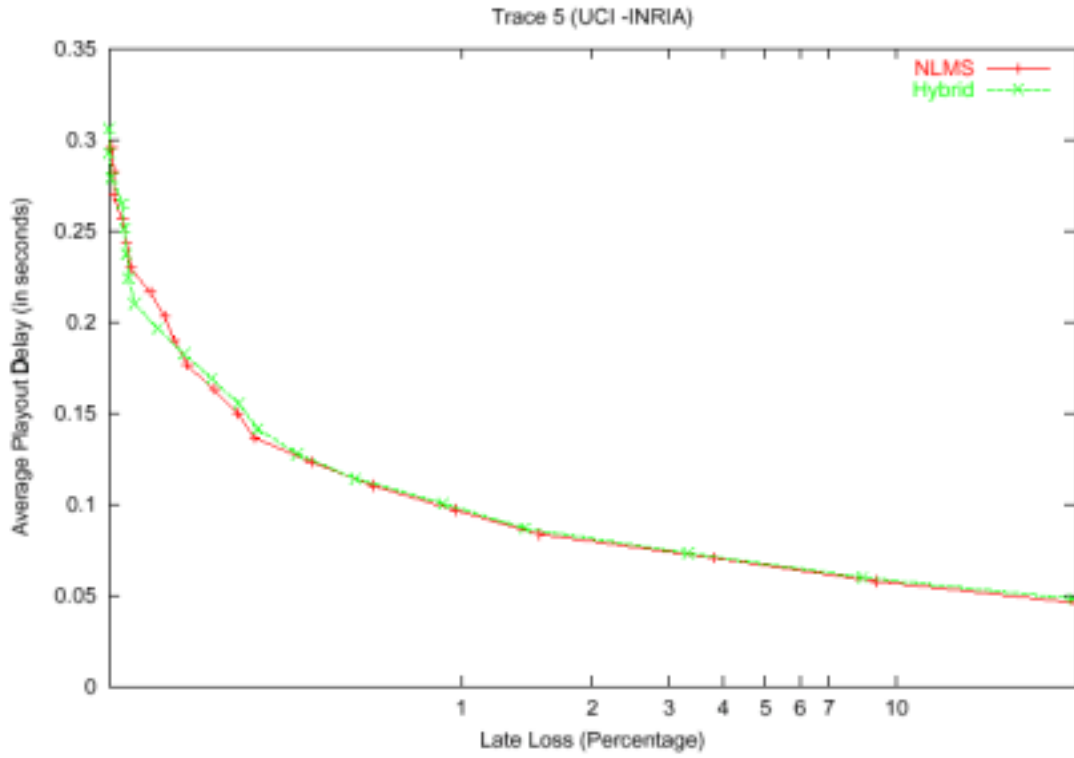
Εικόνα 2.4.20 Σύγκριση αλγορίθμων NLMS & Hybrid «ίχνος 2»



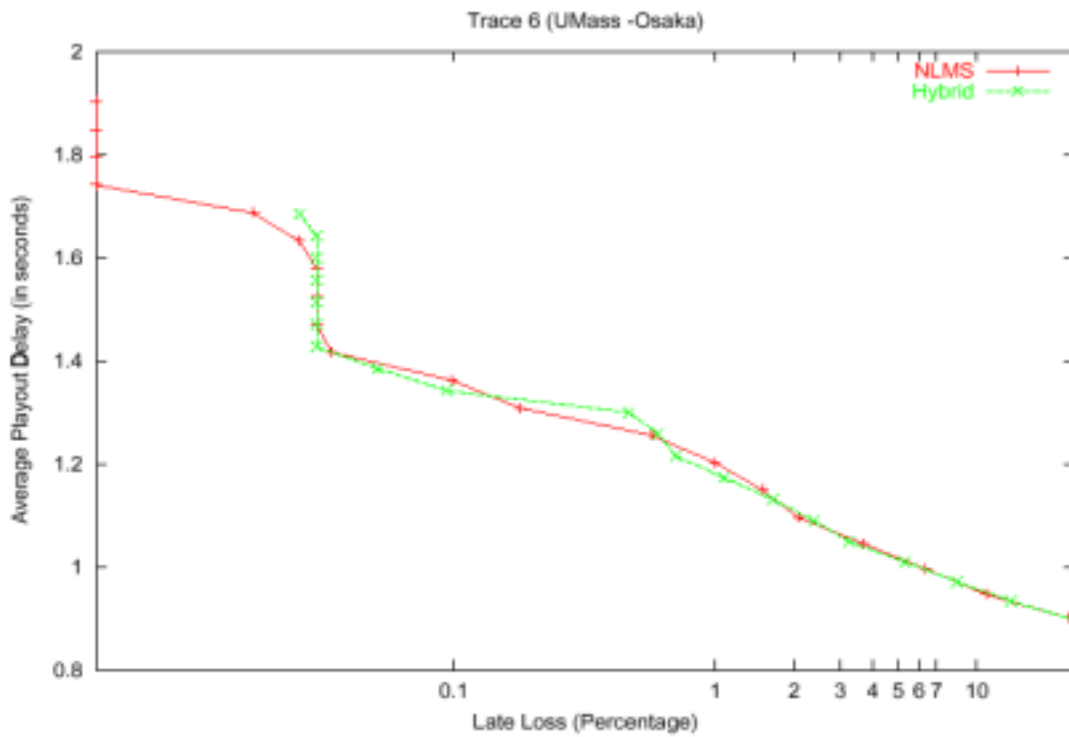
Εικόνα 2.4.21 Σύγκριση αλγορίθμων NLMS & Hybrid «ίχνος 3»



Εικόνα 2.4.22 Σύγκριση αλγορίθμων NLMS & Hybrid «ίχνος 4»



Εικόνα 2.4.23 Σύγκριση αλγορίθμων NLMS & Hybrid «ίχνος 5»



Εικόνα 2.4.24 Σύγκριση αλγορίθμων NLMS & Hybrid «ίχνος 6»

ΚΕΦΑΛΑΙΟ 3

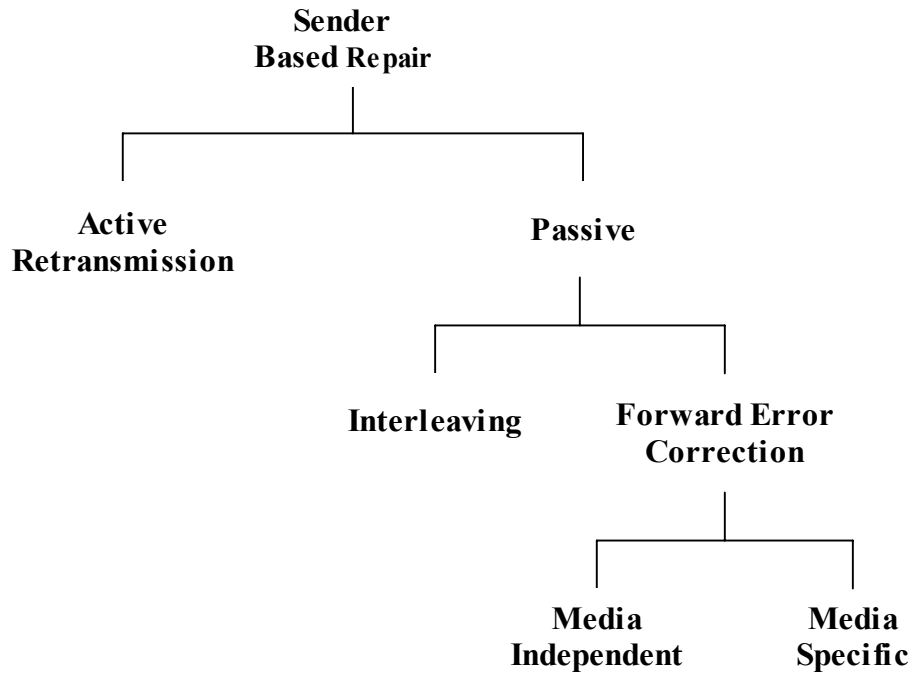
Τεχνικές ανάκτησης χαμένων πακέτων φωνής

3.1 Εισαγωγή

Είναι γνωστό ότι το επίπεδο δικτύου του Internet (IP), για την μετάδοση των πακέτων παρέχει την υπηρεσία βέλτιστης προσπάθειας (best effort). Αυτή η υπηρεσία δεν δίνει καμία εγγύηση όσον αφορά την καθυστέρηση παράδοσης αλλά και την ακεραιότητα του κάθε πακέτου. Έτσι, για να βελτιωθεί η ποιότητα μετάδοσης της φωνής πάνω από IP δίκτυα πρέπει να αναπτυχθούν τεχνικές ανάκτησης ώστε να αντιμετωπιστεί η περίπτωση απώλειας κάποιου πακέτου. Σε αυτό το σημείο πρέπει να αναφερθεί ότι η απώλεια ενός πακέτου ορίζεται με την ευρεία έννοια: «Ένα πακέτο θεωρείται χαμένο είτε αν δεν φτάσει ποτέ στον προορισμό του είτε αν φτάσει μετά την προγραμματισμένη χρονική στιγμή αναπαραγωγής». Στο προηγούμενο κεφάλαιο (βλ κεφάλαιο §2) ασχοληθήκαμε με αλγόριθμους υπολογισμού αυτής της χρονικής στιγμής, έτσι ώστε να εξαλειφθεί η διακύμανση στην καθυστέρηση (jitter) του δικτύου. Σε αυτό το κεφάλαιο αρχικά θα εξεταστούν τεχνικές ανάκτησης των χαμένων πακέτων φωνής [19], οι οποίες έχουν χρησιμοποιηθεί σε ένα ευρύ πεδίο εφαρμογών τηλεσυνδιάσκεψης. Οι τεχνικές αυτές χωρίζονται σε δυο κύριες κατηγορίες: Σε αυτές που βασίζονται στον αποστολέα και σε αυτές που βασίζονται στον παραλήπτη.

3.2 Επανόρθωση βασισμένη στον αποστολέα

Σε αυτό το σημείο θα αναφερθούν οι τεχνικές επανόρθωσης, οι οποίες απαιτούν την συμμετοχή του αποστολέα των πακέτων φωνής. Πιο συγκεκριμένα οι τεχνικές αυτές χωρίζονται σε δύο κύριες κατηγορίες: της «ενεργής επαναμετάδοσης» (active retransmission) και «παθητικής κωδικοποίησης καναλιού» (passive channel coding). Επίσης μπορούμε να υποδιαιρέσουμε ακόμη περισσότερο τις τεχνικές κωδικοποίησης καναλιού: σε τεχνικές «Forward Error Correction» (FEC) και «τεχνικές διαφύλλωσης» (interleaving). Τέλος οι FEC τεχνικές, κατηγοριοποιούνται σε μεθόδους ανεξάρτητες του μέσου (media independent), οι οποίες βασίζονται σε λογικές πράξεις Αποκλειστικού-Η (XOR) και σε μεθόδους άμεσα εξαρτώμενες από τις ιδιότητες του σήματος φωνής (media dependent). Τα παραπάνω συνοψίζονται στην εικόνα 3.2.1.



Εικόνα 3.2.1 Τεχνικές επανόρθωσης βασισμένες στον αποστολέα

3.2.1 Forward Error Correction (FEC)

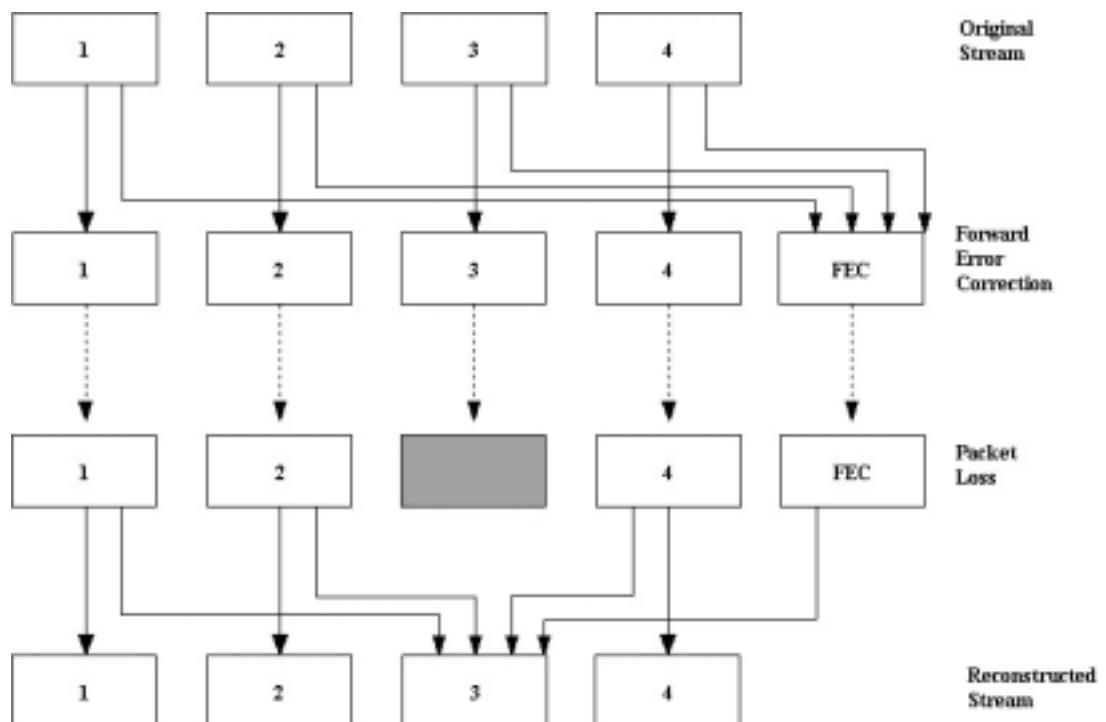
Ένας μεγάλος αριθμός από τεχνικές FEC, έχουν αναπτυχθεί με σκοπό την επιδιόρθωση λαθών κατά την μετάδοση των δεδομένων. Οι τεχνικές αυτές στηρίζονται στην προσθήκη αλλά και μετάδοση επιπλέον πληροφορίας, με την βοήθεια της οποίας είναι δυνατόν να επανακτηθούν ακριβώς ή προσεγγιστικά τα αρχικά δεδομένα, σε περίπτωση λάθους. Γενικά υπάρχουν δύο είδη επιπλέον πληροφορίας: αυτή η οποία είναι ανεξάρτητη των δεδομένων που μεταδίδονται (media independent FEC) και της πληροφορίας που εξαρτάται άμεσα από την φύση των δεδομένων που μεταδίδονται (media specific FEC), όπου στην προκειμένη περίπτωση είναι το ψηφιοποιημένο σήμα φωνής.

3.2.1.1 Media Independent FEC

Υπάρχει πληθώρα Media Independent FEC τεχνικών οι οποίες χρησιμοποιούν αλγεβρικούς (ή μπλοκ) κώδικες για την δημιουργία επιπλέον πληροφορίας η οποία χρησιμοποιείται για την διόρθωση λαθών κατά την μετάδοση. Εμείς όμως θα αναφερθούμε στα δυο πιο γνωστά αλγεβρικά σχήματα κωδικοποίησης: parity coding και Reed-Solomon coding, τα οποία έχουν προταθεί σαν RTP payload [20]. Τα παραπάνω αλγεβρικά σχήματα κωδικοποίησης είχαν αρχικά σχεδιαστεί με σκοπό τον εντοπισμό αλλά την διόρθωση των λαθών μιας ροής μεταδιδόμενων bits, το οποίο είχε σαν αποτέλεσμα τα bit ελέγχου να παράγονται από τα bit δεδομένων. Επειδή, εμείς θέλουμε να αντιμετωπίσουμε την απώλεια ενός ολόκληρου πακέτου οι κώδικες αυτοί εφαρμόζονται στα bit μιας ομάδας πακέτων. Αυτό σημαίνει ότι το i -οστό bit

στο πακέτο ελέγχου έχει παραχθεί από το κάθε i -οστό bit των αντίστοιχων πακέτων των δεδομένων.

Πιο συγκεκριμένα, η μέθοδος parity coding εφαρμόζει την λογική πράξη *Αποκλειστικό-Η* (*XOR*), σε μια ομάδα πακέτων ώστε να παραχθεί το αντίστοιχο πακέτο ισοτιμίας (parity packet). Ένα παράδειγμα αυτής της τεχνικής έχει υλοποιηθεί από τον Rosenberg [21]. Αναλυτικότερα, σε αυτή τη περίπτωση 1 πακέτο ισοτιμίας μεταδίδεται μετά από κάθε $n - 1$ πακέτα δεδομένων. Αν χάνεται ένα πακέτο ανά n που έχουν μεταδοθεί, τότε με τον παραπάνω κώδικα μπορεί να ανακτηθεί η χαμένη πληροφορία (βλ. εικόνα 3.2.2). Επίσης, αν εφαρμοστεί η λογική πράξη *Αποκλειστικό-Η* σε διαφορετικούς συνδυασμούς πακέτων, προκύπτουν αρκετά σχήματα κωδικοποίησης ισοτιμίας τα οποία και αναφέρονται στο [20].



Εικόνα 3.2.2 Επιδιόρθωση χρησιμοποιώντας parity FEC

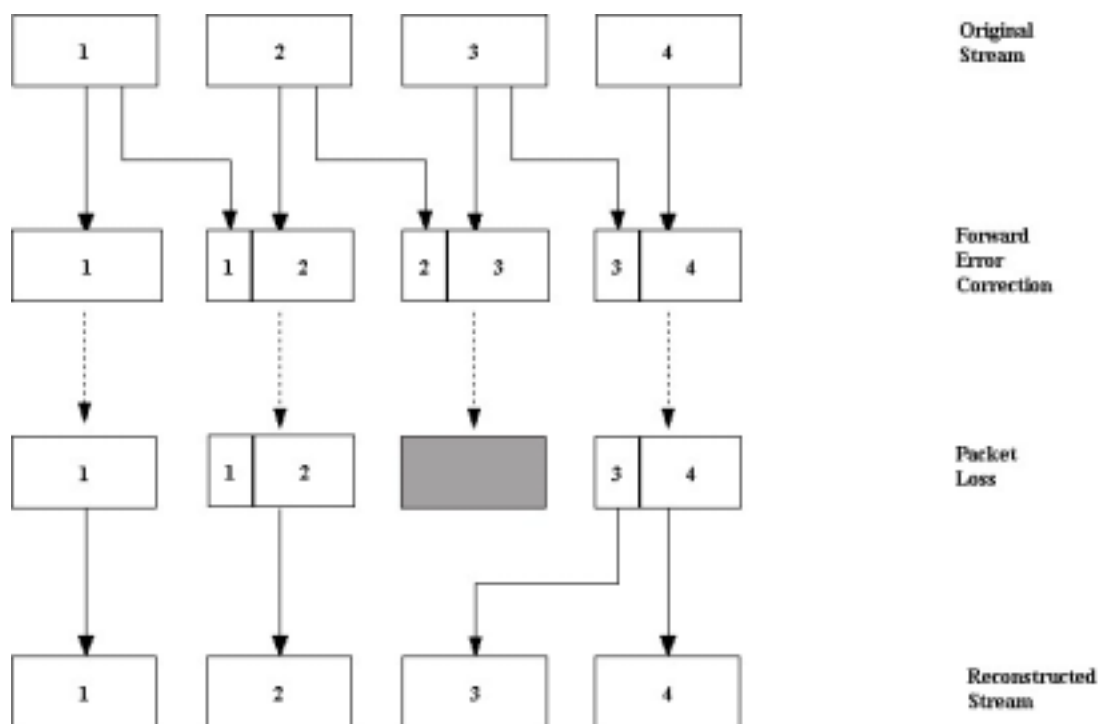
Οι τεχνικές κωδικοποίησης Reed-Solomon [22,23] φημίζονται για τις άριστες ικανότητες διόρθωσης λαθών που έχουν. Η κωδικοποίηση στηρίζεται στις ιδιότητες των πολυωνύμων σε συγκεκριμένες αριθμητικές βάσεις. Βασικά οι κωδικοποιητές παίρνουν ένα σύνολο κωδικών λέξεων και το χρησιμοποιούν σαν συντελεστές ενός πολυωνύμου $f(x)$. Η κωδική λέξη που μεταδίδεται, καθορίζεται από την αποτίμηση του πολυωνύμου για όλες τις μη-μηδενικές τιμές του x , για μια συγκεκριμένη αριθμητική βάση. Αν και φαίνεται αρκετά πολύπλοκο, η διαδικασία κωδικοποίησης είναι σχετικά απλή. Όσον αφορά το θέμα της αποκωδικοποίησης, υπάρχουν βελτιστοποιημένες διαδικασίες όπως οι αλγόριθμοι των Berlekamp και Massey [24,25]. Στην περίπτωση που δεν συμβεί κάποιο λάθος κατά την μετάδοση, η

αποκωδικοποίηση έχει το ίδιο υπολογιστικό κόστος με την κωδικοποίηση, όμως στην περίπτωση λάθους η αποκωδικοποίηση είναι πολύ πιο ακριβή.

Οι παραπάνω τεχνικές έχουν αρκετά πλεονεκτήματα. Βασικότερο αυτών, αποτελεί το γεγονός ότι είναι ανεξάρτητες από την φύση των δεδομένων που μεταφέρονται στα πακέτα. Επίσης, στην περίπτωση σφάλματος κατά την μετάδοση κάποιου πακέτου μπορούμε να αναπαράγουμε πλήρως το χαμένο πακέτο. Τέλος, το υπολογιστικό κόστος που απαιτείται για την παραγωγή του πακέτου ελέγχου, είναι σχετικά μικρό και εύκολο στην υλοποίηση. Όμως, μειονεκτήματα αυτών των μεθόδων αποτελούν η δύσκολη και πολύπλοκη υλοποίηση του αποκωδικοποιητή, η επιπλέον καθυστέρηση που υφίσταται και τέλος οι αυξημένες απαιτήσεις για bandwidth. Για παράδειγμα, αν χρησιμοποιηθεί parity coding, ο ρυθμός μετάδοσης αυξάνεται κατά ένα παράγοντα $1/n$.

3.2.1.2 Media Specific FEC

Ίσως ο πιο απλός τρόπος για την προστασία των δεδομένων, από την απώλεια κάποιου πακέτου είναι να σταλθεί η ίδια πληροφορία σε πολλά πακέτα. Κατά αυτόν τον τρόπο αν κάποιο πακέτο χαθεί, τότε επανακτάται η χαμένη πληροφορία από κάποιο άλλο πακέτο που την περιέχει. Η παραπάνω αρχή παρουσιάζεται στην εικόνα 3.2.3. Η τεχνική αυτή προτάθηκε από τον Hardman [26], και τον Bolot [27] με σκοπό να χρησιμοποιηθεί στο Mbone.



Εικόνα 3.2.3 Επιδιόρθωση χρησιμοποιώντας media specific FEC

Το πρώτο αντίγραφο φωνητικών δεδομένων που μεταδίδεται, θα αναφέρεται σαν βασική κωδικοποίηση ενώ οι επακόλουθες μεταδόσεις θα ονομάζονται δευτερεύουσες κωδικοποιήσεις. Ο αποστολέας είναι σε θέση να αποφασίσει αν οι δευτερεύουσες κωδικοποιήσεις, θα ακολουθούν το ίδιο σχήμα κωδικοποίησης με την βασική ή όχι. Συνήθως, σαν δευτερεύουσα κωδικοποίηση επιλέγεται μια χαμηλότερης ποιότητας και χαμηλότερων απαιτήσεων σε bandwidth, από την βασική. Η επιλογή του είδους των κωδικοποιήσεων αποτελεί ένα αρκετά δύσκολο πρόβλημα καθώς εξαρτάται τόσο από τις απαιτήσεις σε bandwidth, αλλά και την υπολογιστική πολυπλοκότητα της εκάστοτε κωδικοποίησης.

Μια συνήθης επιλογή, είναι η χρήση 8kHz PCM(64 Kbps) σαν βασική κωδικοποίηση και GSM(13 Kbps) σαν δευτερεύουσα. Κατά αυτόν τον τρόπο ο αποστολέας παράγει ένα πακέτο φωνητικών δεδομένων χρησιμοποιώντας την βασική κωδικοποίηση για τα αντίστοιχα δεδομένα και την δευτερεύουσα κωδικοποίηση για να επαναλάβει προηγούμενα φωνητικά δεδομένα. Έτσι στην περίπτωση απώλειας κάποιου πακέτου, ο δέκτης μπορεί να «αναπληρώσει» την απώλεια φωνητικών δεδομένων από το αμέσως επόμενο πακέτο.

Επίσης, στην περίπτωση που θέλουμε να αντιμετωπίσουμε την κατάσταση εμφάνισης συνεχόμενων λαθών (burst errors) μπορούμε να εφαρμόσουμε μια παραλλαγή του παραπάνω αλγορίθμου. Ο αποστολέας αντί στο n -οστό πακέτο να προσαρτεί μόνο τα $(n-1)$ φωνητικά δεδομένα (με χαμηλότερης ποιότητας κωδικοποίηση), θα επισυνάπτει τα $(n-1)$ και $(n-2)$ ή $(n-1)$ και $(n-3)$ κτλ. Βέβαια το μειονέκτημα σε αυτήν την περίπτωση είναι όσο αυξάνεται η επισυναπτόμενη πληροφορία τόσο μεγαλώνουν και οι απαιτήσεις σε bandwidth, αλλά και οι υπαισερχόμενες καθυστερήσεις.

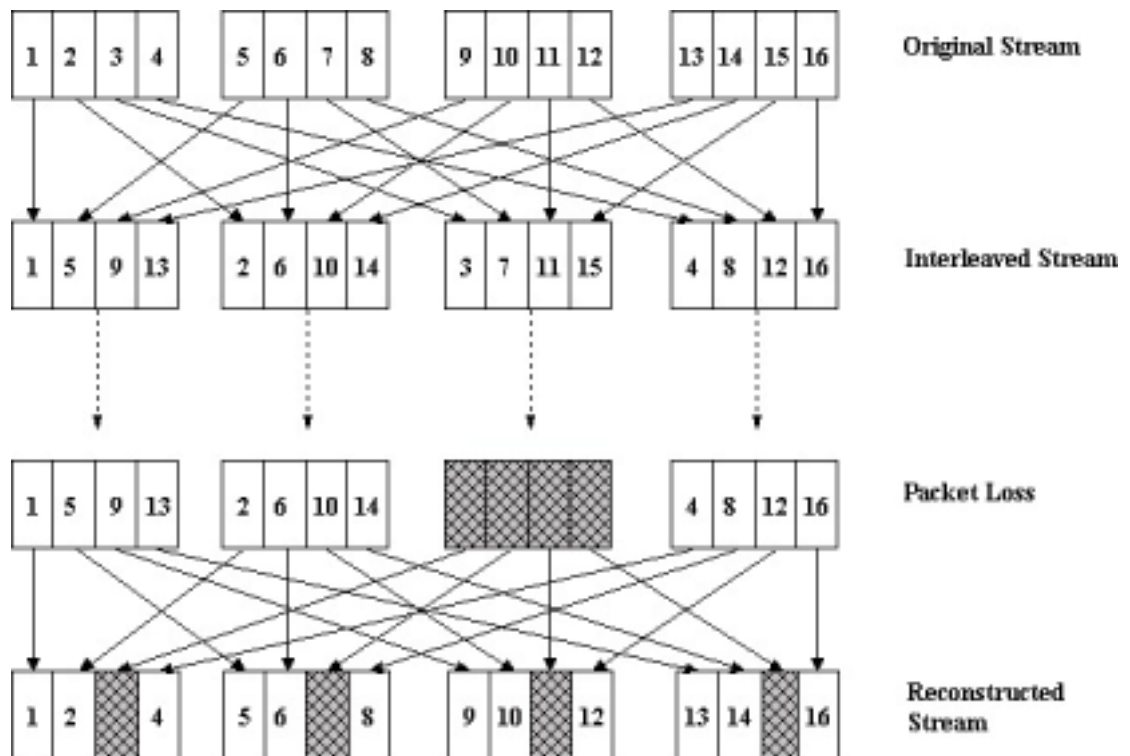
Συνοψίζοντας, μπορούμε να πούμε ότι η χρήση media-specific FEC έχει το πλεονέκτημα της χαμηλής καθυστέρησης, όταν επισυνάπτουμε μόνο τα αμέσως προηγούμενα φωνητικά δεδομένα. Αυτός είναι και ο λόγος για τον οποίο χρησιμοποιείται σε αλληλεπιδραστικές εφαρμογές πραγματικού χρόνου όπως Free – Phone [28] και Robust Audio Tool [29]. Τέλος πρέπει να επισημάνουμε το γεγονός ότι η χρήση media-specific FEC, υποστηρίζεται και από το RTP [30].

3.2.2 Διαφύλλωση

Στην περίπτωση που η μονάδα φωνητικών δεδομένων είναι μικρότερη από το τελικό μέγεθος του πακέτου και είναι ανεχτές κάποιες καθυστερήσεις, τότε μπορεί να χρησιμοποιηθεί η τεχνική της διαφύλλωσης ώστε να μειωθούν οι επιπτώσεις των λαθών κατά την μετάδοση των πακέτων [31]. Κατά την διαδικασία της διαφύλλωσης οι μονάδες φωνητικών δεδομένων επανατοποθετούνται κατά τέτοιο τρόπο έτσι ώστε οι αρχικά γειτονικές ομάδες φωνητικών δεδομένων να «απέχουν» κατά μια σταθερή απόσταση στα μεταδιδόμενα πακέτα. Ο σκοπός της διαφύλλωσης είναι να «διασκορπίσει» τις συνέπειες που θα προκαλέσει η απώλεια ενός πακέτου. Για παράδειγμα αν κάθε μονάδα φωνητικών δεδομένων έχει διάρκεια 5ms και τα πακέτα

που μεταδίδονται έχουν διάρκεια 20ms (δηλαδή 4 μονάδες ανά πακέτο), τότε το πρώτο πακέτο θα περιέχει τις μονάδες 1, 5, 9, 13, το δεύτερο πακέτο θα περιέχει αντίστοιχα τις μονάδες 2, 6, 10, 14 κτλ. (βλ. εικόνα 3.2.4)

Παρατηρούμε λοιπόν, ότι η απώλεια ενός πακέτου έχει σαν αποτέλεσμα την δημιουργία πολλών μικρών «κενών» στα ανακατασκευασμένα πακέτα στον παραλήπτη, το οποίο είναι σαφώς καλύτερο από την ύπαρξη ενός μεγάλου «κενού» στα φωνητικά δεδομένα του παραλήπτη.



Εικόνα 3.2.4 Διαφύλλωση φωνητικών μονάδων σε πολλαπλά πακέτα

Αυτή η διασπορά των συνεπειών από την απώλεια ενός πακέτου είναι πολύ σημαντική για δυο λόγους: Αρχικά, οι περισσότερες εφαρμογές μετάδοσης φωνής πάνω από IP δίκτυα μεταδίδουν πακέτα μήκους παρόμοιο με αυτό των φωνημάτων της ανθρώπινης ομιλίας. Στην περίπτωση που χαθεί ένα ολόκληρο πακέτο, θα προκύψουν σοβαρά προβλήματα στην κατανόηση της ομιλίας. Αν όμως η απώλεια «διασπαρθεί» σε πολλά μικρά κομμάτια έτσι ώστε μικρά μέρη φωνημάτων να χαθούν, γίνεται πιο εύκολο από τον δέκτη να κατανοήσει το περιεχόμενο της ομιλίας [32]. Επίσης, οι περισσότερες τεχνικές ανακατασκευής των αρχικών φωνητικών δεδομένων αποδίδουν καλύτερα όταν τα σφάλματα συμβαίνουν σε μικρά κομμάτια και διεσπαρμένα, παρά όταν ένα μεγάλο κομμάτι του αρχικού σήματος χαθεί. Τέλος κύριο πλεονέκτημα αυτής της τεχνικής αποτελεί το γεγονός ότι δεν αυξάνει τις απαιτήσεις σε bandwidth.

Όμως από την άλλη, βασικό μειονέκτημα της διαφύλλωσης αποτελεί η αύξηση της καθυστέρησης που υπεισέρχεται στον παραλήπτη, γεγονός που περιορίζει τον βαθμό χρήσης της στις εφαρμογές μετάδοσης φωνής πραγματικού χρόνου.

3.2.3 Επαναμετάδοση

Είναι γνωστό ότι οι αλληλεπιδραστικές εφαρμογές έχουν περιορισμένη ανοχή στην συνολική καθυστέρηση μετάδοσης των πακέτων φωνής (end-to-end) και για να γίνουμε πιο συγκεκριμένοι πρέπει να είναι μικρότερη των 250ms [33]. Αυτό αποτελεί και την βασική αιτία που η τεχνική της επαναμετάδοσης δεν εφαρμόζεται σαν μηχανισμός ανάκτησης των δεδομένων σε περίπτωση λάθους.

Παρόλα αυτά μια μελέτη έχει διεξαχθεί από τον Nonnenmacher [34], όπου προτείνεται ένας συνδυασμός των μεθόδων parity FEC και επαναμετάδοσης. Αναλυτικότερα, μόνο στην περίπτωση εμφάνισης κάποιου σφάλματος συμβαίνει η επαναμετάδοση του πακέτου ισοτιμίας.

3.3 Επανόρθωση βασισμένη στον παραλήπτη

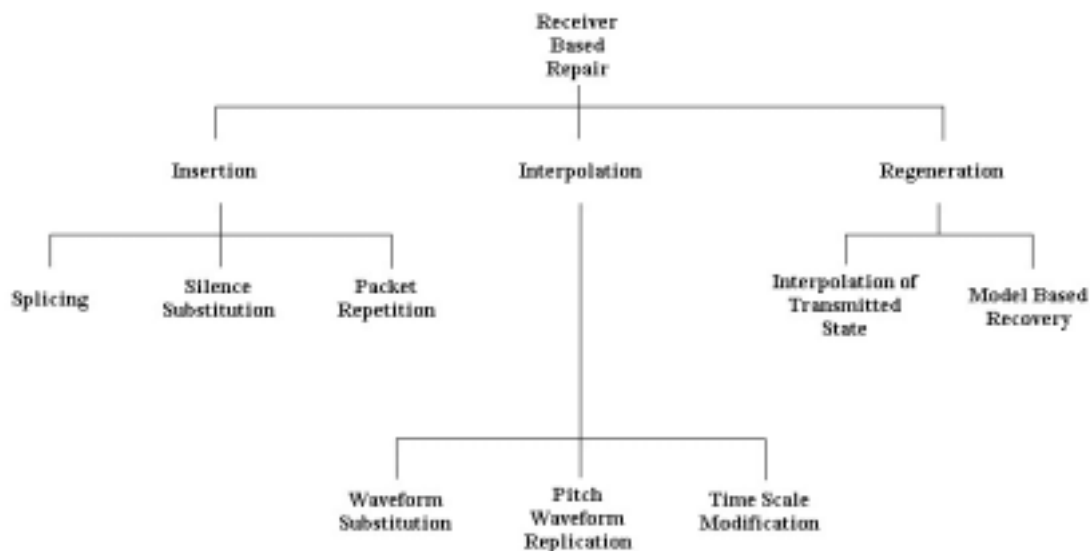
Σε αυτή την παράγραφο αναφέρεται μια σειρά από τεχνικές αντιμετώπισης σφαλμάτων, οι οποίες βασίζονται στον παραλήπτη και δεν απαιτούν την συνεργασία του αποστολέα. Αυτές οι τεχνικές χρησιμοποιούνται όταν οι τρόποι επανόρθωσης που βασίζονται στον αποστολέα αποτύχουν ή όταν ο αποστολέας δεν μπορεί να συμμετάσχει στην διαδικασία ανάκτησης των χαμένων δεδομένων.

Οι τεχνικές αυτές προσπαθούν να αναπληρώσουν κάποιο χαμένο πακέτο, δημιουργώντας ένα καινούργιο όσο πιο «πανομοιότυπο» γίνεται με το αρχικό. Αυτό είναι δυνατό γιατί τα σήματα ήχου και πιο συγκεκριμένα η ομιλία, χαρακτηρίζονται από την ιδιότητα του *sort-term self similarity*. Κατά αυτόν τον τρόπο οι τεχνικές επανόρθωσης που βασίζονται στον παραλήπτη έχουν πολύ καλή απόδοση για σχετικά μικρά ποσοστά λαθών (<15%) και για μικρού μεγέθους πακέτα (4-40ms). Όμως όταν η διάρκεια ενός σφάλματος ξεπεράσει την διάρκεια ενός φωνήματος (5-100ms) οι τεχνικές αυτές αποτυγχάνουν, γιατί ουσιαστικά χάνονται ολόκληρα κάποια φωνήματα.

Σε αυτό το σημείο πρέπει να επισημανθεί το γεγονός ότι τα σχήματα επανόρθωσης βασιζόμενα στον παραλήπτη δεν αποτελούν υποκατάστατο των αντίστοιχων μεθόδων που βασίζονται στον αποστολέα. Αντίθετα, οφείλουν να λειτουργούν παράλληλα. Πιο συγκεκριμένα, οι τεχνικές που βασίζονται στον αποστολέα πρέπει να χρησιμοποιούνται με σκοπό την επανόρθωση μεγάλων λαθών αφήνοντας την διόρθωση των μικρότερων και πιο μεμονωμένων σφαλμάτων στον παραλήπτη.

Μια ταξινόμηση, διαφόρων μεθόδων επανόρθωσης λαθών που βασίζονται στον παραλήπτη παρουσιάζεται στην εικόνα 3.3.1. Παρατηρούμε ότι οι παραπάνω τεχνικές χωρίζονται σε τρεις κατηγορίες:

- **Εισαγωγή (insertion):** Επανορθώνουν ένα σφάλμα εισάγοντας ένα *fill-in* πακέτο. Η δομή του πακέτου αυτού είναι πολύ απλή, όπως σιωπή, θόρυβος ή ακόμη και επανάληψη του τελευταίου πακέτου. Οι τεχνικές αυτές είναι πολύ απλές στην υλοποίηση και με εξαίρεση την επανάληψη του τελευταίου πακέτου έχουν πολύ χαμηλή απόδοση.
- **Παρεμβολή (interpolation):** Προσπαθούν να παράγουν ένα πανομοιότυπο αντίγραφο του χαμένου πακέτου. Για να το επιτύχουν βασίζονται σε ένα συνδυασμό αναγνώρισης προτύπων (pattern matching) και παρεμβολής. Οι τεχνικές αυτές μπορεί να είναι πιο δύσκολες στην υλοποίηση και να απαιτούν μεγαλύτερη υπολογιστική ισχύ, αποδίδουν όμως καλύτερα από τις προηγούμενες.
- **Ανάπλαση (regeneration):** Οι τεχνικές αυτές εξάγουν την κατάσταση του αποκωδικοποιητή εξετάζοντας τα γειτονικά πακέτα του σφάλματος και προσπαθούν κατά αυτόν τον τρόπο να αναπλάσουν το χαμένο πακέτο.



Εικόνα 3.3.1 Τεχνικές επανόρθωσης βασιζόμενες στον παραλήπτη

Στην συνέχεια εξετάζονται με περισσότερες λεπτομέρειες οι παραπάνω κατηγορίες.

3.3.1 Insertion

Όπως προαναφέραμε οι τεχνικές αυτές, αντιμετωπίζουν την περίπτωση σφάλματος εισάγοντας ένα *fill-in* πακέτο. Η πιο απλή περίπτωση εισαγωγής είναι η «συγκόλληση» (*splicing*), κατά την οποία εισάγεται ένα πακέτο μηδενικού μήκους. Μια εναλλακτική προσέγγιση αποτελεί η «αντικατάσταση με σιωπή» (*silence substitution*), όπου το χαμένο πακέτο αντικαθίσταται με σιωπή διάρκειας ίσης με

αυτή του χαμένου πακέτου. Μια παραλλαγή αυτής της τεχνικής είναι αντί της σιωπής να χρησιμοποιηθεί θόρυβος. Τέλος, η επανάληψη του τελευταίου πακέτου (*packet repetition*) αποτελεί και την τελευταία περίπτωση αντιμετώπισης σφαλμάτων με εισαγωγή.

3.3.1.1 Splicing

Τα φωνητικά δεδομένα που χάθηκαν, αποκρύπτονται καθώς συγκολλείται το κενό που αφήνουν με τα γειτονικά δεδομένα του σήματος της φωνής. Όμως αυτό έχει σαν αποτέλεσμα να διαταράσσεται ο συγχρονισμός του αρχικού σήματος φωνής. Αυτή η τεχνική μελετήθηκε από τον Gruber και Strawczynski [35] όπου και αποδείχθηκε ότι έχει χαμηλή απόδοση.

3.3.1.2 Silence Substitution

Η μέθοδος αντικατάστασης με σιωπή, «γεμίζει» το κενό που άφησε ένα χαμένο πακέτο με σιωπή με σκοπό απλά να διατηρήσει τον συγχρονισμό του αρχικού σήματος φωνής. Η απόδοση αυτής της τεχνικής είναι αρκετά καλή στην περίπτωση που το μέγεθος των πακέτων είναι μικρό (<4ms) και ο ρυθμός σφαλμάτων χαμηλός (<2%) [36].

Παρόλο που όσο το μέγεθος των πακέτων μεγαλώνει η απόδοση της παραπάνω μεθόδου μειώνεται δραματικά, χρησιμοποιείται ευρέως λόγω της απλότητας στην υλοποίηση της.

3.3.1.3 Noise Substitution

Επειδή η μέθοδος αντικατάστασης με σιωπή δεν αποδίδει ικανοποιητικά η προφανής επόμενη επιλογή είναι η χρήση του θορύβου σαν μέσο αντικατάστασης του κενού που αφήνει το χαμένο πακέτο.

Μια σειρά από μελέτες, πάνω στον τομέα της ανθρώπινης αντίληψης σε καταστάσεις διακοπόμενης ομιλίας [37] έδειξαν ότι υποσυνείδητα ο ανθρώπινος εγκέφαλος μπορεί να αντικαταστήσει καλύτερα ένα χαμένο κομμάτι ομιλίας, με το σωστό κομμάτι, όταν χρησιμοποιηθεί η μέθοδος αντικατάστασης με θόρυβο παρά η μέθοδος αντικατάστασης με σιωπή.

3.3.1.4 Packet Repetition

Αυτή η τεχνική αντικαθιστά κάποιο χαμένο πακέτο, με το τελευταίο πακέτο που παρέλαβε σωστά. Χαρακτηρίζεται από χαμηλή υπολογιστική πολυπλοκότητα και έχει σχετικά καλή απόδοση. Η απόδοση της μπορεί να βελτιωθεί αν βαθμιαία εφαρμοστεί η τεχνική του *fading* στα επαναλαμβανόμενα πακέτα. Για παράδειγμα το GSM, υποστηρίζει την επανάληψη των πρώτων 20ms με το ίδιο πλάτος και στην συνέχεια την βαθμιαία μείωση του πλάτους του επαναλαμβανόμενου σήματος για τα επόμενα 320ms [38].

Συνοψίζοντας, η μέθοδος της επανάληψης του τελευταίου πακέτου σε συνδυασμό με την χρήση της τεχνικής *fading*, αποτελεί μια καλή συμβιβαστική λύση μεταξύ των χαμηλής απόδοσης τεχνικών εισαγωγής (Insertion) και των υψηλής υπολογιστικής πολυπλοκότητας μεθόδων παρεμβολής (Interpolation) και ανάπλασης (Regeneration).

3.3.2 Interpolation

Οι τεχνικές αυτές επιδιώκουν, εφαρμόζοντας μεθόδους παρεμβολής και αναγνώρισης προτύπων στα γειτονικά πακέτα του σφάλματος να παράγουν ένα καινούργιο πακέτο ώστε να αναπληρωθεί το χαμένο. Το μεγάλο πλεονέκτημα αυτών των μεθόδων έναντι των τεχνικών εισαγωγής είναι ότι λαμβάνουν υπόψιν τους τα φυσικά χαρακτηριστικά του σήματος της φωνής.

3.3.2.1 Waveform Substitution

Η μέθοδος της αντικατάστασης της κυματομορφής χρησιμοποιεί τα δεδομένα φωνής που βρίσκονται στα γειτονικά πακέτα του σφάλματος με σκοπό να βρει το κατάλληλο ηχητικό σήμα για να καλύψει το σφάλμα. Πιο συγκεκριμένα, λαμβάνει υπόψιν της πληροφορία πριν το σφάλμα (μονόπλευρη) και προαιρετικά πληροφορία προερχόμενη μετά το σφάλμα (αμφίπλευρη). Μια μελέτη αυτής της τεχνικής έχει διεξαχθεί από τον Goodman [39], κατά οποία την οποία εξετάστηκαν τόσο η μονόπλευρη όσο η αμφίπλευρη αντικατάσταση κυματομορφής, οι οποίες μάλιστα χρησιμοποιούν έτοιμες «φόρμες» για να εντοπίσουν το καταλληλότερο στιγμιαίο πρότυπο (*pitch pattern*) με σκοπό να καλύψουν το σφάλμα. Στην περίπτωση της μονόπλευρης αντικατάστασης κυματομορφής το επιλεγμένο πρότυπο απλά επαναλαμβάνεται για να καλύψει το κενό του σφάλματος ενώ στην αμφίπλευρη περίπτωση γίνεται χρήση παρεμβολής. Η τελευταία παρουσιάζει καλύτερη απόδοση από την μονόπλευρη τεχνική και οι δυο τους αποδίδουν καλύτερα από τις μεθόδους αντικατάστασης με σιωπή και επανάληψης τελευταίου πακέτου.

3.3.2.2 Pitch Waveform Replication

Μια παραλλαγή της παραπάνω τεχνικής παρουσιάστηκε από τον Wasen [40], ο οποίος χρησιμοποίησε έναν αλγόριθμο *pitch detection* λαμβάνοντας υπόψιν και τα δυο μέρη του σφάλματος. Τα σφάλματα που δεν περιέχουν δεδομένα φωνής επιδιορθώνονται χρησιμοποιώντας την μέθοδο επανάληψης τελευταίου πακέτου, ενώ στην περίπτωση απώλειας φωνητικών δεδομένων επαναλαμβάνεται μια κατάλληλη κυματομορφή. Η παραπάνω μελέτη έδειξε ότι αυτή η μέθοδος λειτουργεί οριακά καλύτερα από την τεχνική της αντικατάστασης κυματομορφής.

3.3.2.3 Time Scale Modification

Η μέθοδος αυτή λειτουργεί «επεκτείνοντας» το σήμα φωνής και από τα δυο μέρη του σφάλματος καλύπτοντας κατά αυτόν τον τρόπο το κενό που αφήνει το

τελευταίο. Η μελέτη αυτής της τεχνικής διεξήχθη από τον Sanneck [41] όπου και αποδείχθηκε ότι ενώ μπορεί να είναι υπολογιστικά πιο απαιτητική από τις προηγούμενες μεθόδους λειτουργεί αποδοτικότερα από αυτές.

3.3.3 Regeneration

Η κατηγορία των τεχνικών αναπαραγωγής χρησιμοποιούν την γνώση του αλγόριθμου συμπίεσης για να εξάγουν τις παραμέτρους του κωδικοποιητή έτσι ώστε να είναι σε θέση να συνθέσουν το χαμένο μέρος των φωνητικών δεδομένων. Οι τεχνικές αυτές εξαρτώνται όπως είναι φυσικό από το είδος του κωδικοποιητή και έχουν πολύ καλή απόδοση. Το μεγάλο μειονέκτημα τους είναι η υπολογιστικές τους απαιτήσεις που τις καθιστούν απαγορευτικές για αλληλεπιδραστικές εφαρμογές πραγματικού χρόνου, όπως στην προκειμένη περίπτωση η μετάδοση φωνής μέσω του διαδικτύου. Στην συνέχεια παρουσιάζεται αναλυτικότερα η οικογένεια αυτών των μεθόδων.

3.3.3.1 Interpolation of Transmitted State

Στους κωδικοποιητές που βασίζονται σε linear prediction ή σε transform coding, είναι δυνατό στον αποκωδικοποιητή η χρήση μεθόδων παρεμβολής μεταξύ των καταστάσεων [42]. Η απόδοση μεν μπορεί να είναι πολύ καλή, αλλά η χρήση αυτών των κωδικοποιητών (στους οποίους μπορεί να εφαρμοστεί η παρεμβολή) προϋποθέτει υψηλές υπολογιστικές απαιτήσεις.

3.3.3.2 Model -Based Recovery

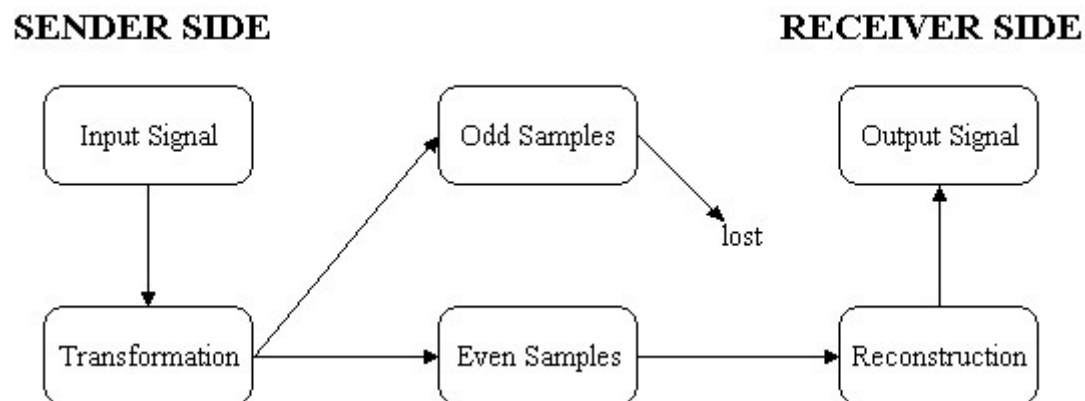
Σε αυτήν την μέθοδο η ομιλία από το ένα ή και από τα δυο μέρη του σφάλματος «προσαρμόζεται» σε ένα μοντέλο, το οποίο χρησιμοποιείται για να παραχθεί η χαμένη ομιλία [43]. Και αυτή η μέθοδος χαρακτηρίζεται από υψηλές απαιτήσεις υπολογιστικού κόστους με αποτέλεσμα οι υπεισερχόμενες καθυστερήσεις να καθιστούν απαγορευτική την χρήση της σε αλληλεπιδραστικές εφαρμογές πραγματικού χρόνου.

3.4 Ανακατασκευή μέσου όρου με χρήση μετασχηματισμών και διαφύλλωσης

Στα προηγούμενα κεφάλαια εξετάστηκαν μέθοδοι ανακατασκευής χαμένων πακέτων φωνής, οι οποίες στηρίζονταν σε λειτουργίες που συντελούνταν είτε στον αποστολέα είτε στον παραλήπτη. Στο συμπέρασμα στο οποίο καταλήξαμε είναι οι ότι παραπάνω μέθοδοι θα πρέπει να χρησιμοποιούνται ταυτόχρονα για να επιτύχουμε τα μεγαλύτερα δυνατά αποτελέσματα. Πιο συγκεκριμένα, για την αντιμετώπιση μεγάλου μήκους σφαλμάτων επιλέγεται μια τεχνική επανόρθωσης βασισμένη στον αποστολέα, ενώ για τα μικρότερου μήκους σφάλματα χρησιμοποιείται κάποια τεχνική επανόρθωσης βασισμένη στον παραλήπτη.

Σε αυτό το κεφάλαιο θα παρουσιαστεί μια τεχνική ανακατασκευής, η οποία βασίζεται στην συνεργασία και των δυο [44]. Ειδικότερα, ο αποστολέας εφαρμόζει αρχικά έναν μετασχηματισμό στα δεδομένα και στην συνέχεια τα αποστέλλει διαφυλλωμένα. Ο μετασχηματισμός αυτός έχει σαν σκοπό την ελαχιστοποίηση του λάθους ανακατασκευής στην περίπτωση κατά την οποία συμβεί κάποια απώλεια πακέτου. Αναλυτικότερα, ο παραλήπτης ανακατασκευάζει τα χαμένα πακέτα χρησιμοποιώντας μια μέθοδο επανόρθωσης βασισμένη στην παρεμβολή (interpolation-based reconstruction). Τέλος, όταν ο ληφθούν όλα τα διαφυλλωμένα πακέτα, ο παραλήπτης είναι σε θέση να αναπαράγει τα αρχικά δεδομένα.

Ο αλγόριθμος παρουσιάζεται συνοπτικά στην εικόνα 3.4.1.



Εικόνα 3.4.1 Ανακατασκευή με χρήση μετασχηματισμού και διαφύλλωσης

Στην περίπτωση που παρουσιάζεται στην παραπάνω εικόνα, η διαφύλλωση είναι ενός επιπέδου και χωρίζει τα δεδομένα σε δυο ροές. Στα πλαίσια αυτής της εργασίας εξετάζεται και η περίπτωση διαφύλλωσης δυο επιπέδων ώστε να καλυφθεί η περίπτωση εμφάνισης συσσωρευμένων λαθών (burst errors).

Ως μέτρο ποιότητας της ανακατασκευής χρησιμοποιούμε τον σηματοθορυβικό λόγο SNR, που δίνεται από την παρακάτω σχέση (Signal to Noise Ratio):

$$SNR = 10 \log_{10} \frac{\sum (s^2)}{\sum (s - \hat{s})^2} \quad (3.1)$$

Όπου το s αποτελεί το δείγμα του αρχικού σήματος φωνής και το \hat{s} , το αντίστοιχο δείγμα του ανακατασκευασμένου σήματος.

3.4.1 Ανακατασκευή μέσου όρου με χρήση μετασχηματισμού και διαφύλλωσης ενός επιπέδου

Σε αυτό το σημείο αρχικά θα εξεταστεί αναλυτικότερα, η περίπτωση εφαρμογής διαφύλλωσης ενός επιπέδου στον αποστολέα και η ανακατασκευή στον παραλήπτη με την χρήση μιας μεθόδου παρεμβολής με την ονομασία ανακατασκευή μέσου όρου (average reconstruction). Επίσης, πρέπει να αναφερθεί ότι η τεχνική ανακατασκευής μέσου όρου δεν έχει υψηλές υπολογιστικές απαιτήσεις, σε αντίθεση με τις υπόλοιπες τεχνικές της οικογένειας των interpolation-based reconstruction μεθόδων.

Υποθέτουμε ότι το αρχικό σήμα αποτελείται από τα δείγματα x_0, x_1, x_2, \dots και το κάθε πακέτο έχει μέγεθος N δειγμάτων. Επειδή εφαρμόζουμε διαφύλλωση σε δύο μέρη, στο πακέτο P_1 τοποθετούνται τα άρτια δείγματα x_0, x_2, x_{2N-2} , και στο πακέτο P_2 τα περιττά δείγματα x_1, x_3, x_{2N-1} .

Ο αποστολέας μεταδίδει στον παραλήπτη μέσω δικτύου τα πακέτα P_1 και P_2 , τα οποία αποτελούν αυτό που ονομάζουμε ζεύγος διαφύλλωσης. Αν υποθέσουμε ότι ο παραλήπτης λαμβάνει κανονικά το P_1 αλλά όχι P_2 , τότε το ανακατασκευάζει χρησιμοποιώντας την μέθοδο παρεμβολής ανακατασκευή μέσου όρου (average reconstruction):

$$\hat{x}_{2j-1} = \frac{x_{2j-2} + x_{2j}}{2}, \text{ όπου } j = 1, \dots, N \quad (3.2)$$

Σε αυτήν την μέθοδο, ο αποστολέας απλά εφαρμόζει μια διαφύλλωση σε δύο μέρη και ομαδοποιεί σχετιζόμενη πληροφορία σε διαφορετικά πακέτα με αποτέλεσμα η εμφάνιση ενός μεμονωμένου σφάλματος να μην προκαλεί την απώλεια ενός ολόκληρου τμήματος της ίδιας πληροφορίας. Διαισθητικά αλλά και πειραματικά η παραπάνω μέθοδος βελτιώνει την ποιότητα της μεταδιδόμενης πληροφορίας, παρουσία κάποιου μεμονωμένου σφάλματος. Παρόλα αυτά, δεν παρέχει καμία εγγύηση για την ποιότητα του ανακατασκευασμένου σήματος της φωνής. Αυτό οφείλεται στο γεγονός ότι η κυματομορφή του σήματος της φωνής μπορεί να έχει πολύ απότομες διακυμάνσεις με αποτέλεσμα η τεχνική της ανακατασκευής μέσου όρου να μην δίνει ακριβή αποτελέσματα.

Για να αντεπεξέλθουμε αυτού του προβλήματος χωρίς να χρειαστεί να αυξήσουμε το bandwidth του δικτύου (όπως στην περίπτωση χρήσης FEC), ο αποστολέας

εφαρμόζει έναν μετασχηματισμό στο αρχικό σήμα φωνής, πριν το διαφυλλώσει και πριν το μεταδώσει μέσω του δικτύου (βλ. εικόνα 3.4.1). Ο μετασχηματισμός αυτός έχει σαν σκοπό στην εμφάνιση κάποιου μεμονωμένου σφάλματος, η προσπάθεια ανακατασκευής στον παραλήπτη με την τεχνική ανακατασκευής μέσου όρου να παράγει την καλύτερη δυνατή προσέγγιση του αρχικού σήματος. Το κριτήριο ποιότητας της ανακατασκευής ορίζεται σαν το μέσο τετραγωνικό λάθος (mean square error) μεταξύ του αρχικού και του ανακατασκευασμένου σήματος.

Στις παραγράφους που ακολουθούν παρουσιάζεται με λεπτομέρειες αυτός ο μετασχηματισμός. Πιο συγκεκριμένα εξετάζονται δυο περιπτώσεις στον παραλήπτη στην πρώτη λαμβάνει χώρα παράδοση ενός από τα δυο πακέτα του ζεύγους διαφύλλωσης, ενώ στην δεύτερη παραλαμβάνονται και τα δυο πακέτα.

3.4.1.1 Απώλεια ενός πακέτου σε ένα ζεύγος διαφύλλωσης

Έστω $\vec{x} = x_0, x_1, \dots, x_{2N-1}$ τα αρχικά δεδομένα φωνής. Ο αποστολέας εφαρμόζει σε αυτά έναν μετασχηματισμό \mathbf{T} και παράγει τα $\vec{y} = y_0, y_1, \dots, y_{2N-1}$. Στην συνέχεια τα μετασχηματισμένα δεδομένα διαφυλλώνονται και αποστέλλονται στον παραλήπτη. Χωρίς απώλεια της γενικότητας, υποθέτουμε ότι μόνο όλα τα άρτια δείγματα $\vec{y}_{even} = y_0, y_2, \dots, y_{2N-2}$ παραδόθηκαν επιτυχώς.

Ο παραλήπτης εφαρμόζοντας την μέθοδο ανακατασκευής μέσου όρου, παράγει το σήμα $\vec{\hat{y}} = \hat{y}_0, \hat{y}_1, \dots, \hat{y}_{2N-1}$ ως εξής:

$$\hat{y}_i = \begin{cases} y_i & i \text{ even} \\ \frac{y_{i-1} + y_{i+1}}{2} & i \text{ odd and } i \neq 2N-1 \\ \frac{y_{2N-2}}{2} & i = 2N-1 \end{cases} \quad (3.3)$$

Όπου σαν λάθος ανακατασκευής RE (Reconstruction Error) ορίζεται το:

$$RE = \sum_{i=0}^{2N-1} (x_i - y_i)^2 = \sum_{n=0}^{N-1} (x_{2n} - y_{2n})^2 + \sum_{n=0}^{N-2} \left(x_{2n+1} - \frac{y_{2n} + y_{2n+2}}{2} \right)^2 + \left(x_{2N-1} - \frac{y_{2N-2}}{2} \right)^2 \quad (3.4)$$

Για να ελαχιστοποιηθεί το λάθος ανακατασκευής πρέπει να υπολογιστεί το y_i έτσι ώστε να ικανοποιούνται οι παρακάτω εξισώσεις, για κάθε άρτιο i .

$$\frac{\partial RE}{\partial y_i} = 0, i = 0, 2, 4, 6, \dots, 2N-2 \quad (3.5)$$

Αντικαθιστώντας το RE στην σχέση (3.5) και χρησιμοποιώντας την σχέση (3.4) καταλήγουμε στον μετασχηματισμό πινάκων:

$$\begin{pmatrix} y_0 \\ y_2 \\ \vdots \\ y_{2N-4} \\ y_{2N-2} \end{pmatrix} = \mathbf{T} \times \begin{pmatrix} x_0 \\ x_1 \\ \vdots \\ x_{2N-2} \\ x_{2N-1} \end{pmatrix} \quad (3.6)$$

όπου

$$\mathbf{T} = \mathbf{A}^{-1} \times \mathbf{B} =$$

$$\begin{pmatrix} 1 & 1/5 & 0 & \cdots & 0 & 0 & 0 \\ 1/6 & 1 & 1/6 & \cdots & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & 1/6 & 1 & 1/6 \\ 0 & 0 & 0 & \cdots & 0 & 1/6 & 1 \end{pmatrix}^{-1} \begin{pmatrix} 4/5 & 2/5 & 0 & 0 & \cdots & 0 & 0 & 0 & 0 & 0 \\ 0 & 1/3 & 2/3 & 1/3 & \cdots & 0 & 0 & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & \cdots & 1/3 & 2/3 & 1/3 & 0 & 0 \\ 0 & 0 & 0 & 0 & \cdots & 0 & 0 & 1/3 & 2/3 & 1/3 \end{pmatrix} \quad (3.7)$$

Η συνθήκη για να έχει μοναδική λύση η εξίσωση (3.6) είναι να υπάρχει αντίστροφος για τον πίνακα \mathbf{A} της σχέσης (3.7). Η απόδειξη της αντιστρεψιμότητας του πίνακα \mathbf{A} , παρατίθεται στο Παράρτημα Α.

Χρησιμοποιώντας την σχέση (3.6), υπολογίζονται τα άρτια y : $y_{2n}, n = 0, 1, 2, \dots, N-1$.

Για να παραχθεί ο πλήρης μετασχηματισμός από τα \bar{x} στα \bar{y} , πρέπει να υπολογιστούν και τα περιττά y : $\bar{y}_{odd} = y_1, y_3, \dots, y_{2N-1}$ ακολουθώντας ανάλογη διαδικασία όπως και για τα άρτια.

Έτσι στην περίπτωση απώλειας όλων των άρτιων δειγμάτων, η εξίσωση (3.3) τροποποιείται σε

$$\hat{y}_i = \begin{cases} y_i & i \text{ odd} \\ \frac{y_{i-1} + y_{i+1}}{2} & i \text{ even and } i \neq 0 \\ \frac{y_1}{2} & i = 0 \end{cases} \quad (3.8)$$

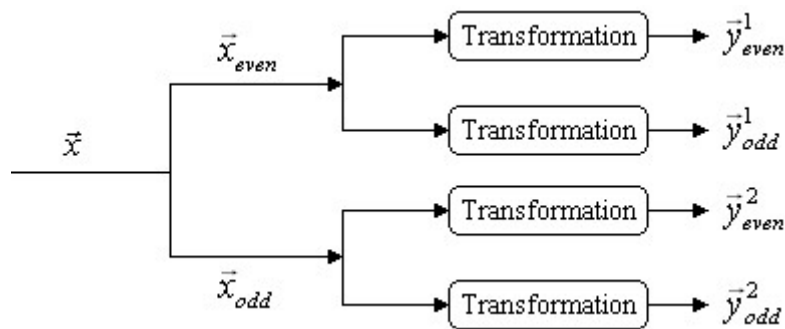
Επίσης και το λάθος ανακατασκευής RE υπολογίζεται ανάλογα:

$$RE = \sum_{i=0}^{2N-1} (x_i - y_i)^2 = \sum_{n=0}^{N-1} (x_{2n+1} - y_{2n+1})^2 + \sum_{n=1}^{N-1} \left(x_{2n} - \frac{y_{2n-1} + y_{2n+1}}{2} \right)^2 + \left(x_0 - \frac{y_1}{2} \right)^2 \quad (3.9)$$

Τέλος ο μετασχηματισμός για τα \bar{y}_{odd} έχει ως εξής:

$$\bar{y}_{odd} = \begin{pmatrix} 1 & 1/6 & 0 & \cdots & 0 & 0 & 0 \\ 1/6 & 1 & 1/6 & \cdots & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & 1/6 & 1 & 1/6 \\ 0 & 0 & 0 & \cdots & 0 & 1/5 & 1 \end{pmatrix}^{-1} \begin{pmatrix} 1/3 & 2/3 & 1/3 & 0 & 0 & \cdots & 0 & 0 & 0 & 0 \\ 0 & 0 & 1/3 & 2/3 & 1/3 & \cdots & 0 & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & \cdots & 0 & 1/3 & 2/3 & 1/3 & 0 \\ 0 & 0 & 0 & 0 & \cdots & 0 & 0 & 0 & 2/5 & 4/5 \end{pmatrix} \bar{x} \quad (3.10)$$

Ένας τρόπος να αντιμετωπιστεί η εμφάνιση συσσωρευμένων λαθών είναι ο συνδυασμός του μετασχηματισμού που ορίστηκε προηγουμένως με την παράλληλη χρήση διαφύλλωσης δυο επιπέδων. (βλ εικόνα 3.4.2)



Εικόνα 3.4.2 Ανακατασκευή με χρήση μετασχηματισμού και διαφύλλωσης δυο επιπέδων

Το παραπάνω σχήμα παρουσιάζει τις τέσσερις ροές που δημιουργούνται στον αποστολέα. Πιο συγκεκριμένα η αρχική ροή \bar{x} διαφυλλώνεται σε μία άρτια \bar{x}_{even} και μια περιττή \bar{x}_{odd} ροή. Στην συνέχεια κάθε μια από αυτές τις ροές μετασχηματίζεται όπως περιγράψαμε προηγουμένως και κατά αυτόν τον τρόπο παράγονται τέσσερις διαφορετικές ροές \bar{y}_{even}^1 , \bar{y}_{odd}^1 , \bar{y}_{even}^2 , \bar{y}_{odd}^2 οι οποίες και μεταδίδονται μέσω του διαδικτύου.

Όταν δεν σημειωθεί κάποιο σφάλμα στην μετάδοση, ο παραλήπτης μπορεί εφαρμόζοντας αρχικά τους αντίστοιχους αντίστροφους μετασχηματισμούς να αναπαράγει τις αρχικές ροές \bar{x}_{even} και \bar{x}_{odd} . Στην συνέχεια εφαρμόζοντας μια απλή αποδιαφύλλωση, ανακτά επακριβώς την αρχική ροή \bar{x} .

Όσον αφορά την περίπτωση εμφάνισης κάποιου σφάλματος κατά την μετάδοση διακρίνονται τέσσερα πιθανά σενάρια:

- Στην πρώτη περίπτωση, τρία πακέτα από τέσσερα χάνονται. Για παράδειγμα μπορούμε να υποθέσουμε ότι χάνονται τα \bar{y}_{odd}^1 , \bar{y}_{even}^2 , \bar{y}_{odd}^2 . Όπως αναφέρθηκε στην προηγούμενη παράγραφο (§3.4.1.1), από το \bar{y}_{even}^1 ανακατασκευάζεται βέλτιστα το \bar{y}^1 . Στην συνέχεια ο παραλήπτης υπολογίζει κάθε δείγμα του \bar{x}_{odd} , εφαρμόζοντας την μέθοδο ανακατασκευής μέσου όρου.
- Στην δεύτερη περίπτωση, παρουσιάζεται απώλεια δυο πακέτων τα οποία ανήκουν στις ροές \bar{y}^1 ή \bar{y}^2 . Για παράδειγμα ας υποθέσουμε ότι χάνονται τα πακέτα \bar{y}_{even}^2 και \bar{y}_{odd}^2 . Ο παραλήπτης εφαρμόζοντας τον αντίστροφο

μετασχηματισμό, που περιγράφηκε στην παράγραφο §3.4.1.2, στα \bar{y}_{even}^1 και \bar{y}_{odd}^1 ανακτά επακριβώς το αρχικό πακέτο της ροής \bar{x}_{even} . Τέλος ο παραλήπτης εφαρμόζοντας την μέθοδο ανακατασκευής μέσου όρου ανακατασκευάζει το πακέτο της ροής \bar{x}_{odd} .

- Στην τρίτη περίπτωση, παρατηρείται ξανά απώλεια δυο πακέτων εκ των οποίων το ένα ανήκει στην ροή \bar{y}^1 και το άλλο στην \bar{y}^2 . Εφαρμόζοντας την μέθοδο που αναφέρθηκε στην παράγραφο §3.4.1.1, ανακατασκευάζονται βέλτιστα τα πακέτα που ανήκουν στις ροές \bar{x}_{even} και \bar{x}_{odd} . Στην συνέχεια εφαρμόζοντας μια απλή αποδιαφύλλωση, ανακτάται το πακέτο \bar{x} της αρχική ροής.
- Στην τελευταία περίπτωση, συμβαίνει απώλεια ενός μόνο πακέτου. Ας υποθέσουμε ότι χάνεται το πακέτο \bar{y}_{odd}^2 . Το πακέτο \bar{x}_{even} , ανακτάται επακριβώς αφού παρελήφθησαν τα \bar{y}_{even}^1 και \bar{y}_{odd}^1 . Από το πακέτο \bar{y}_{even}^2 , ανακατασκευάζεται βέλτιστα το πακέτο \bar{x}_{odd} . Τέλος με την εφαρμογή μια απλής αποδιαφύλλωσης, ανακατασκευάζεται βέλτιστα το πακέτο \bar{x} της αρχική ροής.

Η παραπάνω διαδικασία επιτρέπει την αντιμετώπιση συσσωρευμένων λαθών συνολικής διάρκειας έως και τριών πακέτων. Είναι φανερό ότι εισάγοντας παραπάνω επίπεδα διαφύλλωσης, μπορούμε να αντιμετωπίσουμε συσσωρευμένα λάθη μεγαλύτερης συνολικής διάρκειας. Πιο συγκεκριμένα τα M επίπεδα διαφύλλωσης, σε δυο μέρη, δημιουργούν $2M$ ροές πακέτων, γεγονός που αντιστοιχεί σε αντοχή απωλειών έως και $2M-1$ πακέτων.

Όμως από την άλλη το M δεν μπορεί να είναι πολύ μεγάλο γιατί η υπεισερχόμενη καθυστέρηση αποτελεί ανασταλτικό παράγοντα για εφαρμογές πραγματικού χρόνου όπως και στην περίπτωση μετάδοσης φωνής μέσω του διαδικτύου. Έτσι για αυτού του είδους τις εφαρμογές, το άνω αποδεκτό όριο της τιμής του M δεν πρέπει να ξεπερνά το 2.

3.4.3 Πειραματικά αποτελέσματα

Σε αυτό το σημείο, εφαρμόζονται οι παραπάνω αλγόριθμοι σε πραγματικά αρχεία ήχου με σκοπό να μετρηθεί η απόδοση τους ώστε να εξαχθούν τα κατάλληλα συμπεράσματα. Για τον σκοπό αυτό υλοποιήθηκε ένας προσομοιωτής, ο οποίος δέχεται ως παραμέτρους το αρχείο ήχου καθώς και το μέγεθος του πακέτου που θα μεταδιδόταν μέσω του διαδικτύου. Στην συνέχεια επιλέγεται κάποια από τις παραπάνω τεχνικές ανακατασκευής και υπολογίζεται ο αντίστοιχος σηματοθορυβικός

λόγος SNR (βλ. σχέση 3.1). Ο λόγος αυτός δείχνει πόσο πιο ισχυρό είναι το αρχικό σήμα ήχου σε σχέση με τον θόρυβο που παράγεται από την εκάστοτε τεχνική ανακατασκευής.

Επίσης, ο προσομοιωτής παρέχει την δυνατότητα στον χρήστη να ακούσει το τελικό αρχείο ήχου με τα ανακατασκευασμένα δείγματα ώστε να μπορέσει και ο ίδιος να αντιληφθεί την ποιότητα ανακατασκευής της αντίστοιχης τεχνικής που έχει επιλέξει.

Η υλοποίηση του προσομοιωτή πραγματοποιήθηκε με την χρήση της γλώσσας Java [11] και για την επεξεργασία των αρχείων ήχου χρησιμοποιήθηκε ένα εξειδικευμένο API της Java, ονόματι Java™ Sound API [45].

Όλες οι τεχνικές εφαρμόστηκαν στο ίδιο σύνολο τεσσάρων αρχείων ήχου, των οποίων μια σύντομη περιγραφή παρατίθεται στον παρακάτω πίνακα:

Όνομα Αρχείου	Περιγραφή	Τύπος
x-files.wav	Μουσικό	Μονοφωνικό
milfish.wav	Παιδική φωνή	Στερεοφωνικό
internet.wav	Παιδική φωνή	Μονοφωνικό
computers.wav	Αντρική φωνή	Μονοφωνικό

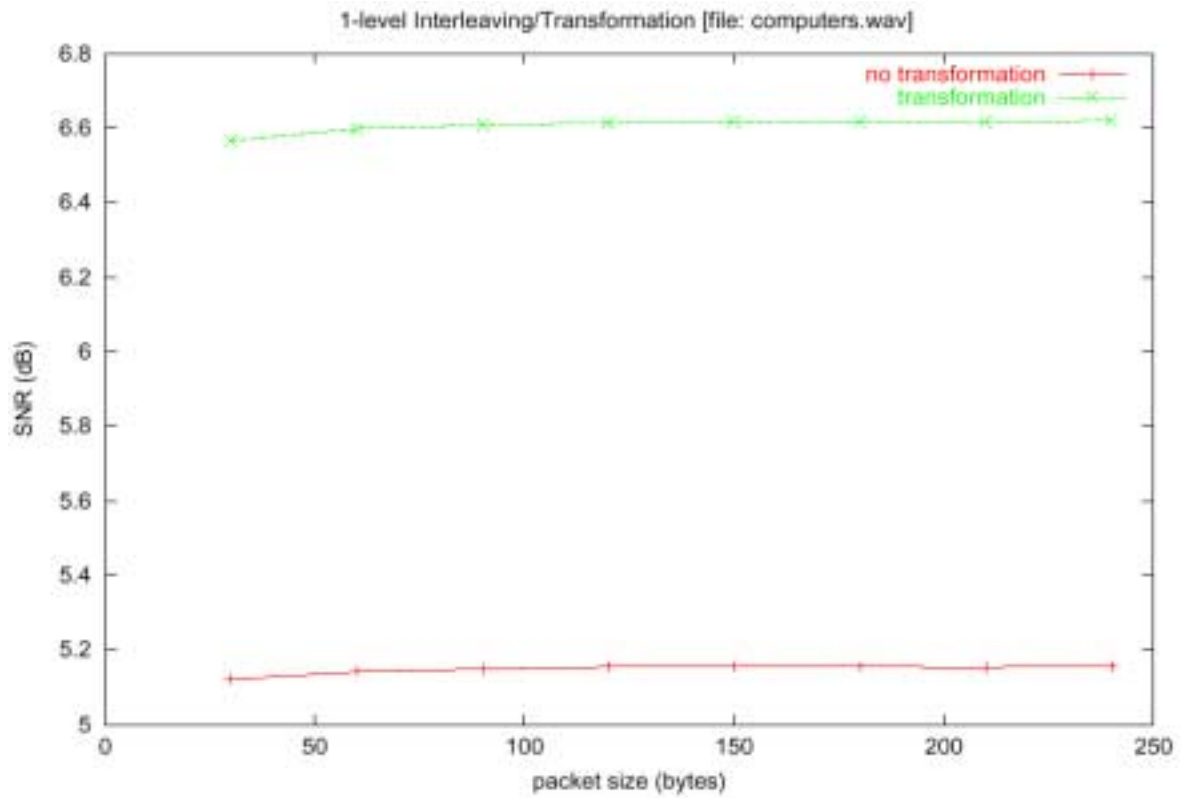
Πίνακας 3.4.1 Χαρακτηριστικά αρχείων ήχου

Το πρώτο σύνολο πειραμάτων αφορά την περίπτωση που αναφέρθηκε στην παράγραφο §3.4.1.1. Πιο συγκεκριμένα εξετάστηκε η περίπτωση κατά την οποία σημειώνεται μια απώλεια πακέτου σε ένα ζεύγος διαφύλλωσης. Διεξήχθησαν πειράματα με την χρήση του μετασχηματισμού αλλά και χωρίς αυτόν ώστε να εξαχθούν τα απαραίτητα συμπεράσματα. Για λόγους απλότητας θεωρήσαμε ότι σε κάθε ζεύγος διαφύλλωσης χάνεται ακριβώς ένα πακέτο (πάντα το άρτιο). Τέλος για κάθε αρχείο, οι προσομοιώσεις πραγματοποιήθηκαν για διάφορα μεγέθη αρχείων: από 20 bytes έως και 240 bytes. Η επιλογή αυτή του εύρους του μεγέθους των πακέτων βασίζεται στο γεγονός ότι οι περισσότερες εφαρμογές μετάδοσης φωνής μέσω του διαδικτύου, παράγουν πακέτα αντιστοίχων μεγεθών [21,29].

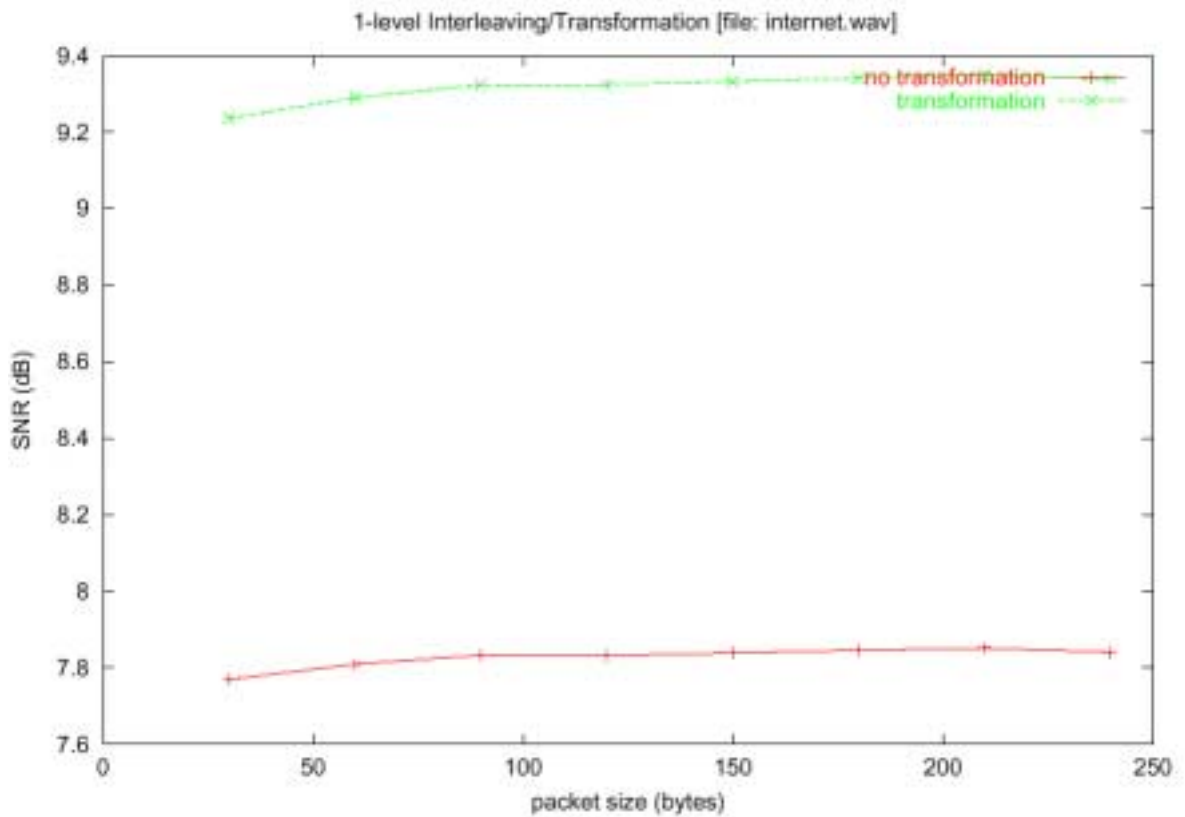
Τα αποτελέσματα παρουσιάζονται στις εικόνες 3.4.3 έως και 3.4.6. Είναι προφανές ότι η μέθοδος ανακατασκευής με χρήση μετασχηματισμού μπορεί να εφαρμοστεί εξίσου σε αρχεία φωνής αλλά και μουσικά αρχεία και αποδίδει σαφώς καλύτερα από την περίπτωση εφαρμογής μεθόδου ανακατασκευής χωρίς την χρήση του μετασχηματισμού. Επίσης παρατηρούμε ότι σε όλες τις περιπτώσεις ο σηματοθορυβικός λόγος SNR , του ανακατασκευασμένου σήματος που έχει υποστεί τον μετασχηματισμό είναι κατά 1.5 με 2.5 dB μεγαλύτερος, το οποίο σημαίνει ότι το λάθος ανακατασκευής RE αυτού του σήματος είναι κατά 70% - 85% μικρότερο από

το αντίστοιχο λάθος ανακατασκευής του σήματος στο οποίο δεν έχει εφαρμοστεί ο μετασχηματισμός.

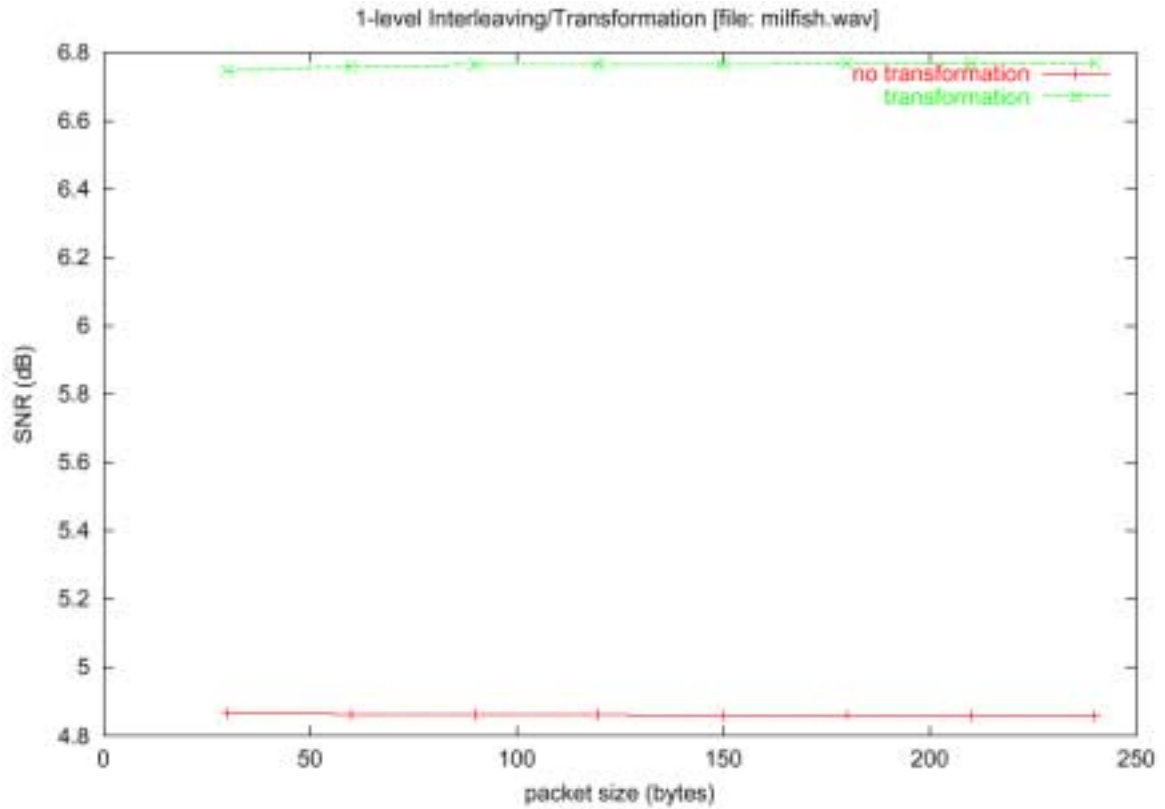
Τέλος πρέπει να σημειωθεί και το γεγονός ότι το μέγεθος πακέτου δεν παίζει βασικό ρόλο στην ποιότητα της ανακατασκευής του σήματος. Αυτό είναι αρκετά σημαντικό γιατί το μέγεθος του πακέτου είναι ανάλογο της καθυστέρησης που υπεισέρχεται τόσο λόγω της διαφύλλωσης αλλά και του υπολογισμού των μετασχηματισμών με την χρήση μεγάλων πινάκων.



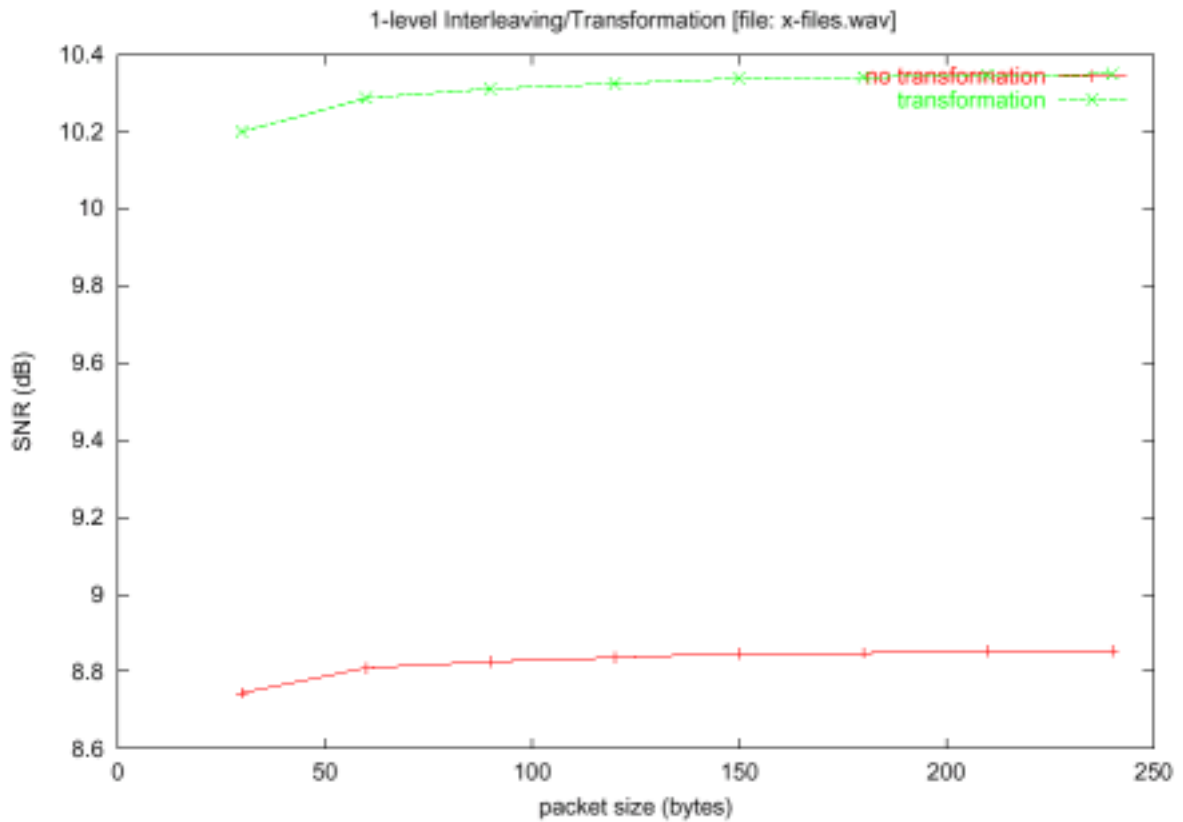
Εικόνα 3.4.3 Σχέση SNR με μέγεθος πακέτου με την χρήση ή όχι μετασχηματισμού(*computers.wav*)



Εικόνα 3.4.4 Σχέση SNR με μέγεθος πακέτου με την χρήση ή όχι μετασχηματισμού(*internet.wav*)



Εικόνα 3.4.5 Σχέση SNR με μέγεθος πακέτου με την χρήση ή όχι μετασχηματισμού(milfish.wav)



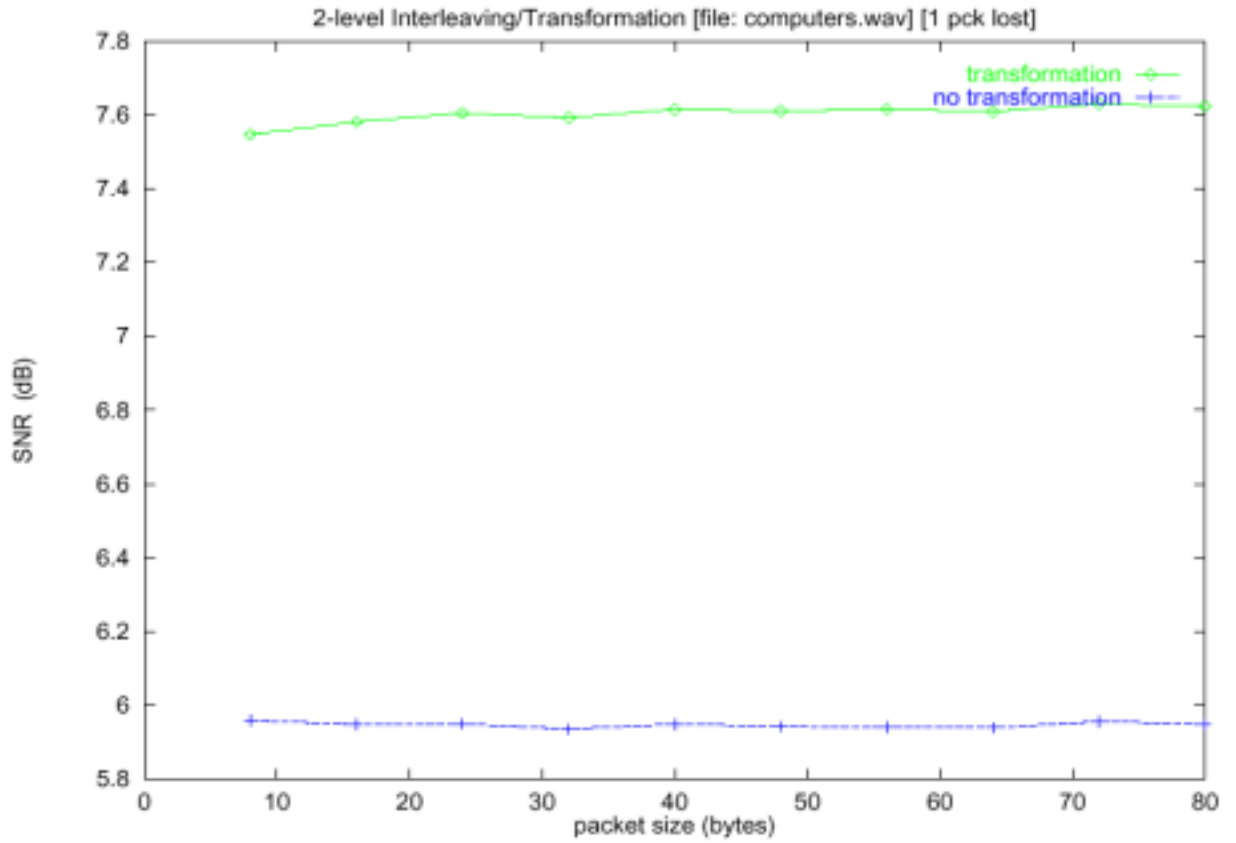
Εικόνα 3.4.6 Σχέση SNR με μέγεθος πακέτου με την χρήση ή όχι μετασχηματισμού(x-files.wav)

Το δεύτερο σύνολο πειραμάτων καλύπτει την περιοχή ανακατασκευής με την εφαρμογή διαφύλλωσης δυο επιπέδων και την χρήση ή όχι μετασχηματισμού. Οι λεπτομέρειες της τεχνικής αυτής έχουν αναλυθεί διεξοδικά στην παράγραφο §3.4.2. Αξίζει σε αυτό το σημείο να υπενθυμίσουμε την σκοπιμότητα αυτής της μεθόδου, η οποία είναι η αντοχή σε απώλειες έως και τριών πακέτων.

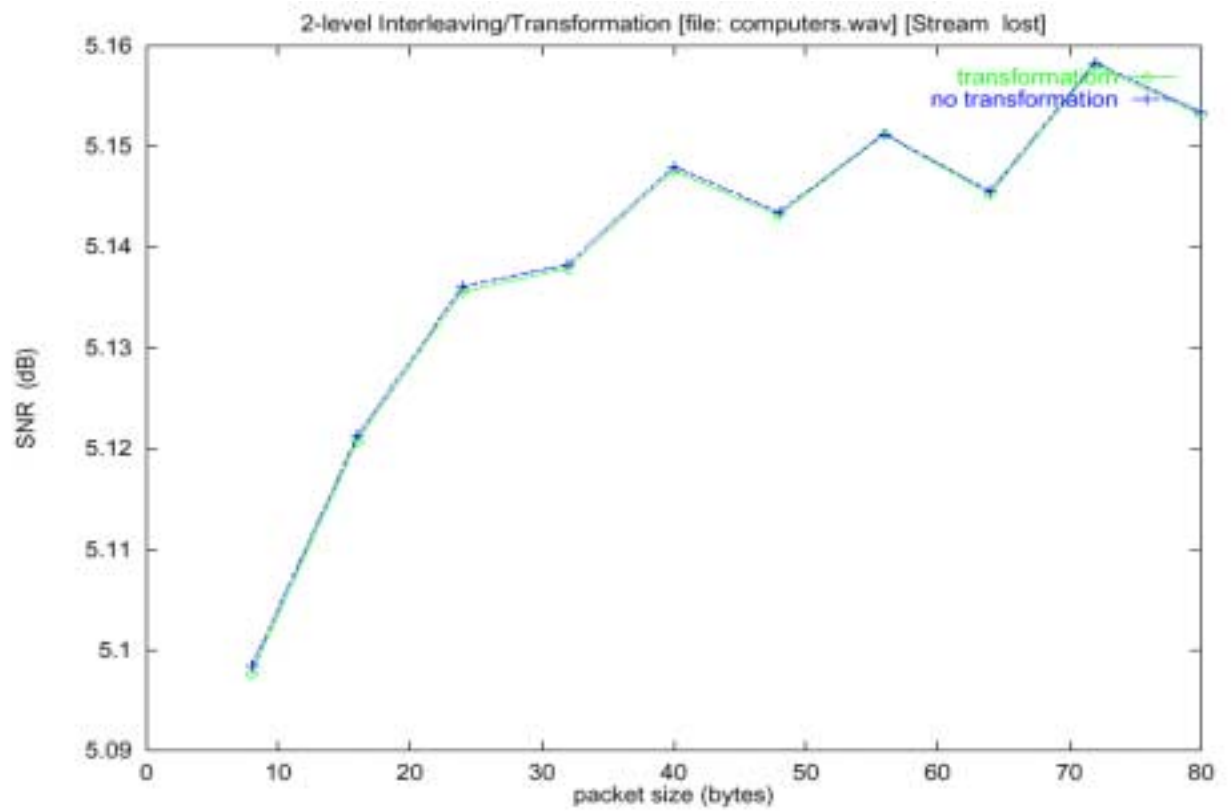
Τα αποτελέσματα παρουσιάζονται στις εικόνες 3.4.7 έως και 3.4.22. Τα συμπεράσματα που εξάγονται και σε αυτήν την περίπτωση είναι ανάλογα όπως και προηγουμένως. Είναι προφανές ότι η τεχνική ανακατασκευής με την εφαρμογή διαφύλλωσης δυο επιπέδων και την χρήση μετασχηματισμού αποδίδει σαφώς καλύτερα από την περίπτωση εφαρμογής μεθόδου ανακατασκευής χωρίς την χρήση του μετασχηματισμού και παράλληλα μπορεί να εφαρμοστεί εξίσου σε αρχεία φωνής αλλά και μουσικά αρχεία.

Αν εξαιρεθεί η περίπτωση, κατά την οποία χάνονται δυο πακέτα της ίδιας ροής (y^1 ή y^2), όπου η ποιότητα ανακατασκευής είναι ανεξάρτητη της εφαρμογής μετασχηματισμού σε όλες τις άλλες περιπτώσεις ο σηματοθορυβικός λόγος SNR , του ανακατασκευασμένου σήματος που έχει υποστεί τον μετασχηματισμό είναι κατά 1.5 με 2.0 dB μεγαλύτερος, το οποίο σημαίνει ότι το λάθος ανακατασκευής RE , αυτού του σήματος είναι κατά 70% - 80% μικρότερο από το αντίστοιχο λάθος ανακατασκευής σήματος στο οποίο δεν έχει εφαρμοστεί ο μετασχηματισμός.

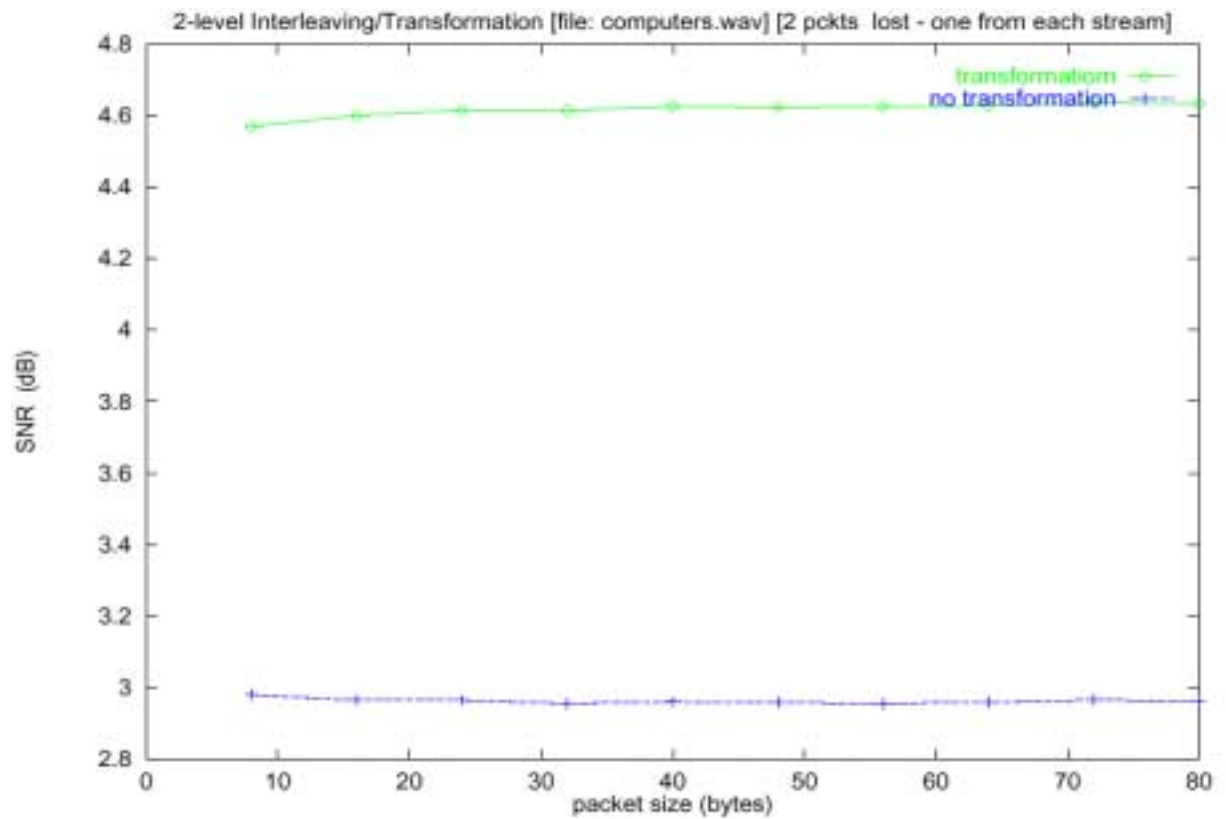
Τέλος παρατηρούμε για ακόμη μια φορά ότι το μέγεθος του πακέτου δεν παίζει βασικό ρόλο στην ποιότητα ανακατασκευής του σήματος. Όπως είχαμε επισημάνει προηγουμένως, το γεγονός αυτό είναι αρκετά σημαντικό γιατί το μέγεθος του πακέτου είναι ανάλογο της καθυστέρησης που υπεισέρχεται τόσο λόγω της διαφύλλωσης όσο και από την χρήση μεγάλων πινάκων.



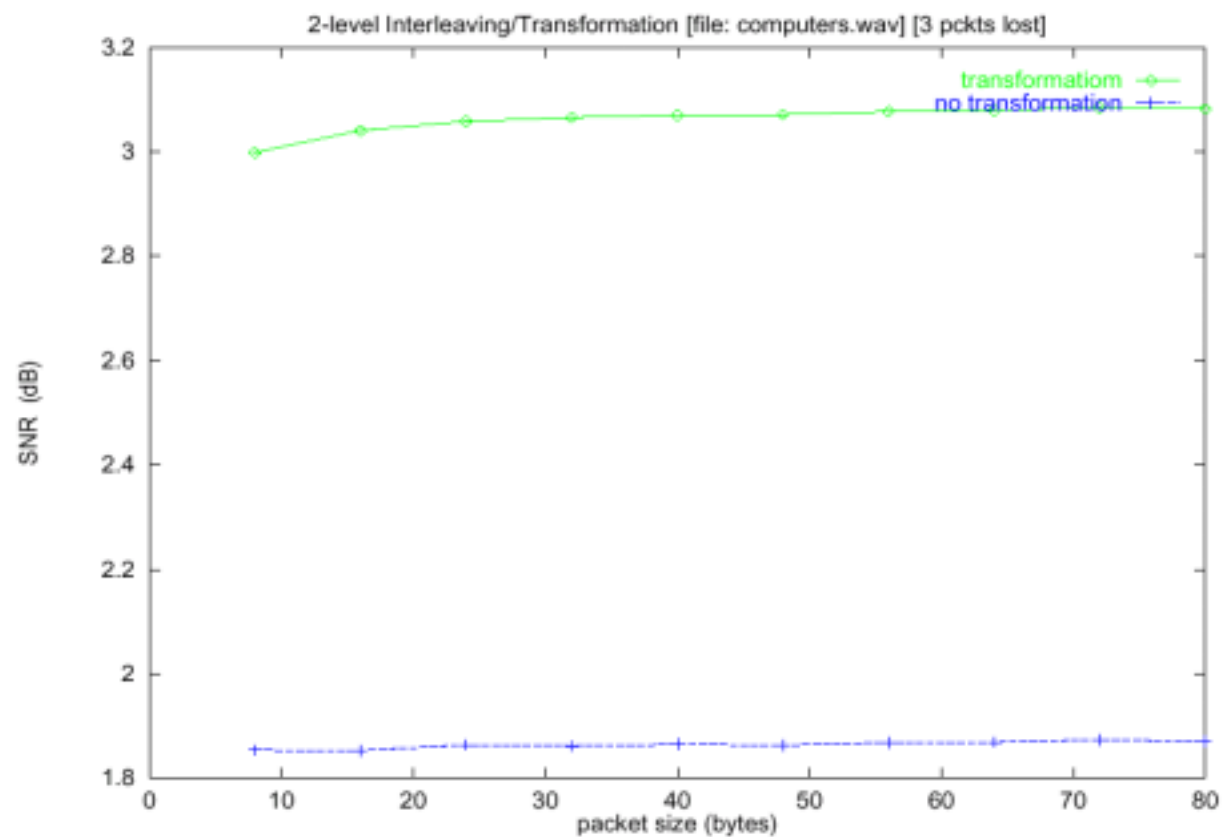
Εικόνα 3.4.7 Απώλεια 1 πακέτου (*computers.wav*)



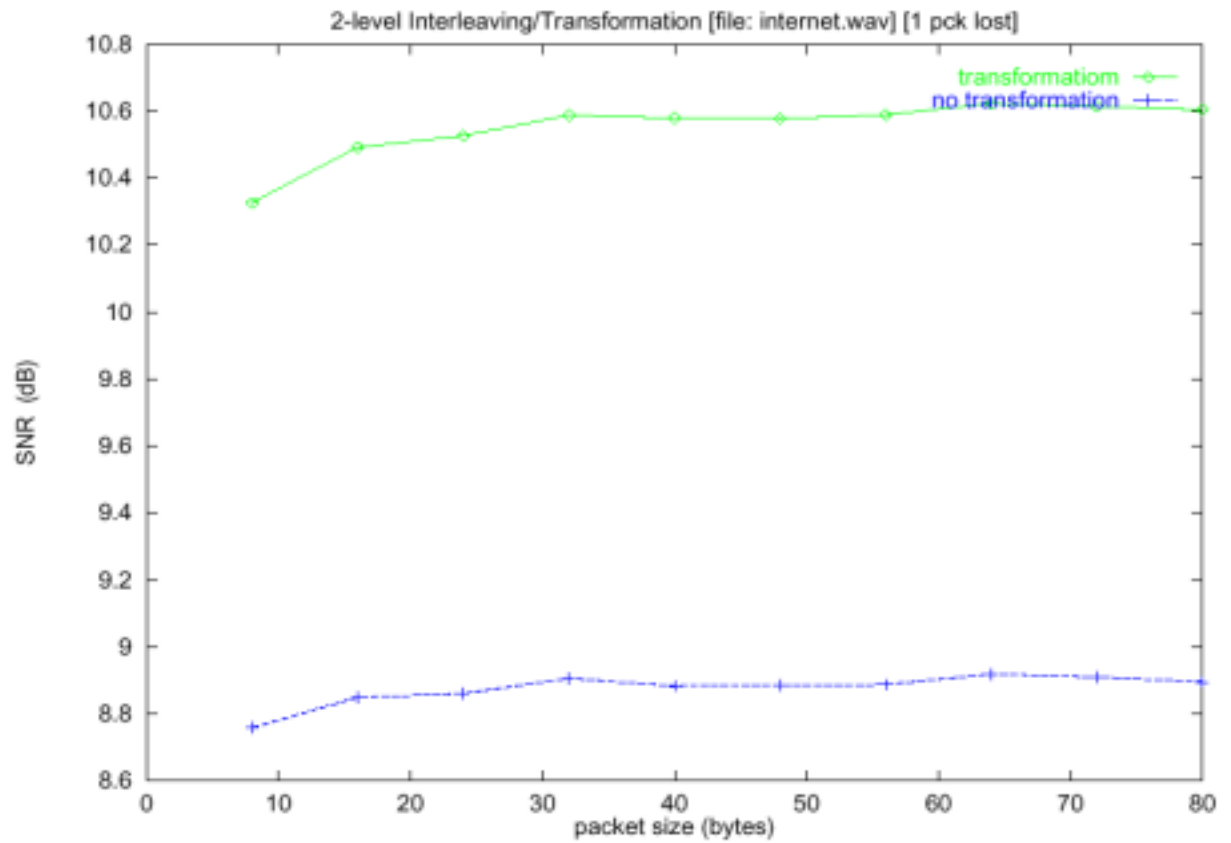
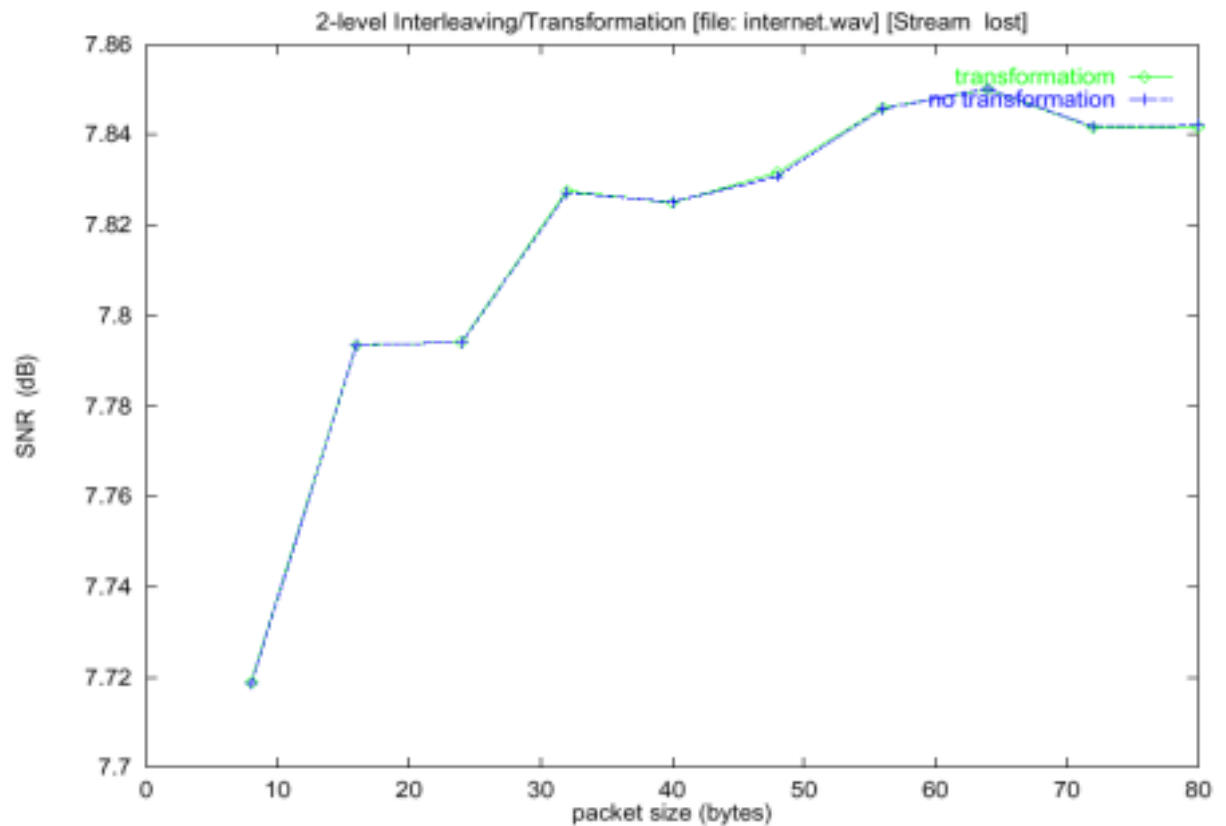
Εικόνα 3.4.8 Απώλεια 2 πακέτων της ίδιας ροής (*computers.wav*)

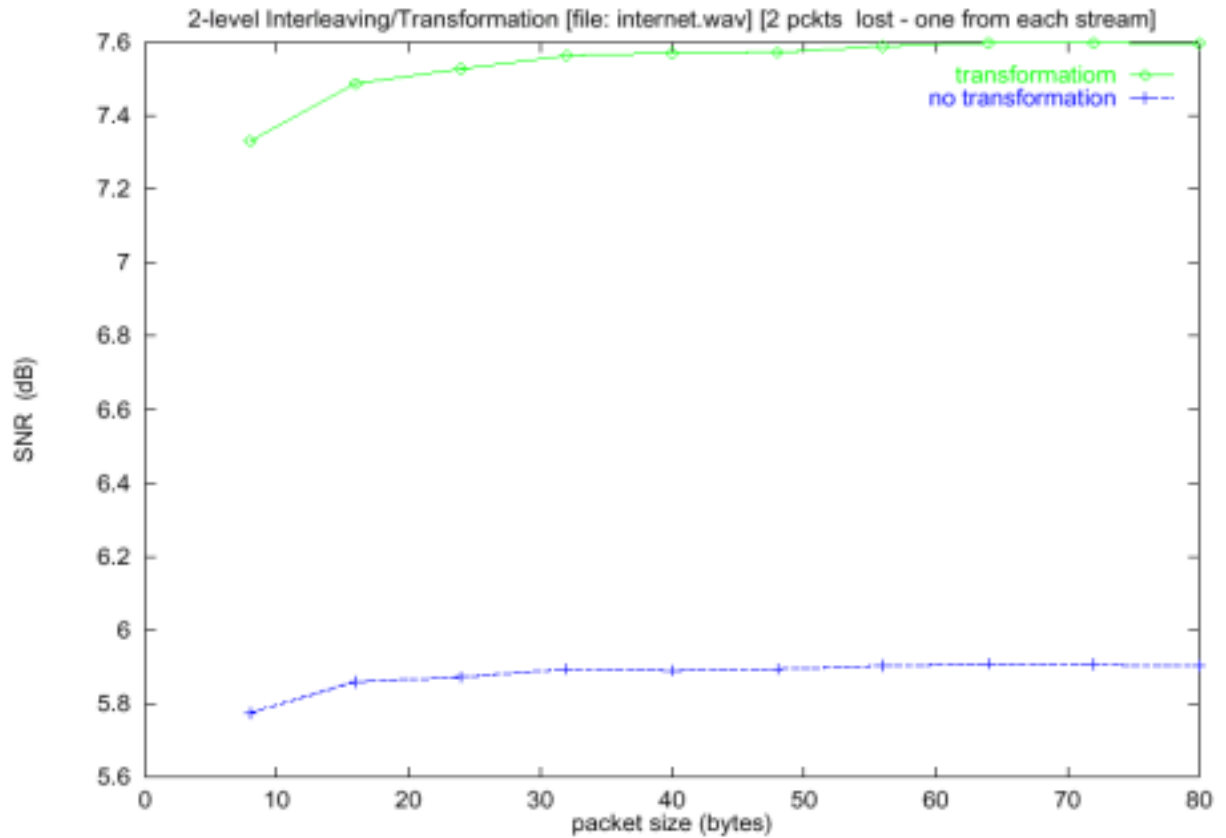


Εικόνα 3.4.9 Απώλεια 2 πακέτων ένα από κάθε ροή(*computers.wav*)

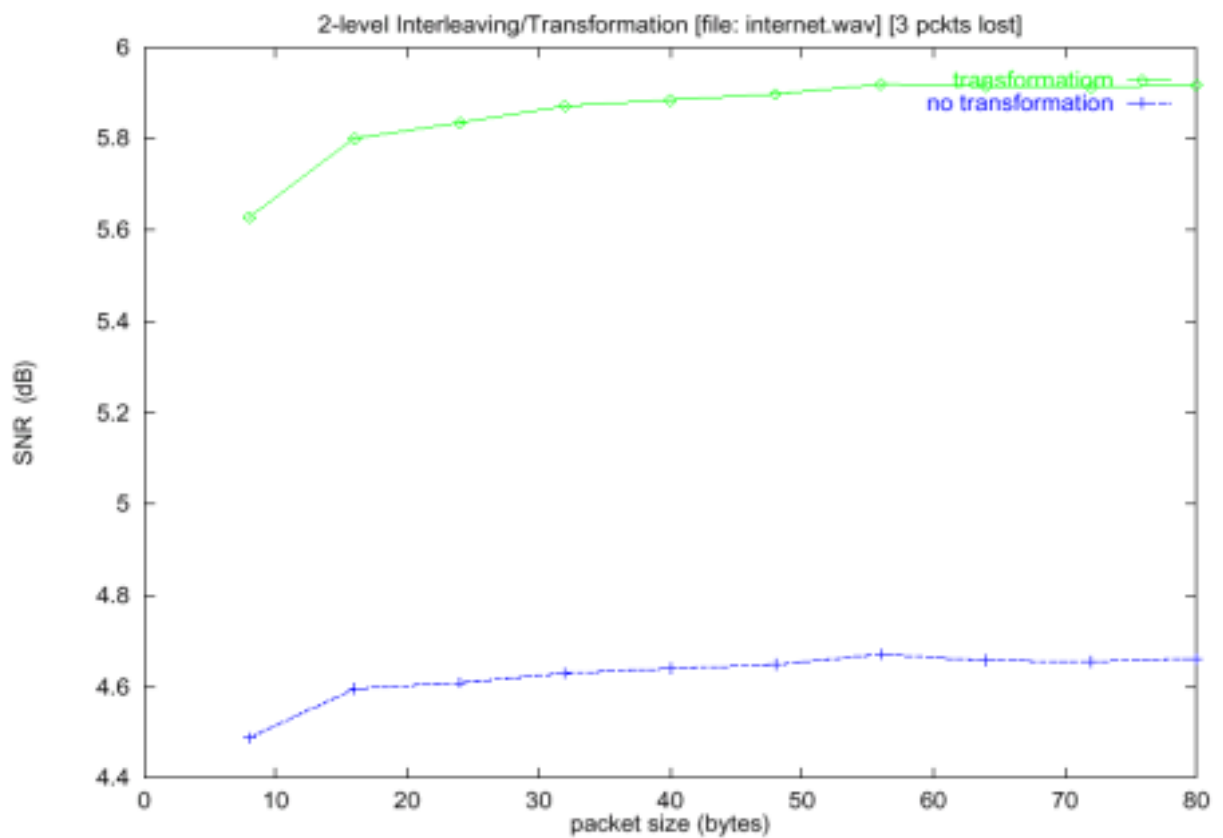


Εικόνα 3.4.10 Απώλεια 3 πακέτων (*computers.wav*)

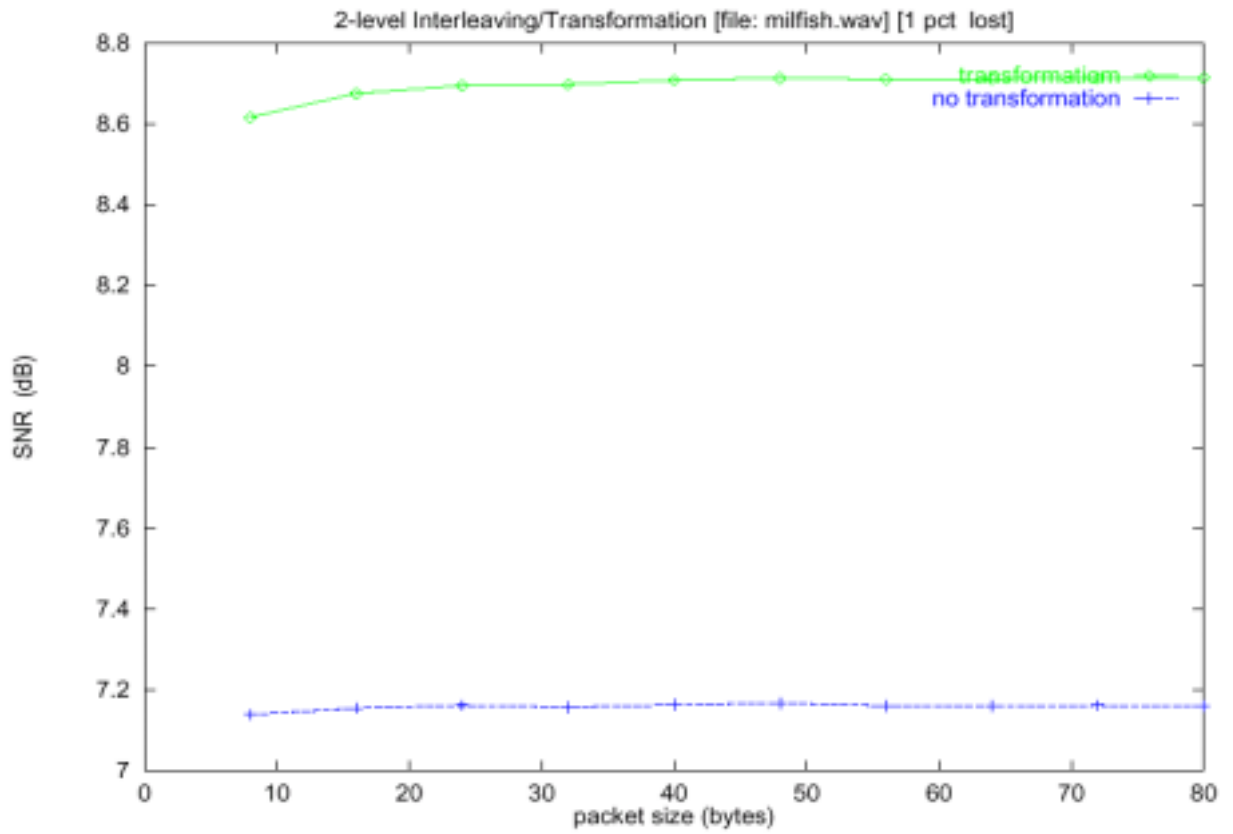
Εικόνα 3.4.11 Απώλεια 1 πακέτου (*internet.wav*)Εικόνα 3.4.12 Απώλεια 2 πακέτων της ίδιας ροής (*internet.wav*)



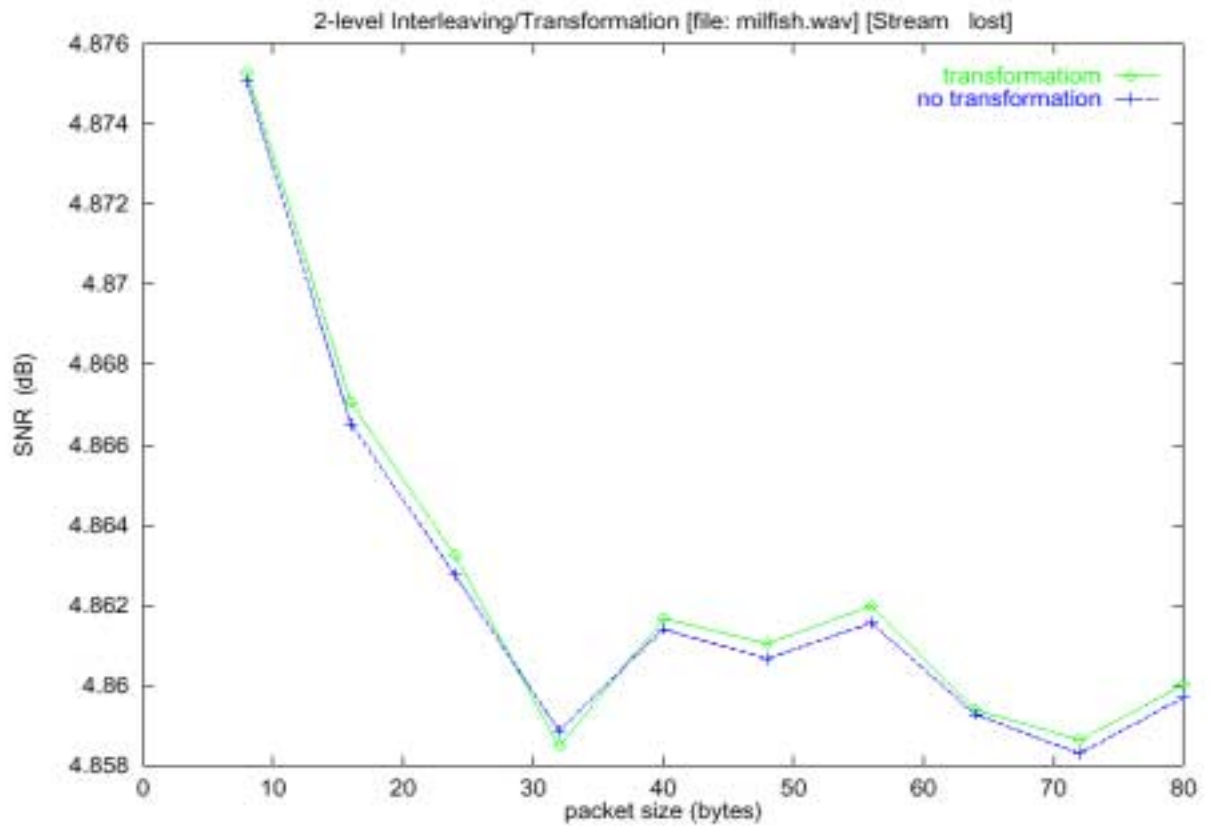
Εικόνα 3.4.13 Απώλεια 2 πακέτων διαφορετικών ροών (*internet.wav*)



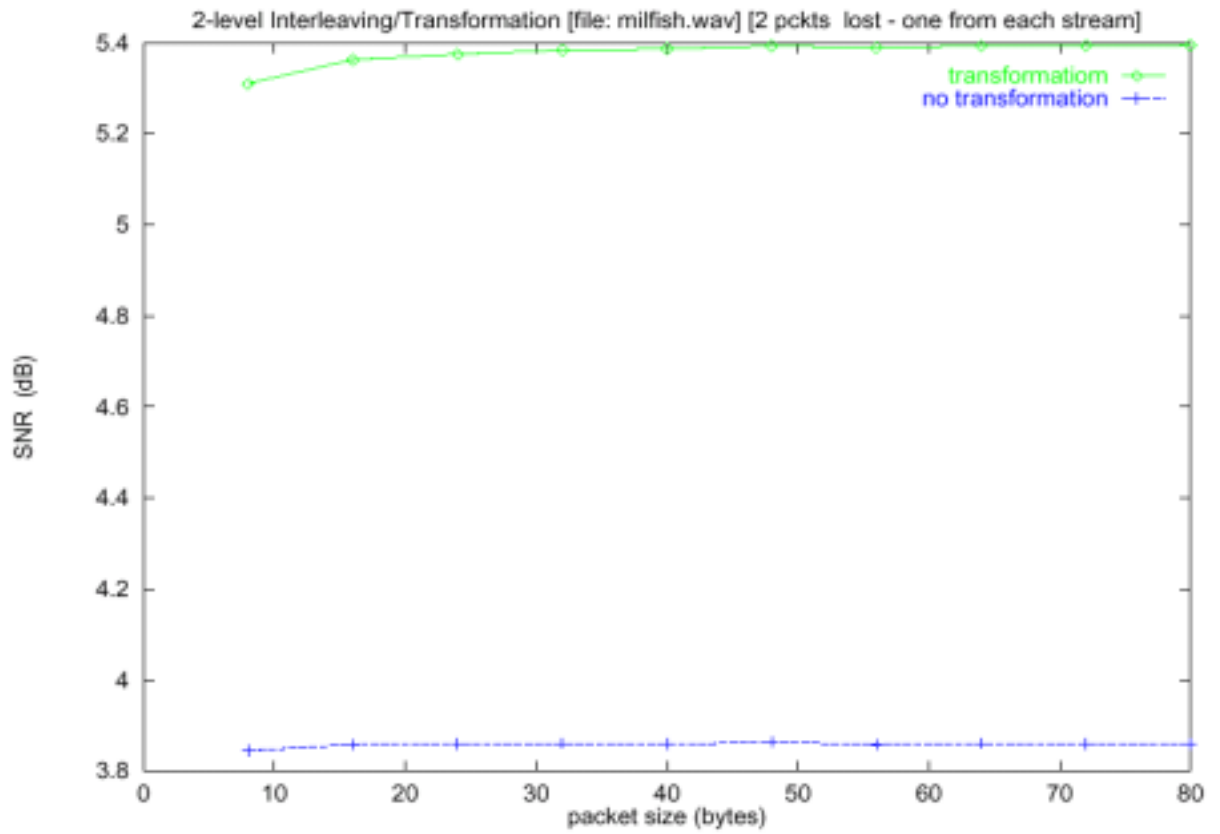
Εικόνα 3.4.14 Απώλεια 3 πακέτων (*internet.wav*)



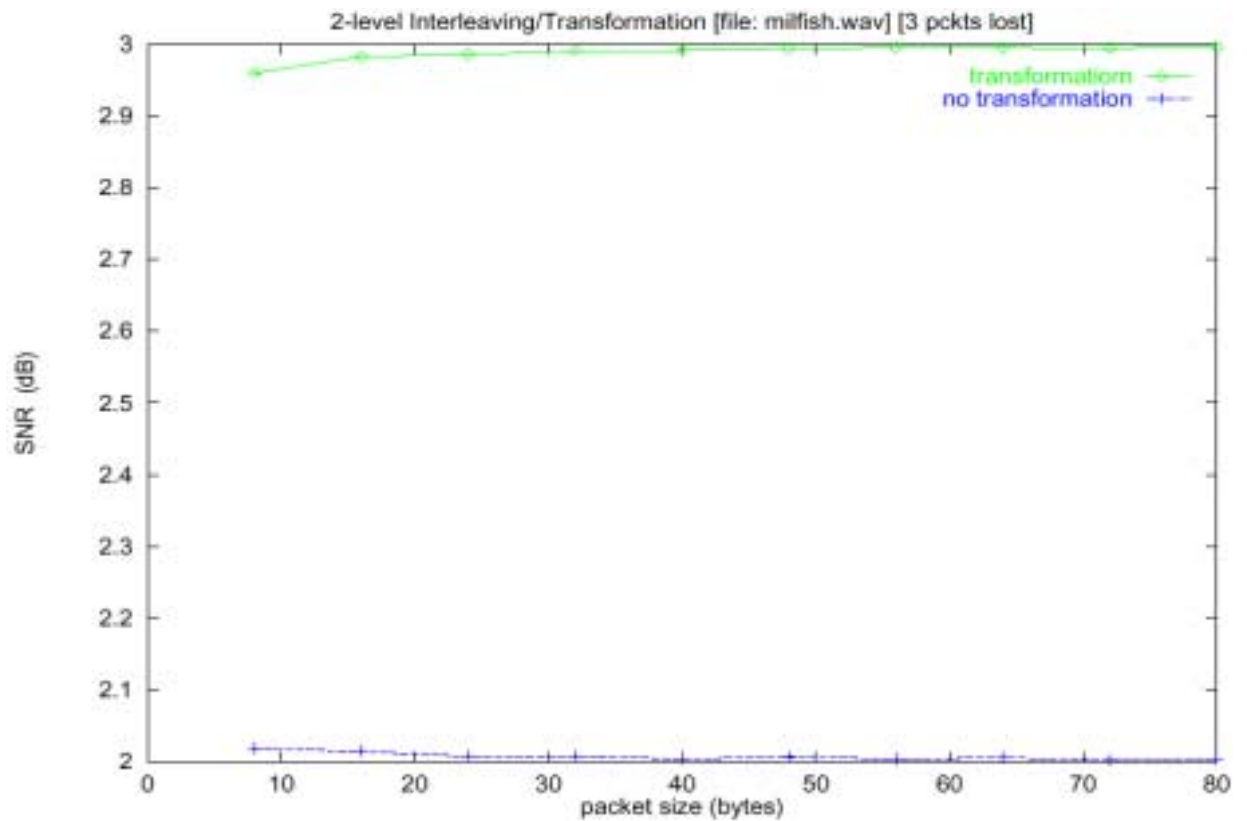
Εικόνα 3.4.15 Απώλεια 1 πακέτου(milfish.wav)



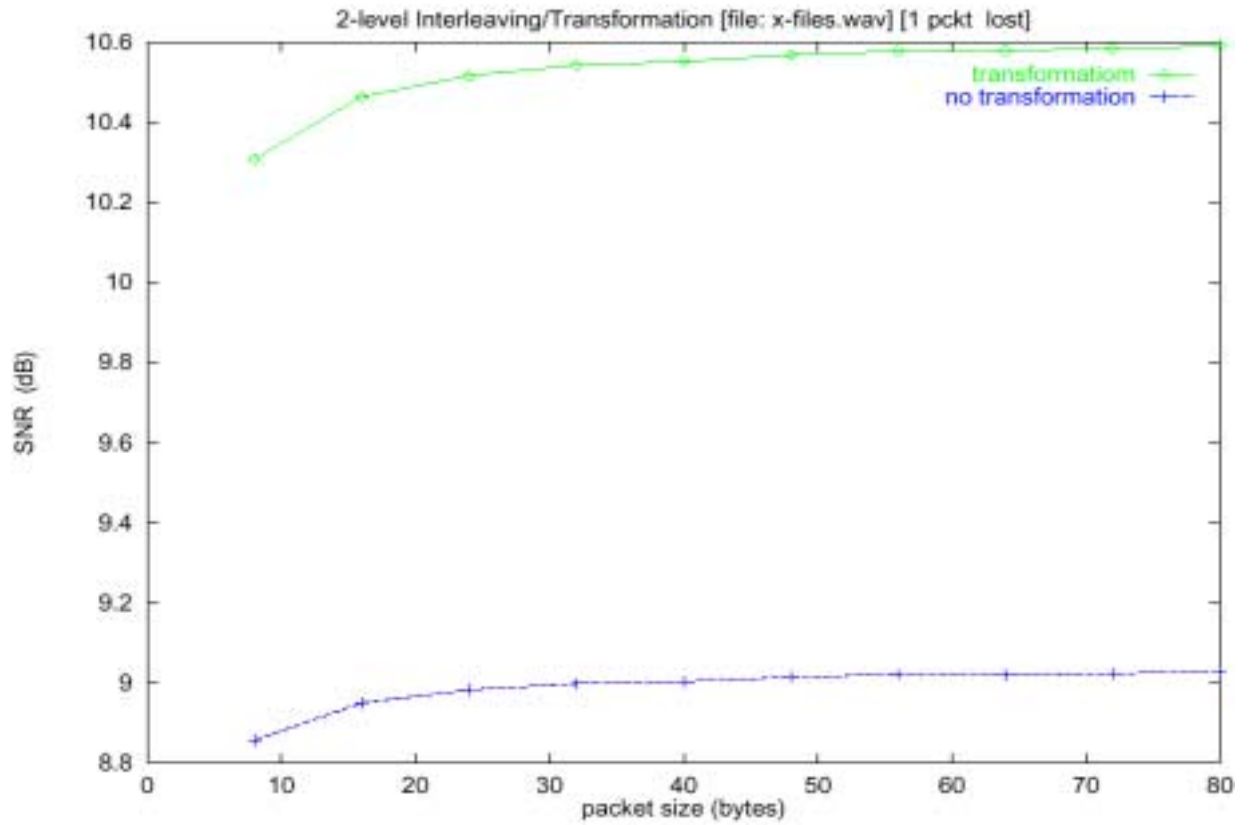
Εικόνα 3.4.16 Απώλεια 2 πακέτων της ίδιας ροής (milfish.wav)



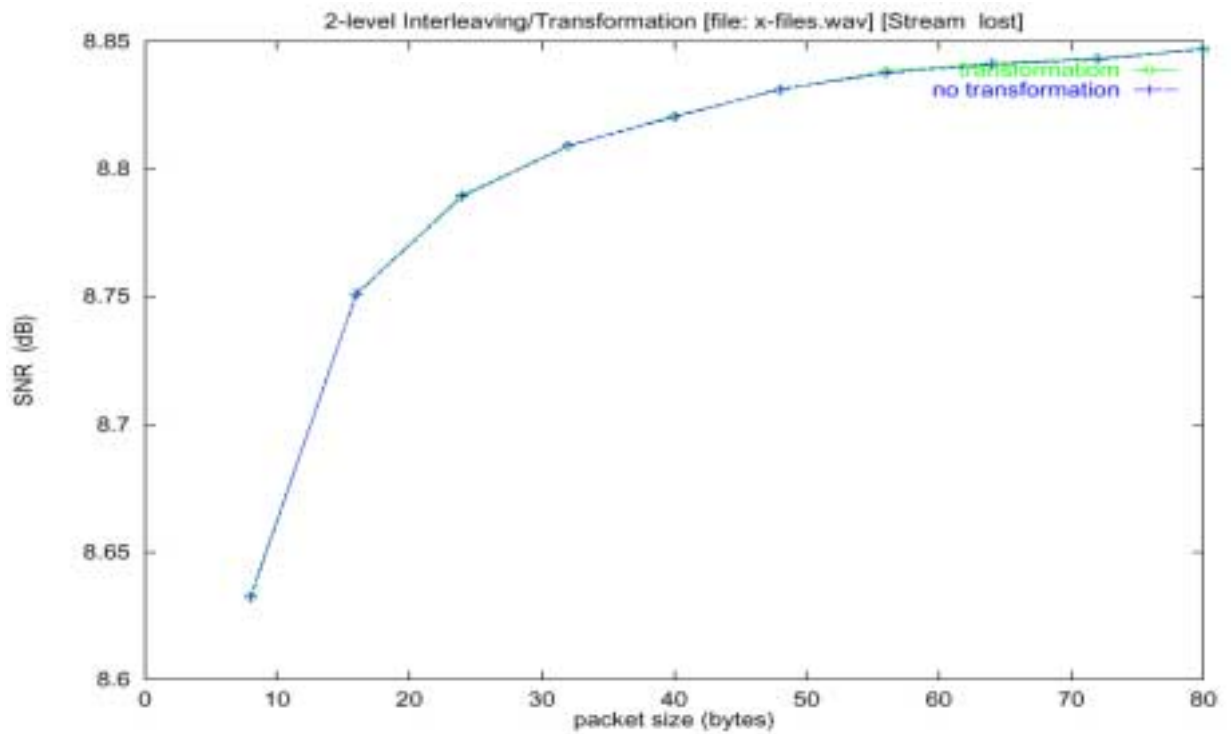
Εικόνα 3.4.17 Απώλεια 2 πακέτων διαφορετικών ροών (milfish.wav)



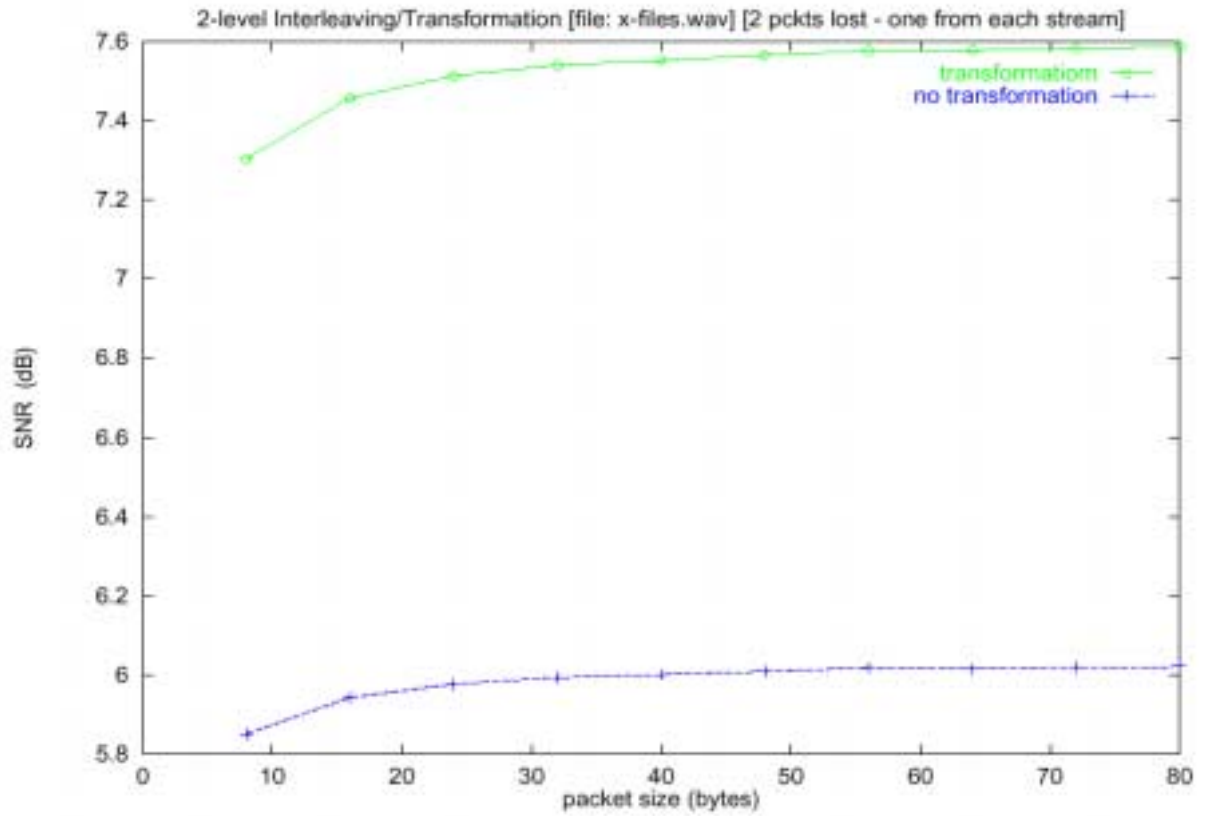
Εικόνα 3.4.18 Απώλεια 3 πακέτων (milfish.wav)



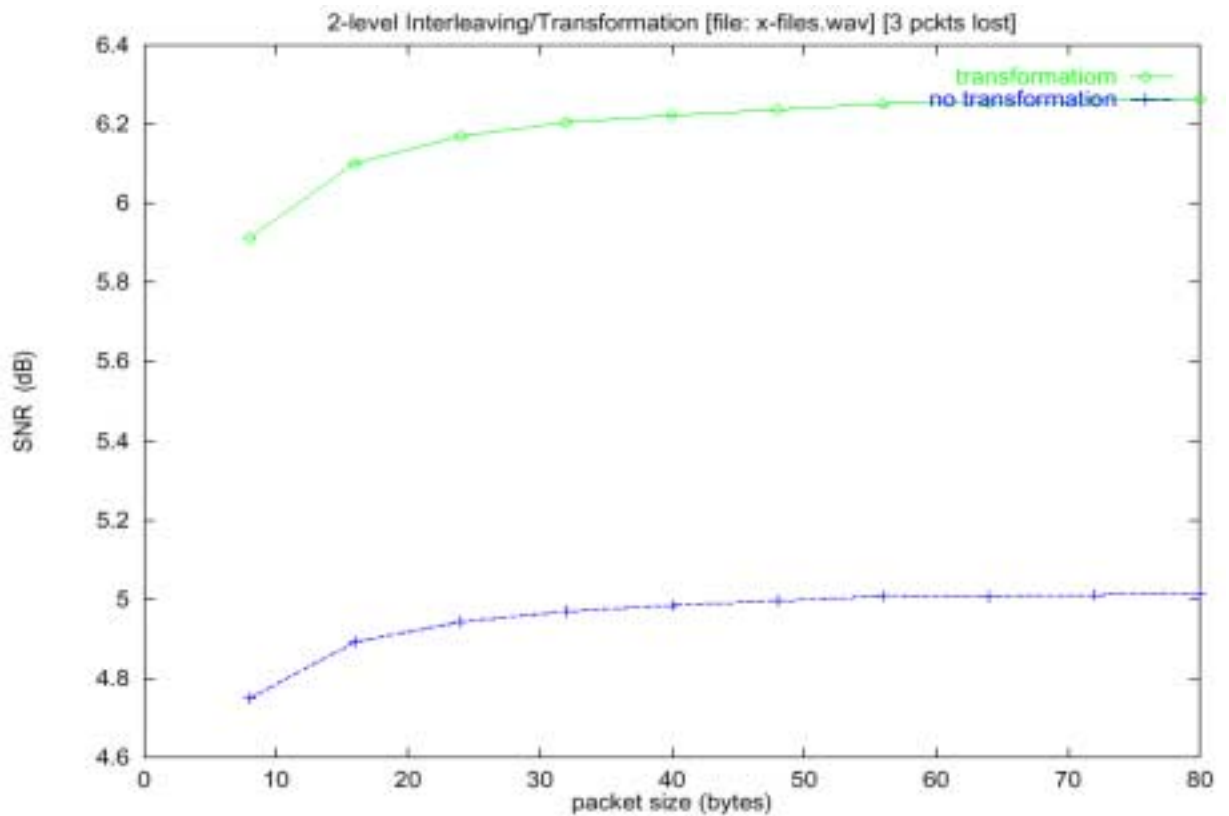
Εικόνα 3.4.19 Απόλεια 1 πακέτου (x-files.wav)



Εικόνα 3.4.20 Απόλεια 2 πακέτων της ίδιας ροής (x-files.wav)



Εικόνα 3.4.21 Απώλεια 2 πακέτων διαφορετικών ροών(x-files.wav)



Εικόνα 3.4.22 Απώλεια 3 πακέτων (x-files.wav)

3.5 Ανακατασκευή Gauss με χρήση μετασχηματισμών και διαφύλλωσης

Σε αυτήν την παράγραφο θα εξεταστεί μια παραλλαγή της τεχνικής που παρουσιάστηκε εκτενώς στην προηγούμενη παράγραφο (§3.4). Ο παραλήπτης για την ανακατασκευή χαμένων δειγμάτων κάνει χρήση μιας μεθόδου παρεμβολής με την ονομασία Gaussian interpolation [46]. Ουσιαστικά δανειζόμαστε αυτήν την τεχνική από τον χώρο της επεξεργασίας εικόνων. Βασικό της χαρακτηριστικό αποτελεί το γεγονός ότι για την ανακατασκευή ενός χαμένου δείγματος δεν συνεισφέρουν μόνο τα δυο γειτονικά δείγματα, αλλά τα τέσσερα γειτονικά.

3.5.1 Ανακατασκευή Gauss με χρήση μετασχηματισμών και διαφύλλωσης ενός επιπέδου

Για να γίνουμε πιο συγκεκριμένοι, ας υποθέσουμε ότι το αρχικό σήμα αποτελείται από τα δείγματα x_0, x_1, x_2, \dots και ότι το κάθε πακέτο έχει μέγεθος N δειγμάτων. Εφαρμόζοντας διαφύλλωση σε δυο μέρη, στο πακέτο P_1 θα τοποθετηθούν τα άρτια δείγματα x_0, x_2, x_{2N-2} , ενώ στο πακέτο P_2 τα περιττά δείγματα $x_1, x_3, \dots, x_{2N-1}$. Στην συνέχεια ο αποστολέας μεταδίδει τα πακέτα P_1 και P_2 . Ο παραλήπτης από την στιγμή που θα λάβει και τα δύο πακέτα εφαρμόζει αποδιαφύλλωση και ανακτά το αρχικό σήμα. Στην περίπτωση όμως που σημειωθεί κάποιο λάθος, ο παραλήπτης ανακατασκευάζει τα χαμένα δείγματα ως εξής:

$$\hat{x}_i = bx_{i-3} + ax_{i-1} + ax_{i+1} + bx_{i+3} \quad (3.12)$$

Ουσιαστικά θέτοντας $b=0$ και $a=0.5$, καταλήγουμε στην μέθοδο ανακατασκευής μέσου όρου. Το ζητούμενο στην σχέση (3.12) είναι οι τιμές των a και b έτσι ώστε η ανακατασκευή να γίνεται κατά τρόπο βέλτιστο. Σύμφωνα με την θεωρία, για επιτύχουμε την καλύτερη δυνατή ποιότητα ανακατασκευής τα a, b πρέπει να ακολουθούν την κατανομή Gauss. Έτσι ακολουθώντας την διαδικασία δημιουργίας πυρήνων Gauss [46] καταλήγουμε στις τιμές των a, b : όπου $a = 0.4088$ και $b = 0.0912$. Στο Παράρτημα Β αναφέρεται λεπτομερώς αυτή η διαδικασία.

Όπως και στο προηγούμενο κεφάλαιο, έτσι και εδώ ακολουθώντας ανάλογη διαδικασία θα υπολογίσουμε τους αντίστοιχους πίνακες μετασχηματισμών ώστε η ανακατασκευή στον να γίνεται κατά τρόπο βέλτιστο.

3.5.1.1 Απώλεια ενός πακέτου σε ένα ζεύγος διαφύλλωσης

Έστω $\vec{x} = x_0, x_1, \dots, x_{2N-1}$ το αρχικό σήμα φωνής. Ο αποστολέας εφαρμόζει σε αυτά έναν μετασχηματισμό \mathbf{T} και παράγει τα $\vec{y} = y_0, y_1, \dots, y_{2N-1}$. Στην συνέχεια τα μετασχηματισμένα δεδομένα διαφυλλώνονται και μεταδίδονται στον παραλήπτη. Μπορούμε να υποθέσουμε ότι μόνο όλα τα άρτια δείγματα $\vec{y}_{even} = y_0, y_2, \dots, y_{2N-2}$ παραδόθηκαν επιτυχώς.

Έτσι ο παραλήπτης εφαρμόζοντας την μέθοδο ανακατασκευής Gauss, παράγει το σήμα $\hat{y} = \hat{y}_0, \hat{y}_1, \dots, \hat{y}_{2N-1}$ ως εξής:

$$\hat{y}_i = \begin{cases} y_i & i = \text{any even} \\ \frac{y_0 + y_2}{2} & i = 1 \\ \frac{y_{2N-2}}{2} & i = 2N-1 \\ \frac{y_{2N-4} + y_{2N-2}}{2} & i = 2N-3 \\ by_{i-3} + ay_{i-1} + ay_{i+1} + by_{i+3} & i = \text{other odd} \end{cases} \quad (3.13)$$

Και σαν λάθος ανακατασκευής RE (Reconstruction Error) ορίζουμε το:

$$RE = \sum_{i=0}^{2N-1} (x_i - y_i)^2 \quad (3.14)$$

Για να ελαχιστοποιήσουμε το λάθος ανακατασκευής, πρέπει να υπολογιστεί το y_i έτσι ώστε να ικανοποιούνται οι παρακάτω εξισώσεις, για κάθε άρτιο i .

$$\frac{\partial RE}{\partial y_i} = 0, i = 0, 2, 4, 6, \dots, 2N-2 \quad (3.15)$$

Αντικαθιστώντας το RE στην σχέση (3.15), χρησιμοποιώντας την σχέση (3.14) καταλήγουμε στον μετασχηματισμό πινάκων:

$$\begin{pmatrix} y_0 \\ y_2 \\ \vdots \\ y_{2N-4} \\ y_{2N-2} \end{pmatrix} = \mathbf{T} \times \begin{pmatrix} x_0 \\ x_1 \\ \vdots \\ x_{2N-2} \\ x_{2N-1} \end{pmatrix} \quad (3.16)$$

όπου

$$\mathbf{T} = \mathbf{A}^{-1} \mathbf{B}$$

και οι πίνακες \mathbf{A} , \mathbf{B} παράγονται από τις παρακάτω εξισώσεις:

$$\frac{\partial RE}{\partial y_i} = 0 \Leftrightarrow \begin{cases} i = 0 \\ i = 2 \\ i = 4 \\ i = \text{even} \& \lt (2N-6) \Leftrightarrow \\ i = 2N-6 \\ i = 2N-4 \\ i = 2N-2 \end{cases} \quad (3.17)$$

$$\left\{ \begin{array}{l} (5/4 + b^2)y_0 + (1/4 + ab)y_2 + aby_4 + b^2y_6 = x_0 + \frac{x_1}{2} + bx_3 \\ (1/4 + ab)y_0 + (5/4 + a^2 + b^2)y_2 + (a^2 + ab)y_4 + 2aby_6 + b^2y_8 = \frac{x_1}{2} + x_2 + ax_3 + bx_5 \\ aby_0 + (a^2 + ab)y_2 + (2a^2 + b^2 + 1)y_4 + (2ab + a^2)y_6 + 2aby_8 + b^2y_{10} = ax_3 + ax_4 + bx_5 \\ b^2y_{i-6} + 2aby_{i-4} + (2ab + a^2)y_{i-2} + (2a^2 + 2b^2 + 1)y_i + (2ab + a^2)y_{i+2} + 2aby_{i+4} + b^2y_{i+6} = bx_{i-3} + ax_{i-1} + ax_{i+1} + bx_{i+3} \\ b^2y_{i-6} + 2aby_{i-4} + (2ab + a^2)y_{i-2} + (2a^2 + b^2 + 1)y_i + (ab + a^2)y_{i+2} + aby_{i+4} = bx_{i-3} + ax_{i-1} + ax_{i+1} \\ b^2y_{i-6} + 2aby_{i-4} + (ab + a^2)y_{i-2} + (a^2 + b^2 + 5/4)y_i + (ab + 1/4)y_{i+2} = bx_{i-3} + ax_{i-1} + x_i + \frac{x_{i+1}}{2} \\ b^2y_{i-6} + aby_{i-4} + (ab + 1/4)y_{i-2} + (a^2 + b^2 + 3/2)y_i = bx_{i-3} + \frac{x_{i-1}}{2} + x_i + \frac{x_{i+1}}{2} \end{array} \right.$$

και αντίστοιχα για την περίπτωση κατά την οποία χάνονται τα άρτια:

$$\frac{\partial RE}{\partial y_i} = 0 \Leftrightarrow \left\{ \begin{array}{l} i = 1 \\ i = 3 \\ i = 5 \\ i = \text{odd} \& \lt (2N - 5) \Leftrightarrow \\ i = 2N - 5 \\ i = 2N - 3 \\ i = 2N - 1 \end{array} \right. \quad (3.18)$$

$$\left\{ \begin{array}{l} (3/2 + b^2)y_1 + (1/4 + ab)y_3 + aby_5 + b^2y_7 = \frac{x_0}{2} + x_1 + \frac{x_2}{2} + bx_4 \\ (1/4 + ab)y_1 + (5/4 + a^2 + b^2)y_3 + (a^2 + ab)y_5 + 2aby_7 + b^2y_9 = \frac{x_2}{2} + x_3 + ax_4 + bx_6 \\ aby_1 + (a^2 + ab)y_3 + (2a^2 + b^2 + 1)y_5 + (2ab + a^2)y_7 + 2aby_9 + b^2y_{11} = ax_4 + x_5 + ax_6 + bx_8 \\ b^2y_{i-6} + 2aby_{i-4} + (2ab + a^2)y_{i-2} + (2a^2 + 2b^2 + 1)y_i + (2ab + a^2)y_{i+2} + 2aby_{i+4} + b^2y_{i+6} = bx_{i-3} + ax_{i-1} + ax_{i+1} + bx_{i+3} \\ b^2y_{i-6} + 2aby_{i-4} + (2ab + a^2)y_{i-2} + (2a^2 + b^2 + 1)y_i + (ab + a^2)y_{i+2} + aby_{i+4} = bx_{i-3} + ax_{i-1} + x_i + ax_{i+1} \\ b^2y_{i-6} + 2aby_{i-4} + (ab + a^2)y_{i-2} + (a^2 + b^2 + 5/4)y_i + (ab + 1/4)y_{i+2} = bx_{i-3} + ax_{i-1} + x_i + \frac{x_{i+1}}{2} \\ b^2y_{i-6} + aby_{i-4} + (ab + 1/4)y_{i-2} + (b^2 + 5/4)y_i = bx_{i-3} + \frac{x_{i-1}}{2} + x_i \end{array} \right.$$

Ακολουθώντας την διαδικασία του Παραρτήματος Α, μπορούμε εύκολα να δούμε ότι οι παραπάνω πίνακες αντιστρέφονται. Έτσι αυτήν την στιγμή ο μετασχηματισμός από τα \bar{x} στα \bar{y} είναι πλήρης. Ο αποστολέας είναι σε θέση να στέλνει το μετασχηματισμένο σήμα φωνής μέσω του διαδικτύου και από την στιγμή που συμβεί κάποιο μεμονωμένο σφάλμα μετάδοσης, ο παραλήπτης μπορεί να ανακατασκευάσει το χαμένο πακέτο με βέλτιστο τρόπο.

3.5.1.2 Πλήρης παράδοση ζεύγους διαφύλλωσης

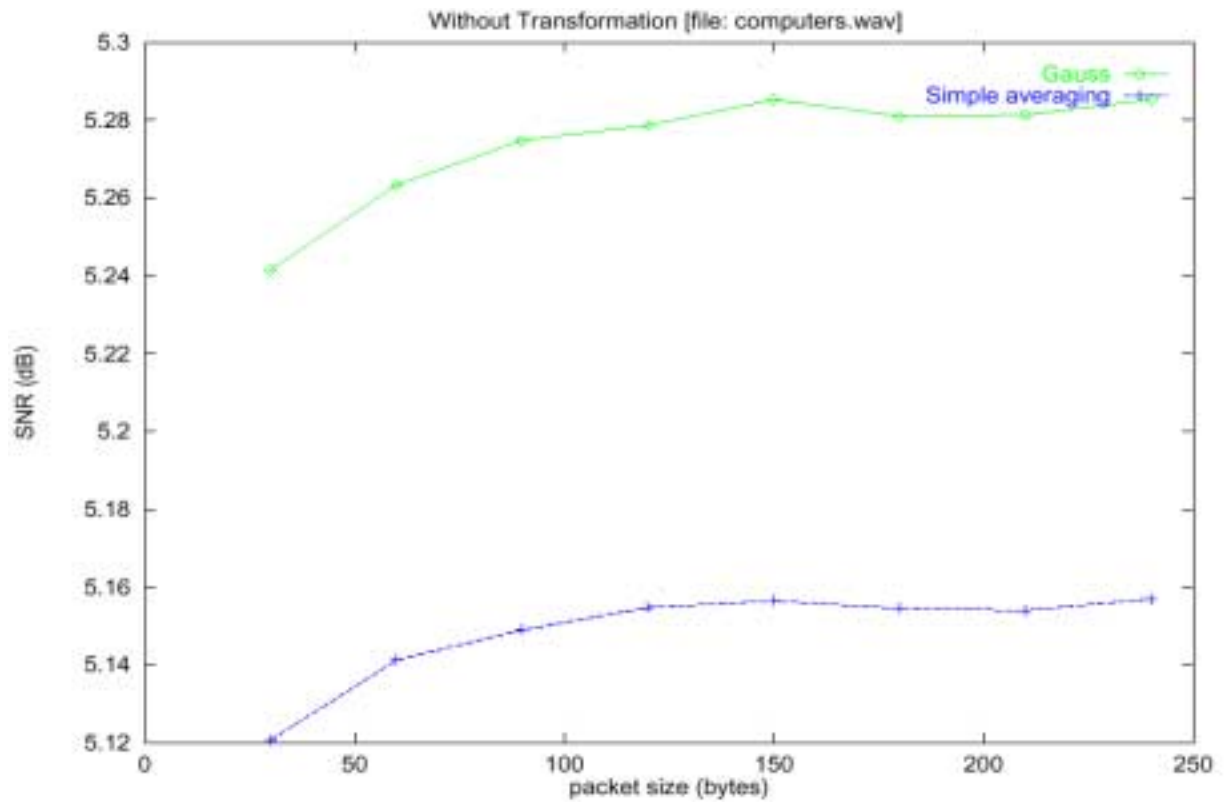
Συνδυάζοντας τις σχέσεις (3.17) και (3.18) κατά τρόπο ανάλογο με αυτόν που αναφέρθηκε στην παράγραφο §3.4.1.2, ο παραλήπτης είναι σε θέση να ανακτήσει επακριβώς το αρχικό σήμα φωνής, δεδομένου βέβαια ότι έχουν ληφθεί τα πακέτα του ζεύγους διαφύλλωσης χωρίς σφάλματα.

3.5.2 Πειραματικά αποτελέσματα

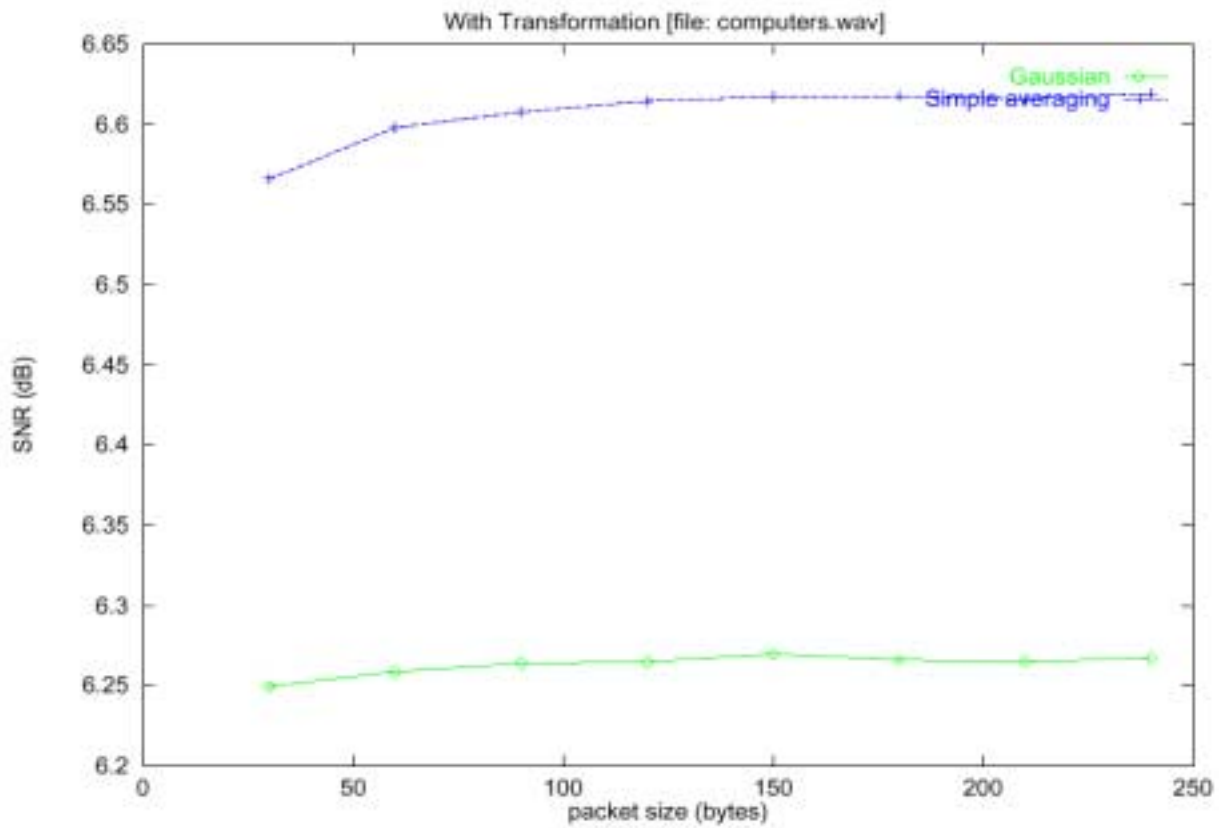
Ο προσομοιωτής που είχε υλοποιηθεί για την μέτρηση απόδοσης της τεχνικής ανακατασκευής μέσου όρου, επεκτάθηκε ώστε να υποστηρίζει και την μέθοδο ανακατασκευής Gauss. Όπως και προηγουμένως οι δυο τεχνικές εφαρμόστηκαν πάνω στην ίδια ομάδα αρχείων ήχου (βλ. πίνακα 3.4.1) και κριτήριο ποιότητας ανακατασκευής αποτελεί ξανά ο σηματοθορυβικός λόγος SNR (βλ. σχέση 3.1). Επίσης για λόγους απλότητας θεωρήσαμε ότι σε κάθε ζεύγος διαφύλλωσης χάνεται ακριβώς ένα πακέτο (πάντα το άρτιο).

Τέλος πρέπει να επισημανθεί ότι οι συγκρίσεις μεταξύ των δυο μεθόδων γίνονται κατά αντιστοιχία χρήσης ή όχι του ανάλογου μετασχηματισμού. Δηλαδή, πρώτα συγκρίνουμε την απόδοση κάθε τεχνικής χωρίς την εφαρμογή μετασχηματισμού και στην συνέχεια εκτελούνται πάλι τα αντίστοιχα πειράματα αυτή την φορά όμως με χρήση των ανάλογων μετασχηματισμών.

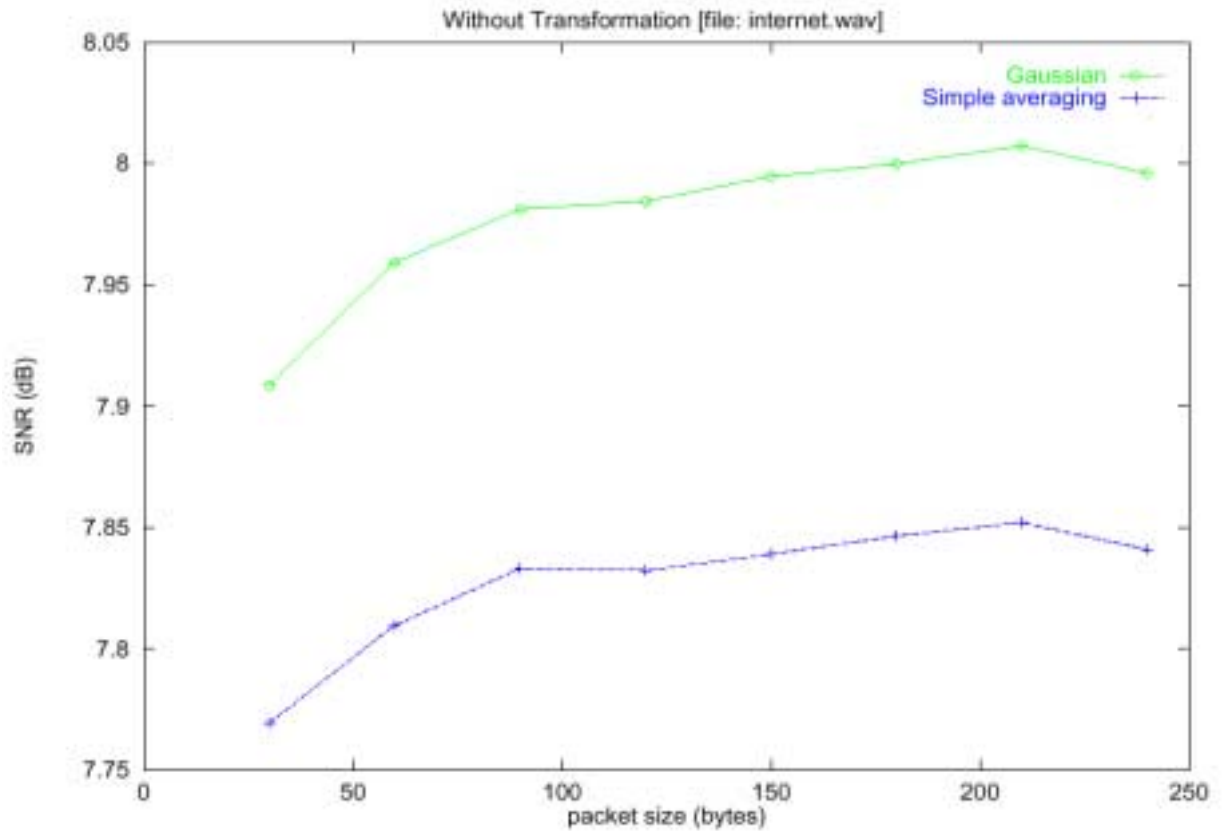
Στις εικόνες 3.5.1 έως και 3.5.8, παρουσιάζονται τα αποτελέσματα αυτών των μετρήσεων από τις οποίες μπορούν να εξαχθούν χρήσιμα συμπεράσματα: Αρχικά παρατηρούμε την τεχνική ανακατασκευής μέσου όρου να υστερεί οριακά (περίπου κατά 0.05 dB) της μεθόδου ανακατασκευής Gauss, χωρίς την χρήση μετασχηματισμού. Όταν όμως γίνεται χρήση των αντίστοιχων μετασχηματισμών η τεχνική ανακατασκευής Gauss είναι αυτή που υστερεί και μάλιστα σε αρκετά μεγάλο βαθμό (περίπου κατά 1 dB). Διαισθητικά θα περιμέναμε η μέθοδος αυτή να υπερτερεί, αφού για την ανακατασκευή ενός χαμένου δείγματος συμμετέχουν τέσσερα γειτονικά δείγματα και όχι δυο. Όμως έχοντας εφαρμόσει διαφύλλωση σε δυο μέρη, στην ανακατασκευή του δείγματος x_i συμμετέχουν τα αμέσως γειτονικά δείγματα x_{i-1} και x_{i+1} , αλλά όμως και τα x_{i-3} , x_{i+3} τα οποία όπως αποδείχθηκε ότι δεν έχουν και τόσο υψηλό βαθμό συσχέτισης με το χαμένο δείγμα.



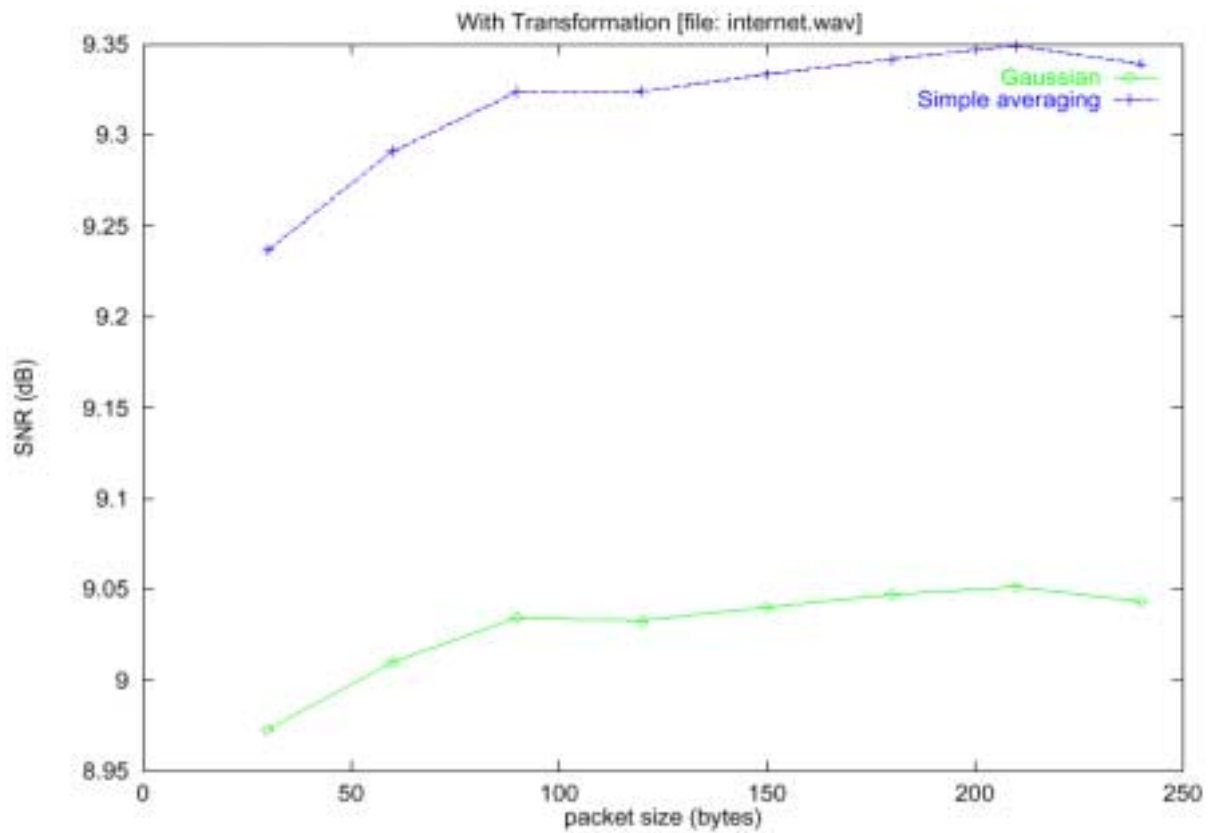
Εικόνα 3.5.1 Σύγκριση τεχνικών χωρίς χρήση μετασχηματισμού(*computers.wav*)



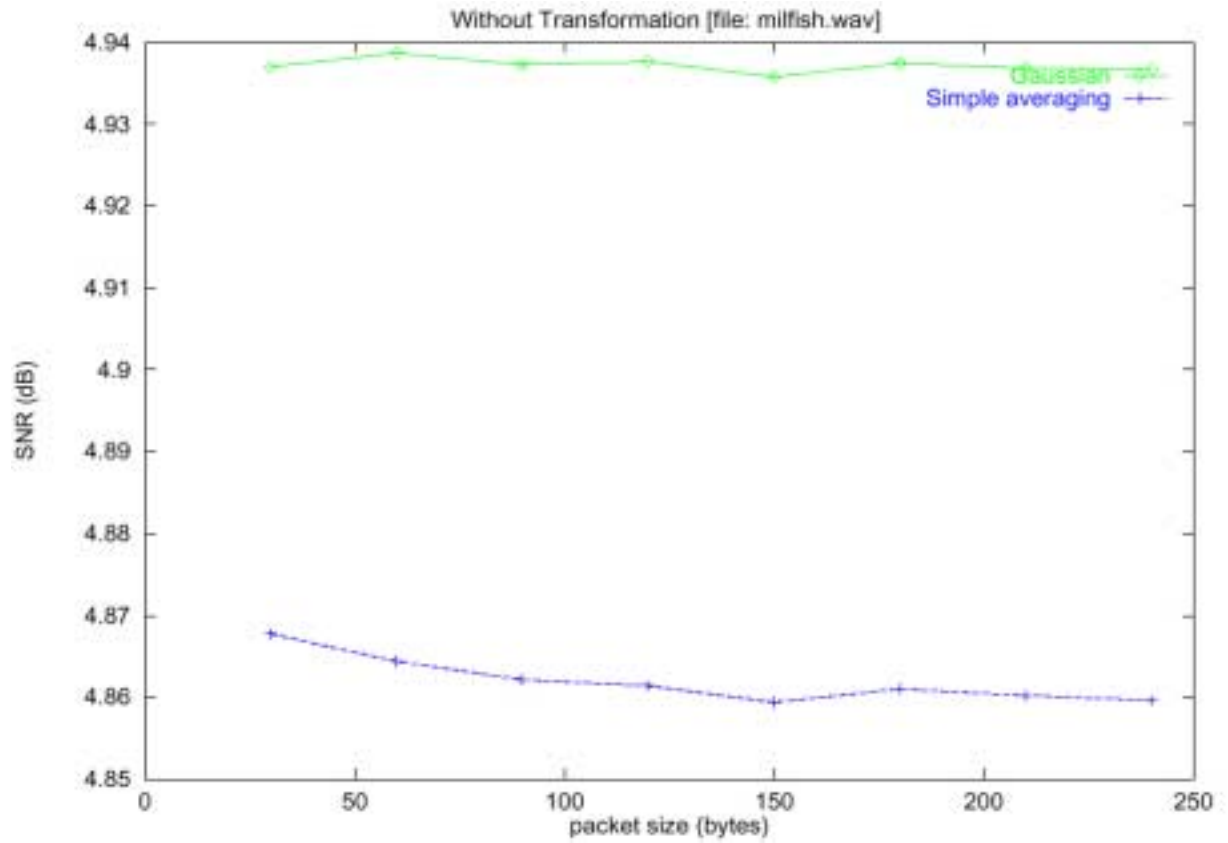
Εικόνα 3.5.2 Σύγκριση τεχνικών με χρήση μετασχηματισμού(*computers.wav*)



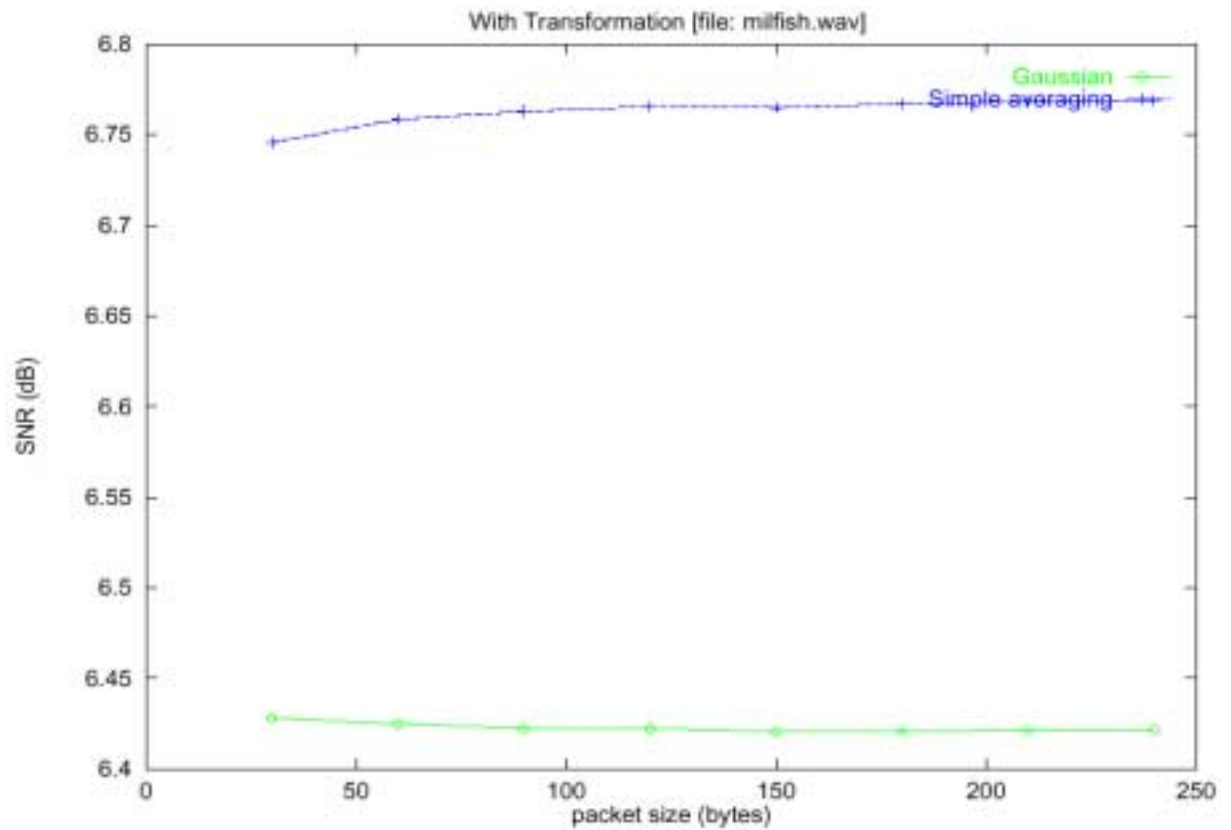
Εικόνα 3.5.3 Σύγκριση τεχνικών χωρίς χρήση μετασχηματισμού(*internet.wav*)



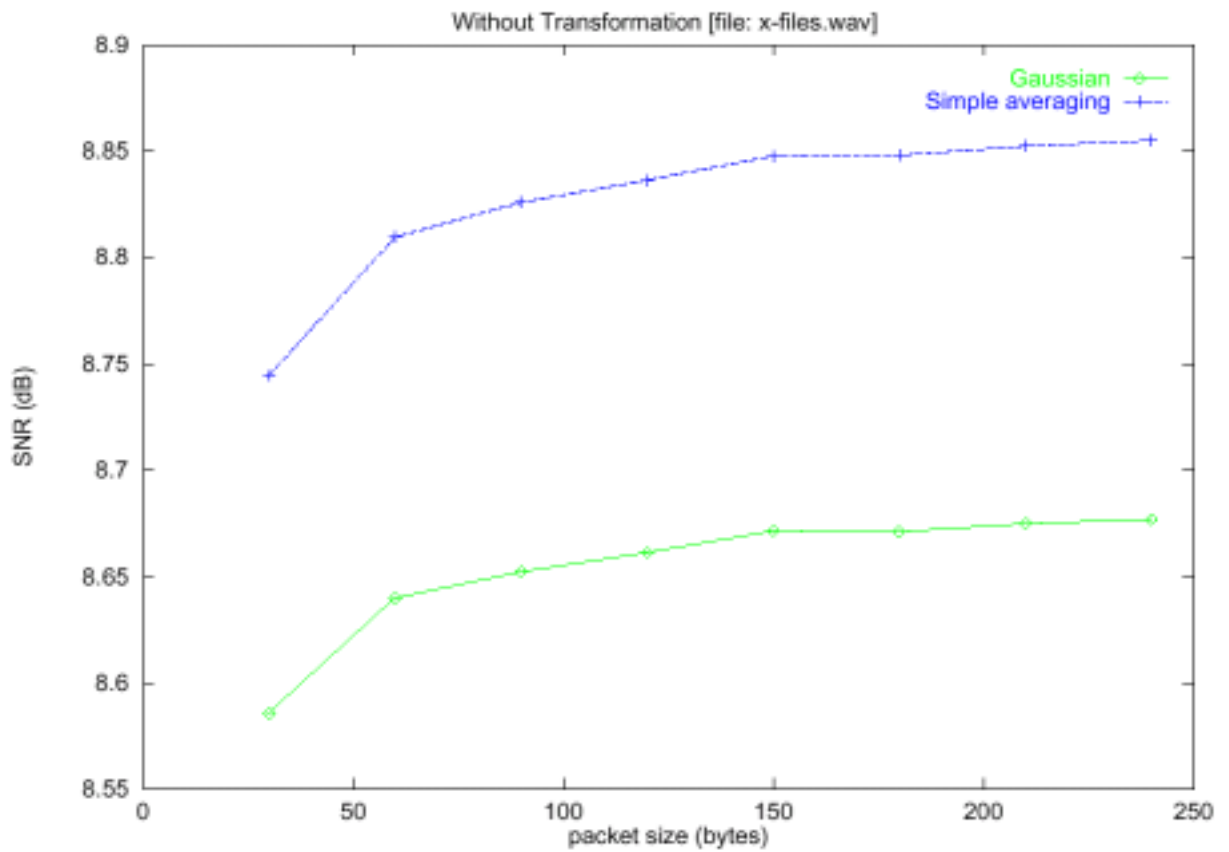
Εικόνα 3.5.4 Σύγκριση τεχνικών με χρήση μετασχηματισμού(*internet.wav*)



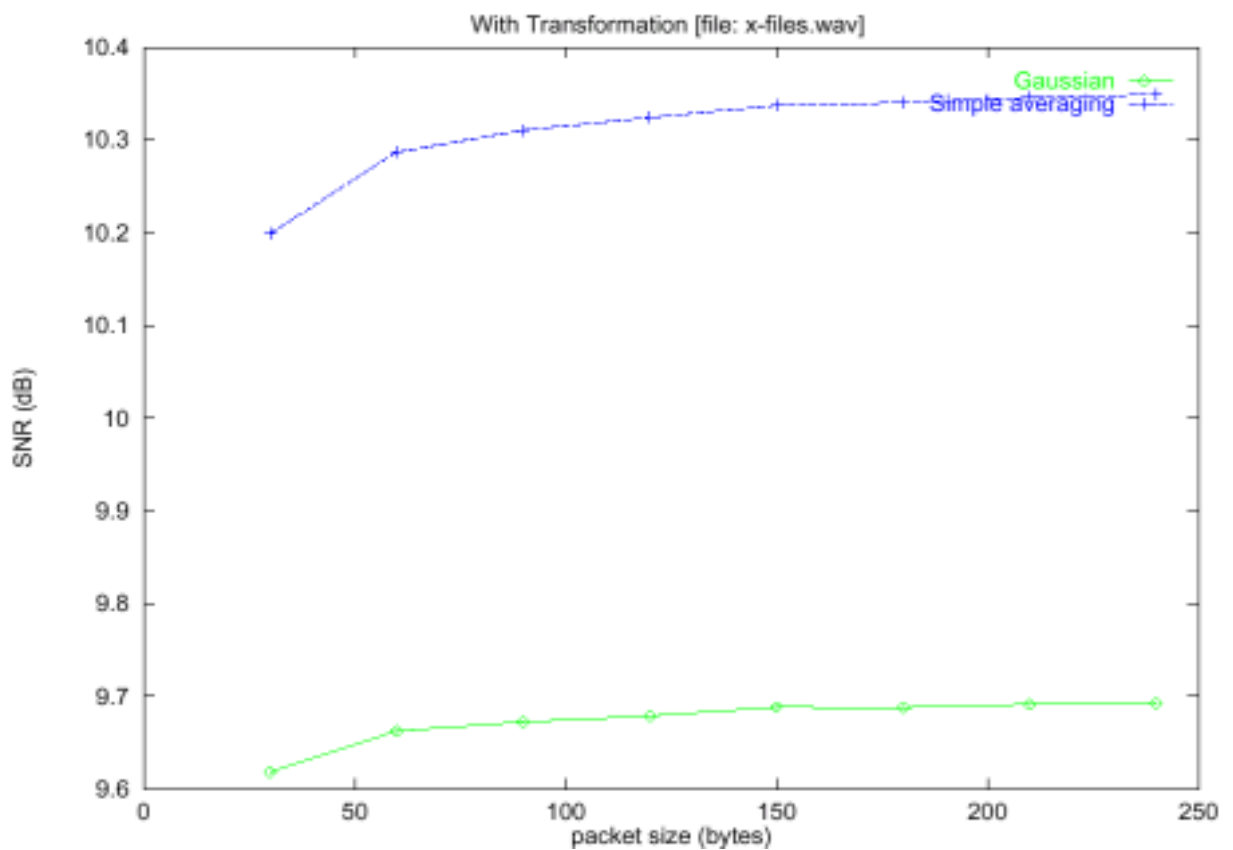
Εικόνα 3.5.5 Σύγκριση τεχνικών χωρίς χρήση μετασχηματισμού(milfish.wav)



Εικόνα 3.5.6 Σύγκριση τεχνικών με χρήση μετασχηματισμού(milfish.wav)



Εικόνα 3.5.7 Σύγκριση τεχνικών χωρίς χρήση μετασχηματισμού(x-files.wav)



Εικόνα 3.5.8 Σύγκριση τεχνικών με χρήση μετασχηματισμού(x-files.wav)

3.6 Ανακατασκευή μέσου όρου με χρήση διαφύλλωσης σε τρία μέρη

Στα προηγούμενα κεφάλαια ασχοληθήκαμε με την εφαρμογή διαφύλλωσης δεδομένων σε δύο μέρη, άρτια και περιττά, πετυχαίνοντας μάλιστα την βέλτιστη ανακατασκευή τους με την χρήση απαραίτητων μετασχηματισμών. Στην περίπτωση που θέλαμε να αντιμετωπίσουμε καταστάσεις εμφάνισης συσσωρευμένων λαθών (βλ. παράγραφο §3.4.2) εφαρμόζαμε διαφύλλωση σε δυο επίπεδα με αποτέλεσμα να χωρίζονται τα αρχικά δεδομένα σε τέσσερα μέρη. Σε αυτή την παράγραφο θα εξετάσουμε την χρήση διαφύλλωσης των δεδομένων σε τρία μέρη.

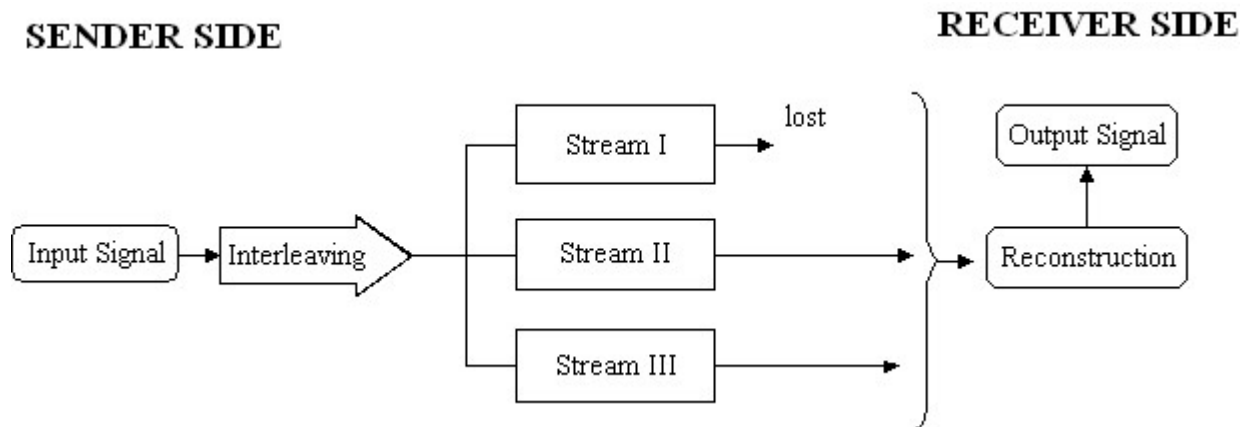
Υποθέτουμε ότι το αρχικό σήμα αποτελείται από τα δείγματα $\vec{x} = x_0, x_1, x_2, \dots$. Εφαρμόζοντας την παραπάνω διαφύλλωση, δημιουργούνται τρεις ροές:

- Stream I : x_0, x_3, x_6, \dots
- Stream II : x_1, x_4, x_7, \dots
- Stream III: x_2, x_5, x_8, \dots

Κατά αυτόν τον τρόπο ο αποστολέας κάθε φορά στέλνει στον παραλήπτη μια τριάδα πακέτων, την οποία και θα ονομάσουμε τριάδα διαφύλλωσης.

3.6.1 Απώλεια ενός πακέτου τριάδας διαφύλλωσης

Η μέθοδος που θα αναλυθεί παρακάτω μπορεί να αντιμετωπίσει ένα μόνο σφάλμα σε κάθε μετάδοση μιας τριάδας διαφύλλωσης. Πιο συγκεκριμένα, αν χαθεί ένα οποιοδήποτε πακέτο από τα τρία (βλ. εικόνα 3.6.1), ο παραλήπτης είναι πάντα σε θέση να ανακατασκευάσει κάθε χαμένο δείγμα χρησιμοποιώντας την μέθοδο ανακατασκευής μέσου όρου (βλ παράγραφο §3.4.1).



Εικόνα 3.6.1 Διαφύλλωση σε τρία μέρη και ανακατασκευή μέσου όρου

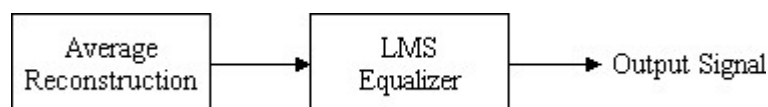
Δηλαδή για κάθε χαμένο δείγμα x_i θα υπάρχουν πάντα τα δύο γειτονικά του, x_{i+1} και x_{i-1} από τα οποία μπορεί να ανακατασκευαστεί: $\hat{x}_i = \frac{x_{i-1} + x_{i+1}}{2}$.

3.6.2 Ανακατασκευή μέσου όρου με χρήση equalizer

Η παραπάνω τεχνική διαφύλλωσης, προσφέρει και ένα άλλο πλεονέκτημα το οποίο μπορούμε να εκμεταλλευτούμε. Στην ανακατασκευή κάθε χαμένου δείγματος x_i μπορούν να συμμετέχουν εκτός των δυο γειτονικών δειγμάτων x_{i+1} , x_{i-1} και τα αμέσως επόμενα γειτονικά δείγματα x_{i+2} και x_{i-2} . Αρά θα μπορούσαμε να χρησιμοποιήσουμε την μέθοδο ανακατασκευής Gauss, η οποία όπως είδαμε είχε έστω και οριακά καλύτερη απόδοση από την αντίστοιχη τεχνική ανακατασκευής μέσου όρου. Όμως το βασικό μειονέκτημα των παραπάνω μεθόδων ανακατασκευής είναι κάθε φορά υπολογίζουν το χαμένο δείγμα με τις ίδιες τιμές στις παραμέτρους τους. Ουσιαστικά δεν λαμβάνουν υπόψιν τους τα φυσικά χαρακτηριστικά του σήματος και κατά συνέπεια εφαρμόζουν μια τεχνική ανακατασκευής στατικών παραμέτρων αγνοώντας κάθε φορά το λάθος ανακατασκευής που παράγεται.

Σε αυτήν την παράγραφο προτείνεται ένας προσαρμοστικός αλγόριθμος ανακατασκευής ο οποίος κάθε φορά προσαρμόζει τις τιμές των παραμέτρων του, με σκοπό να μειώσει το λάθος ανακατασκευής βασιζόμενος σε προηγούμενα αλλά και επόμενα γειτονικά δείγματα.

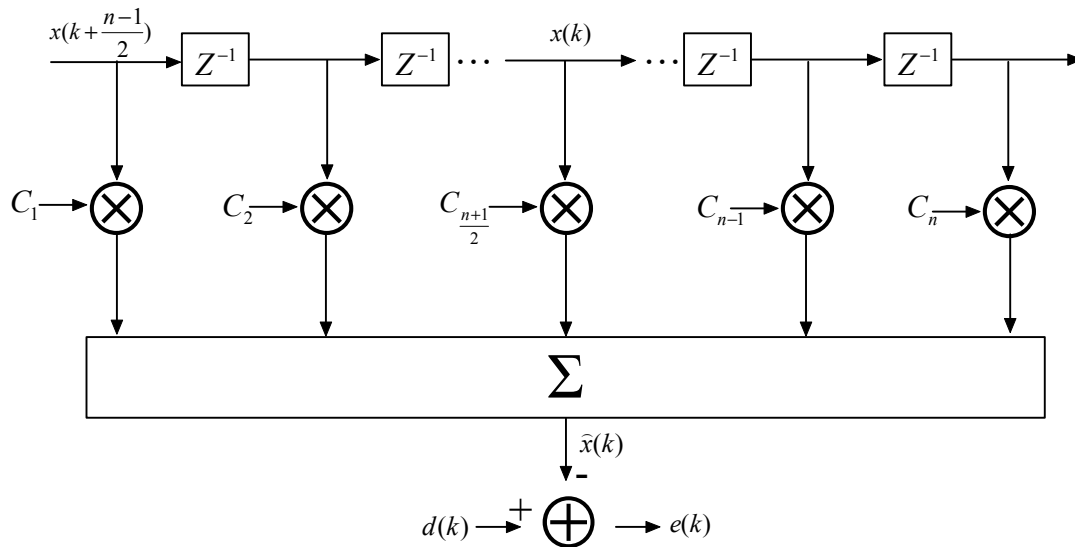
Όπως παρατηρούμε, και από το παρακάτω σχήμα (βλ. εικόνα 3.6.2), ο παραλήπτης αφού εφαρμόσει την τεχνική ανακατασκευής μέσου όρου τροφοδοτεί ένα equalizer, με το ανακατασκευασμένο σήμα. Το equalizer, είναι ένα προσαρμοστικό φίλτρο το οποίο προσπαθεί να προβλέψει την τιμή ενός δείγματος λαμβάνοντας υπόψιν προηγούμενα αλλά και επόμενα δείγματα. Υπάρχουν αρκετά είδη equalizer που χρησιμοποιούνται στον χώρο των τηλεπικοινωνιών, αλλά σε αυτή την εργασία επιλέχθηκε ο Linear Mean Square (LMS) equalizer γιατί είναι αφενός αρκετά ευέλικτος, συγκλίνει γρήγορα και χαρακτηρίζεται από μικρή υπολογιστική πολυπλοκότητα, παράγοντες αρκετά σημαντικοί για αλληλεπιδραστικές εφαρμογές πραγματικού χρόνου όπως είναι η μετάδοση φωνής μέσω του διαδικτύου.



Εικόνα 3.6.2 Χρήση Equalizer για την ανακατασκευή

Ένας τυπικός αλγόριθμος LMS [18], προσπαθεί στηριζόμενος σε προηγούμενες τιμές προσπαθεί να προβλέψει την παρούσα, και παράλληλα να ελαχιστοποιήσει το λάθος της πρόβλεψης που έκανε. Όμως στην προκειμένη περίπτωση, λόγω της διαφύλλωσης που έχει εφαρμοστεί ο παραλήπτης έχει στην διάθεση του όχι μόνο τα προηγούμενα αλλά και τα επόμενα δείγματα αυτού που θέλει να προσεγγίσει. Έτσι, ο τυπικός αλγόριθμος LMS τροποποιήθηκε έτσι ώστε να λαμβάνει υπόψιν του προηγούμενα αλλά και τα επόμενα δείγματα του ανακατασκευασμένου, που τοποθετείται στην

μέση του φίλτρου. Την τεχνική αυτή, την ονομάσαμε LMS equalizer (βλ εικόνα 3.6.3).



Εικόνα 3.6.3 LMS Equalizer

Όπως παρατηρούμε και στο σχήμα τα C_1 έως C_n αποτελούν τους συντελεστές βάση των οποίων γίνεται η πρόβλεψη αλλά και το equalizing των ανακατασκευασμένων δειγμάτων. Επίσης το πλήθος n των συντελεστών C_j , ορίζει το μήκος του φίλτρου, το οποίο αποτελεί ένα σημαντικό κριτήριο απόδοσης. Είναι προφανές ότι δεν θα πρέπει να είναι πολύ μεγάλο γιατί, αφενός υφίστανται μεγάλες καθυστερήσεις και αφετέρου η συσχέτιση των δειγμάτων φωνής που απέχουν αρκετά είναι συνήθως μικρή. Από την άλλη βέβαια δεν πρέπει να είναι και πολύ μικρό, γιατί περιορίζουμε την ακρίβεια πρόβλεψης του φίλτρου.

Βασίζόμενοι λοιπόν, στα αποτελέσματα από την τεχνική ανακατασκευής Gauss, ορίσαμε το μήκος του φίλτρου ίσο με 5. Δηλαδή, στο equalizing θα συμμετέχουν εκτός των δυο γειτονικών δειγμάτων x_{i+1} , x_{i-1} και τα αμέσως επόμενα γειτονικά δείγματα x_{i+2} και x_{i-2} . Ακόμη, σημαντικό στοιχείο για την ταχύτητα σύγκλισης του φίλτρου παίζουν και οι αρχικές τιμές των συντελεστών C_i , που στην προκειμένη περίπτωση αποδείχθηκε ότι ένα καλό σημείο εκκίνησης είναι το $C_{\frac{(n+1)}{2}-1} = 0.5$, $C_{\frac{n+1}{2}} = 0.01$, $C_{\frac{(n+1)}{2}+1} = 0.5$, και με όλους τους υπόλοιπους συντελεστές $C_j = 0$. Αναλυτικότερα η λειτουργία του equalizer χωρίζεται σε δύο καταστάσεις, στην κατάσταση εκπαίδευσης και στην κατάσταση μη-εκπαίδευσης. Στην πρώτη κατάσταση εισέρχεται, όταν όλα τα δεδομένα στον equalizer, έχουν ληφθεί σωστά χωρίς να σημειωθεί κάποιο λάθος στην μετάδοση. Κατά την διάρκεια αυτής της περιόδου, ο παραλήπτης υπολογίζει τις εκτιμήσεις για τα πραγματικά δεδομένα Στην

συνέχεια βάση της διαφοράς της εκτίμησης και της πραγματικής τιμής του ληφθέντος δείγματος υπολογίζει το λάθος της εκτίμησης (e). Το λάθος αυτό στην συνέχεια χρησιμοποιείται για να ρυθμιστούν οι συντελεστές C_j ως εξής:

$$C(k+1) = C(k) - \frac{\beta}{2} \nabla_c [e_c^2(k)] = C(k) - \beta e_c(k)x(k) \quad (3.19)$$

ή για κάθε συντελεστή ξεχωριστά:

$$C_j(k+1) = C_j(k) + \beta e(k)x\left(k - \frac{n-1}{2} - (j-1)\right) \quad (3.20)$$

όπου $C_j(k+1)$ είναι ο j -οστός συντελεστής ο οποίος εφαρμόζεται στο δείγμα $x(k+1)$, β είναι ο βαθμωτός συντελεστής του οποίου την τιμή θέσαμε ίση με 10^{-10} (βάση βιβλιογραφίας) και τέλος το $e(k)$ αποτελεί το λάθος που υπολογίζεται ως την διαφορά της εκτίμησης και της αντίστοιχης εισόδου του equalizer:

$$e(k) = \hat{x}(k) - x\left(k - \frac{n-1}{2} - (j-1)\right) \quad (3.21)$$

Όταν εισέλθουν στο equalizer, δεδομένα τα οποία έχουν ανακατασκευαστεί ο equalizer υπεισέρχεται σε κατάσταση μη-εκπαίδευσης. Κατά την διάρκεια της, οι συντελεστές C_j δεν ανανεώνονται και το αποτέλεσμα του equalizer χρησιμοποιείται ως η καινούργια τιμή του ανακατασκευασμένου δείγματος.

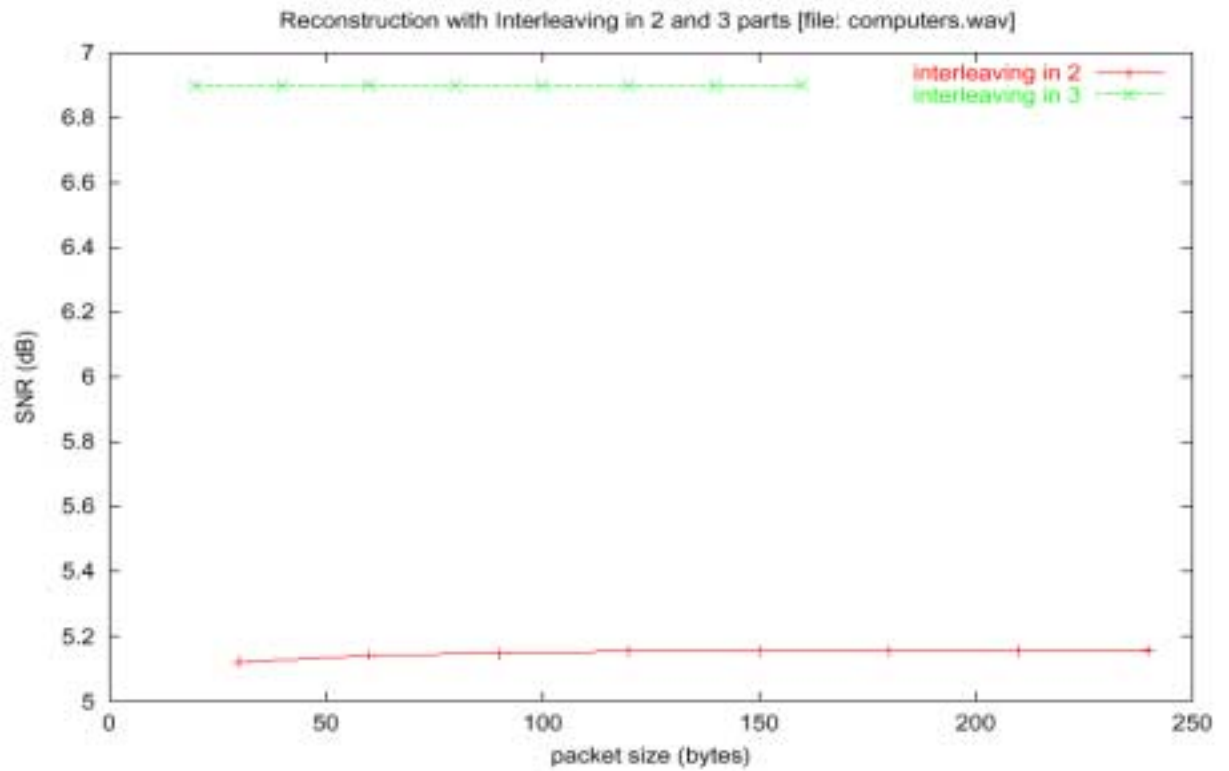
3.6.3 Πειραματικά αποτελέσματα

Σε αυτή την παράγραφο μελετήθηκε σε επίπεδο προσομοίωσης η απόδοση των προαναφερθέντων τεχνικών και συγκρίθηκαν με τις αντίστοιχες μεθόδους ανακατασκευής μέσου όρου με την χρήση ή μη μετασχηματισμού (βλ. παράγραφο §3.4.1). Ο προσομοιωτής που είχε υλοποιηθεί, επεκτάθηκε έτσι ώστε να υποστηρίζει και τις δυο τελευταίες μεθόδους ανακατασκευής. Τα πειράματα πραγματοποιήθηκαν στην ίδια ομάδα αρχείων ήχου τα οποία περιγράφονται στον πίνακα 3.4.1 και ως κριτήριο ποιότητας ανακατασκευής θεωρείται ξανά ο σηματοθορυβικός λόγος SNR (βλ. σχέση 3.1). Τέλος, όπως και προηγουμένως θεωρούμε ότι ένα πακέτο από κάθε τριάδα διαφύλλωσης χάνεται. Πιο αναλυτικά πραγματοποιήθηκαν τρία σύνολα μετρήσεων:

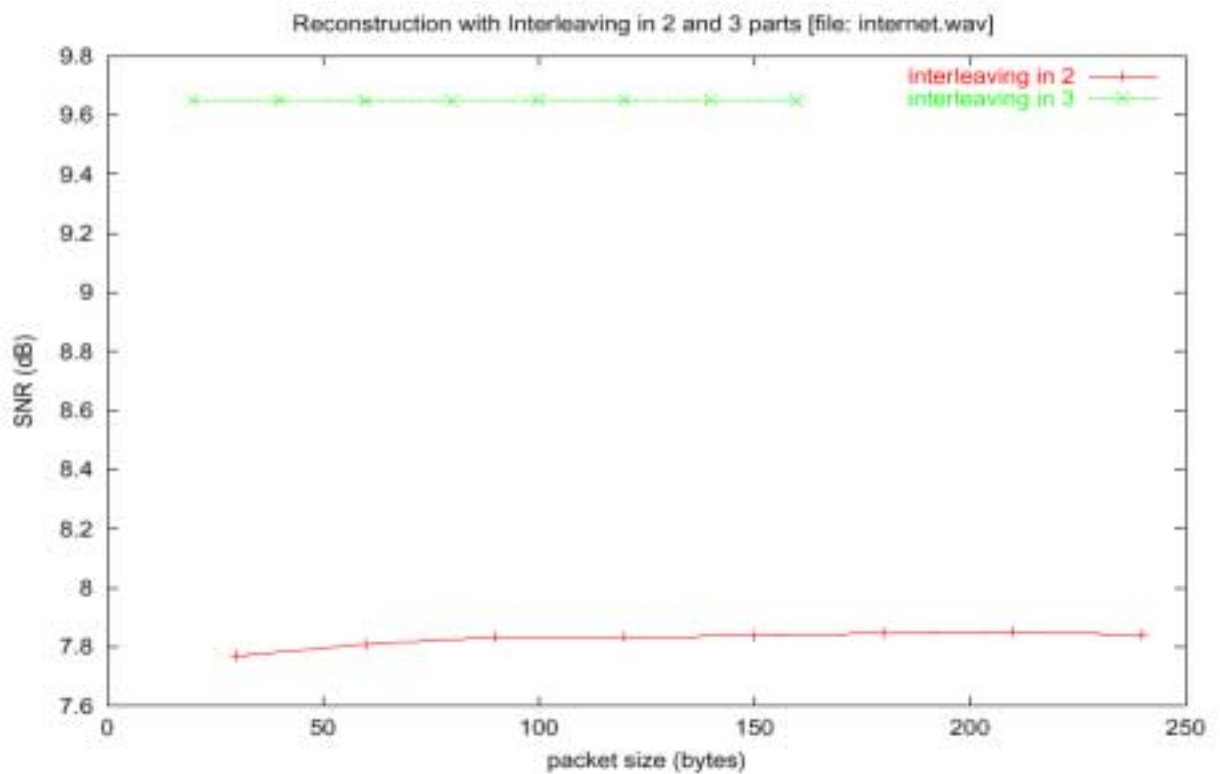
Το πρώτο σύνολο μετρήσεων, αφορά την σύγκριση της τεχνικής ανακατασκευής μέσου όρου με χρήση διαφύλλωσης σε δυο μέρη χωρίς την εφαρμογή μετασχηματισμού, με την αντίστοιχη μέθοδο ανακατασκευής μέσου όρου με χρήση διαφύλλωσης σε τρία μέρη. Όπως είναι λογικό η δεύτερη υπερτερεί αφού οι απώλειες είναι της τάξης του 30%, ενώ στην πρώτη είναι 50%. Αυτό γίνεται εμφανές στις εικόνες 3.6.4 έως και 3.6.7.

Αυτό που παρουσιάζει ιδιαίτερο ενδιαφέρον είναι το δεύτερο σύνολο μετρήσεων που αναφέρεται στην σύγκριση μεταξύ των τεχνικών ανακατασκευής μέσου όρου με εφαρμογή μετασχηματισμού και χρήση διαφύλλωσης σε δυο μέρη, με την αντίστοιχη μέθοδο ανακατασκευής μέσου όρου με χρήση διαφύλλωσης σε τρία μέρη. Παρατηρώντας τις εικόνες 3.6.8 έως 3.6.11, καταλήγουμε στο συμπέρασμα ότι η χρήση διαφύλλωσης σε τρία μέρη, υπερτερεί κατά τουλάχιστο 0.5 dB, της τεχνικής ανακατασκευής μέσου όρου με εφαρμογή μετασχηματισμού και χρήση διαφύλλωσης σε δυο μέρη. Αυτό αποτελεί ένα πολύ σημαντικό συμπέρασμα, γιατί η υπολογιστική ισχύ και πολυπλοκότητα που απαιτεί η τεχνική διαφύλλωσης σε τρία μέρη είναι κατά πολύ μικρότερη αυτής που χρησιμοποιεί μετασχηματισμό. Αυτό συνεπάγεται μικρότερες καθυστερήσεις στην επεξεργασία των δειγμάτων φωνής, παράγοντας πολύ σημαντικός για αλληλεπιδραστικές εφαρμογές πραγματικού χρόνου, όπως στην περίπτωση μετάδοσης φωνής μέσω του διαδικτύου. Σε αυτό το σημείο κρίνεται απαραίτητο, να επισημάνουμε τον τρόπο με τον οποίο γίνεται η σύγκριση μεταξύ των παραπάνω τεχνικών. Επειδή, η μια εφαρμόζει διαφύλλωση σε δυο μέρη ενώ η άλλη σε τρία οι συγκρίσεις πρέπει να γίνουν σε ανάλογα μεγέθη πακέτων. Πιο συγκεκριμένα, έστω N το μέγεθος των δεδομένων που θέλει να μεταδώσει ο αποστολέας. Ουσιαστικά N θα είναι και το μέγεθος ενός ζεύγους διαφύλλωσης ή μιας τριάδας διαφύλλωσης. Έτσι με χρήση της πρώτης τεχνικής ο αποστολέας μεταδίδει 2 πακέτα μεγέθους $N/2$, ενώ με την άλλη τεχνική αντίστοιχα θα μεταδώσει 3 πακέτα μεγέθους $N/3$. Άρα, η σύγκριση των δυο μεθόδων γίνεται μετρώντας την απόδοση κάθε τεχνικής, αντιστοιχίζοντας πακέτα μεγέθους M της πρώτης τεχνικής με πακέτα μεγέθους $3M/2$ της δεύτερης.

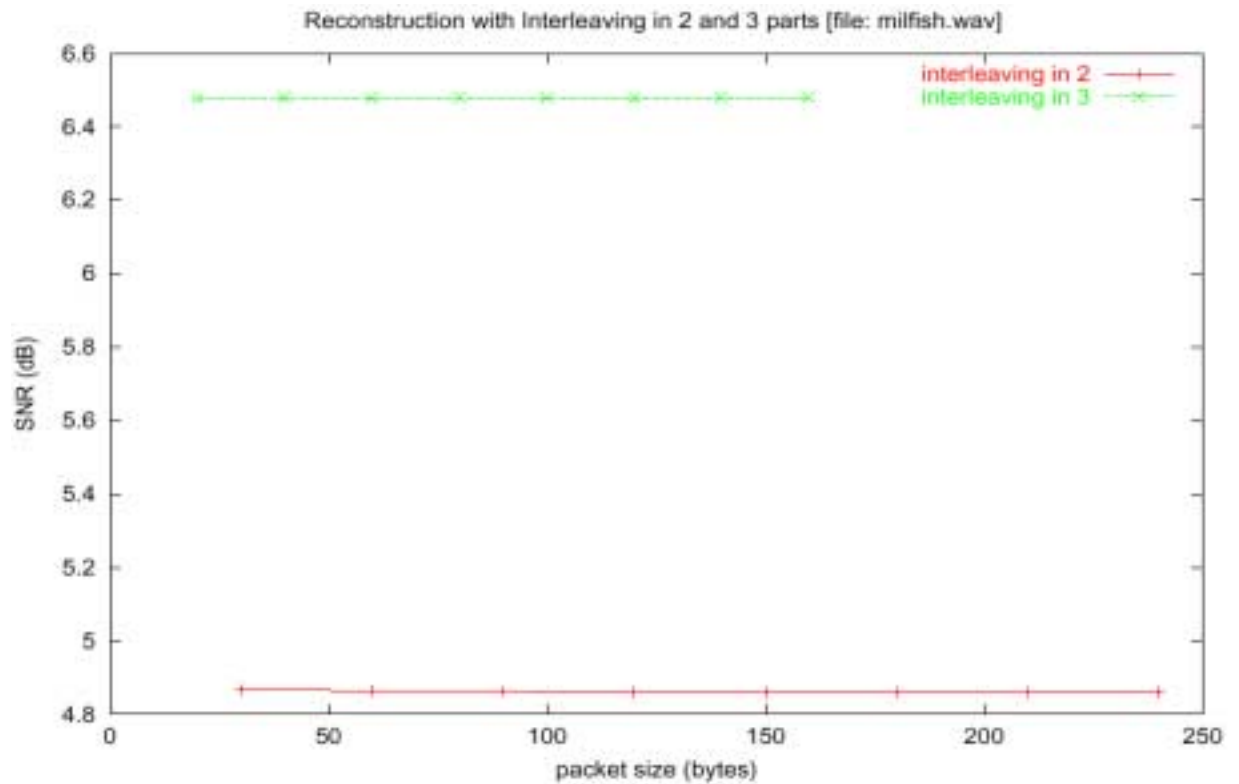
Τέλος το τρίτο μέρος των μετρήσεων, αναφέρεται στην βελτίωση που μπορούμε να πετύχουμε εφαρμόζοντας μετά την ανακατασκευή μέσου όρου το LMS equalizing. Τα αποτελέσματα των μετρήσεων παρουσιάζονται στις εικόνες 3.6.12 έως 3.6.15. Εύκολα μπορούμε να παρατηρήσουμε μια βελτίωση της ποιότητας ανακατασκευής της τάξης τουλάχιστο 1dB, η οποία είναι αρκετή για να εξισορροπήσει το κόστος σε καθυστέρηση και υπολογιστική ισχύ που εισάγεται από την χρήση του equalizer. Όπως είχαμε αναφέρει στην αντίστοιχη παράγραφο (βλ. παράγραφο §3.6.2) ο equalizer λειτουργεί σε δυο καταστάσεις: σε κατάσταση εκπαίδευσης και σε κατάσταση μη-εκπαίδευσης. Αν ακολουθούσαμε το προηγούμενο πρότυπο εμφάνισης σφαλμάτων (ένα σε κάθε τριάδα διαφύλλωσης) ο equalizer δεν θα εισερχόταν ποτέ σε κατάσταση εκπαίδευσης. Έτσι το πρότυπο εμφάνισης σφαλμάτων τροποποιήθηκε σε μια πιο ρεαλιστική μορφή, και πιο συγκεκριμένα έτσι ώστε να συμβαίνει μια απώλεια ενός πακέτου ανά δεύτερη τριάδα διαφύλλωσης.



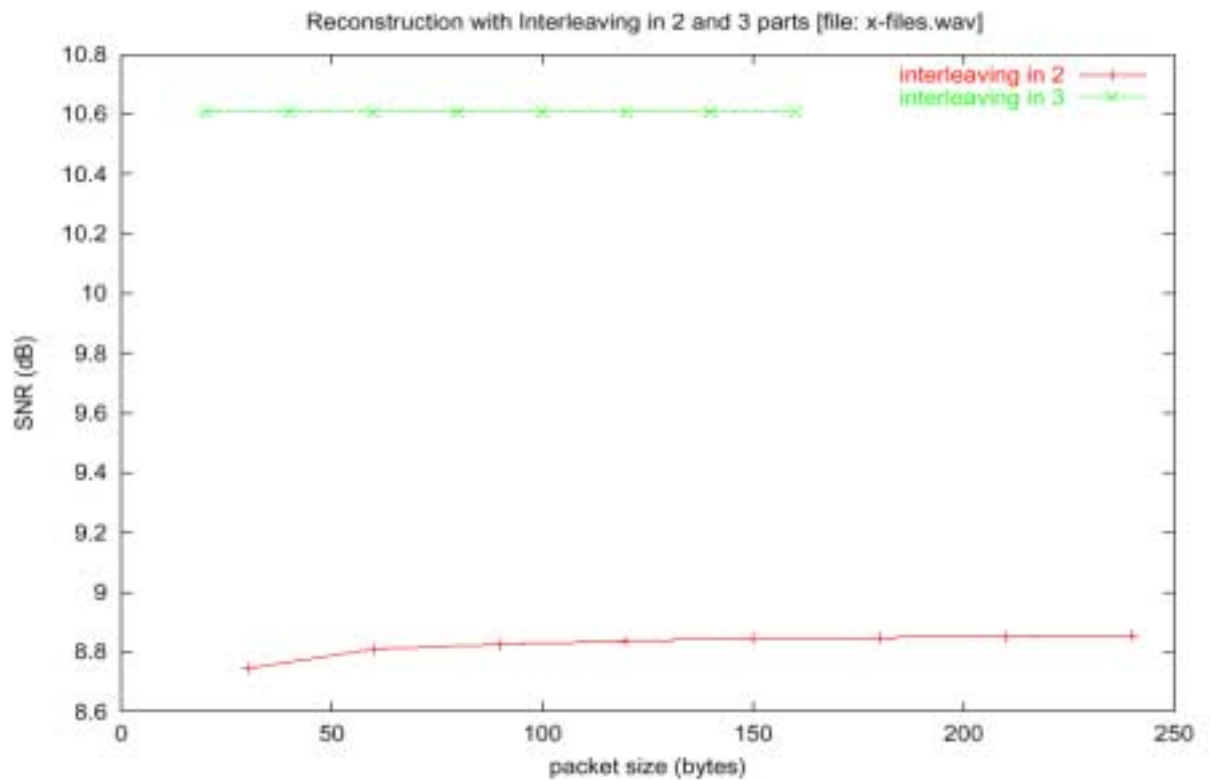
Εικόνα 3.6.4 Σύγκριση τεχνικών διαφυλλώσεων σε 2 ή 3 μέρη (computers.wav)



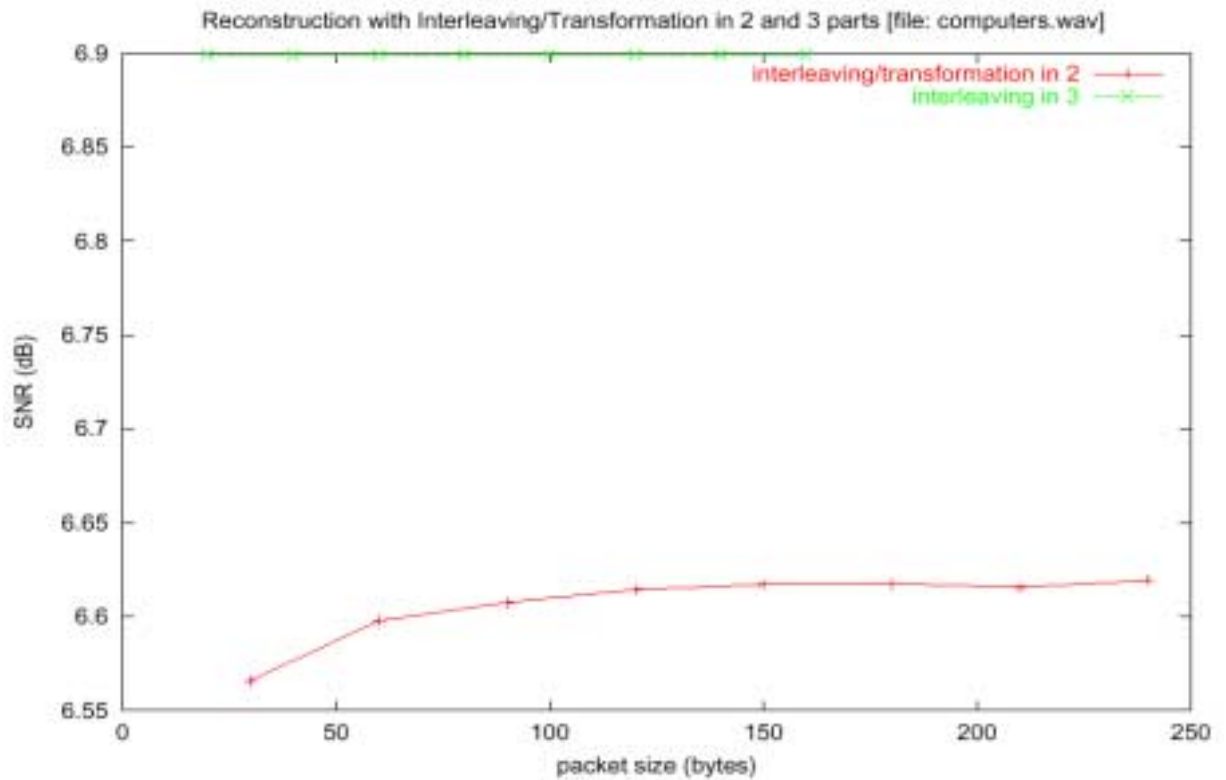
Εικόνα 3.6.5 Σύγκριση τεχνικών διαφυλλώσεων σε 2 ή 3 μέρη (internet.wav)



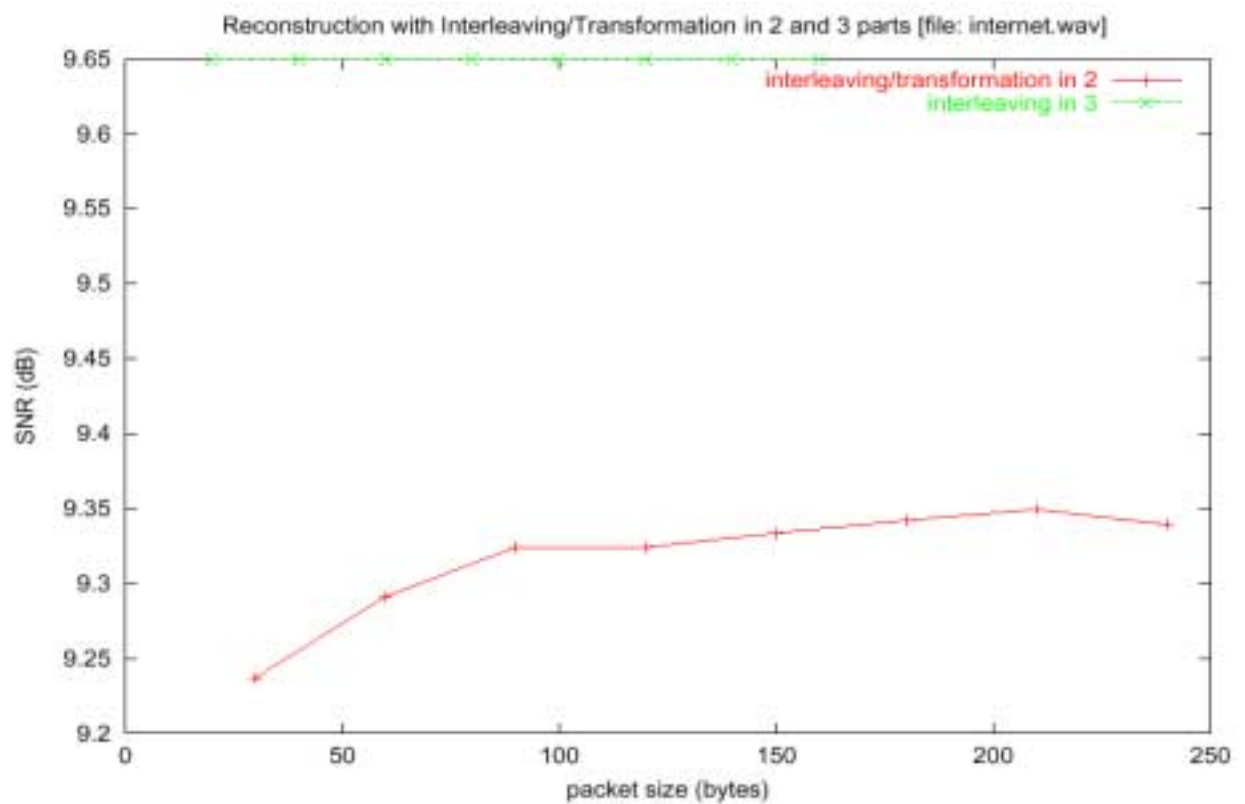
Εικόνα 3.6.6 Σύγκριση τεχνικών διαφυλλώσεων σε 2 ή 3 μέρη (milfish.wav)



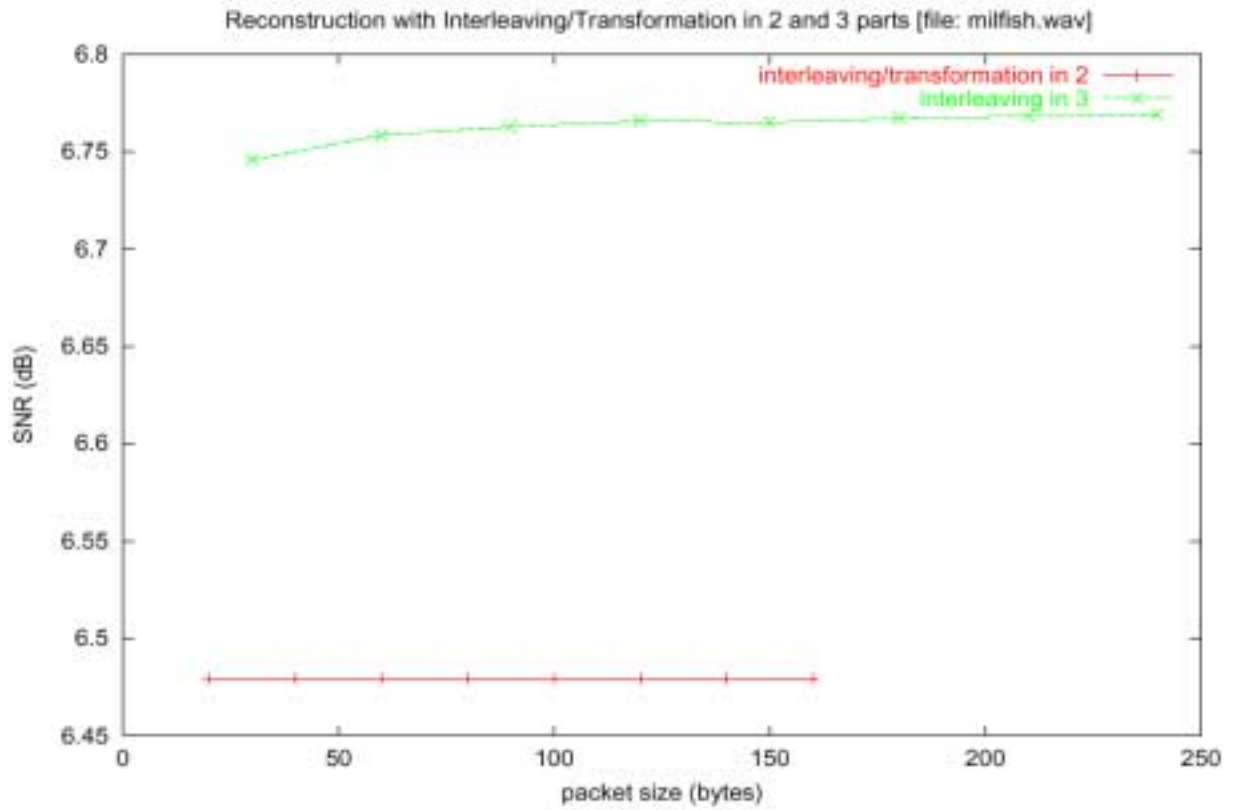
Εικόνα 3.6.7 Σύγκριση τεχνικών διαφυλλώσεων σε 2 ή 3 μέρη (x-files.wav)



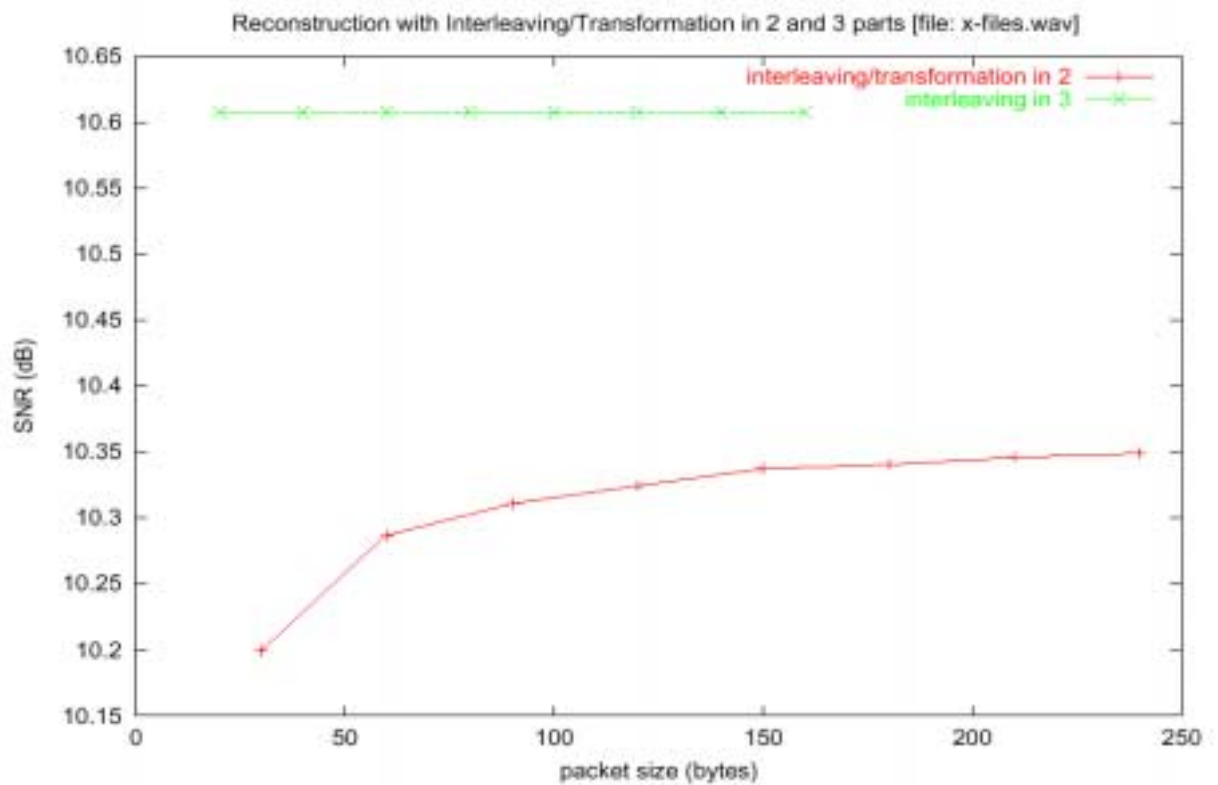
Εικόνα 3.6.8 Σύγκριση διαφύλλωσης στα 3 και μετασχηματισμού (*computers.wav*)



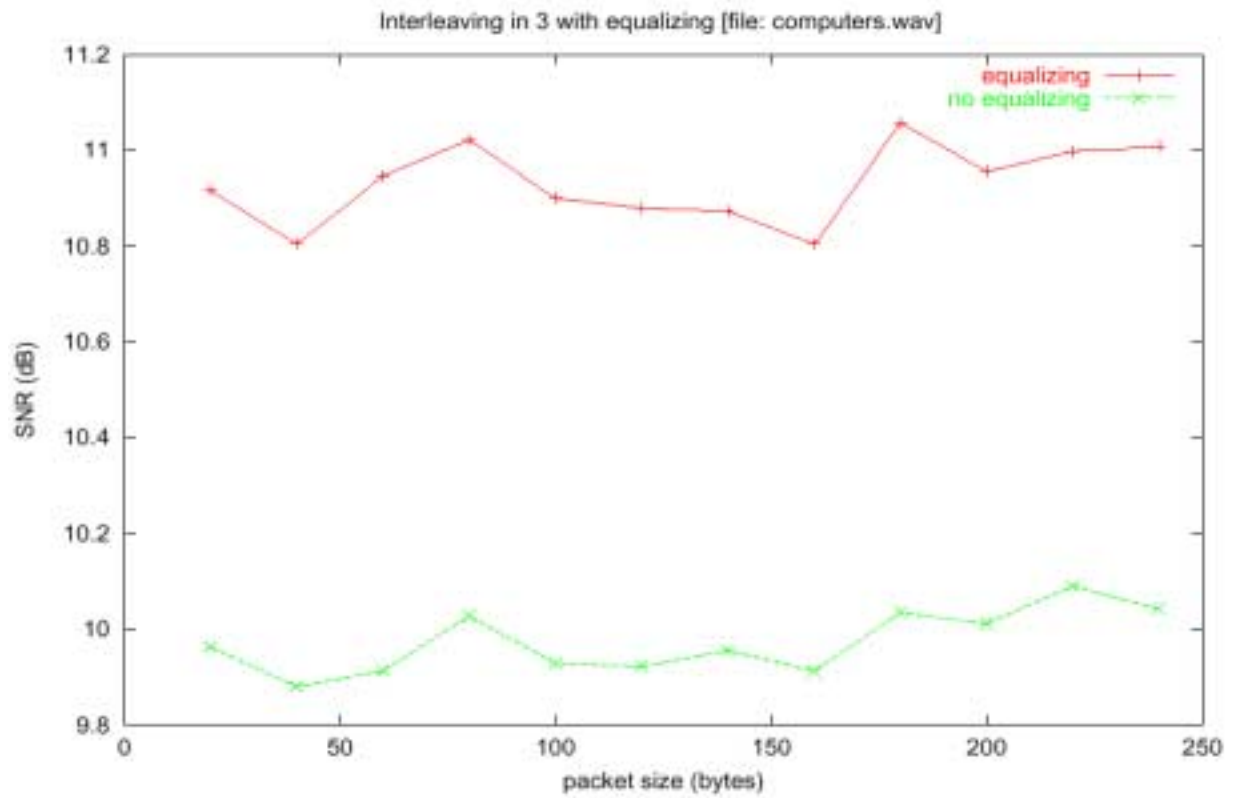
Εικόνα 3.6.9 Σύγκριση διαφύλλωσης στα 3 και μετασχηματισμού (*internet.wav*)



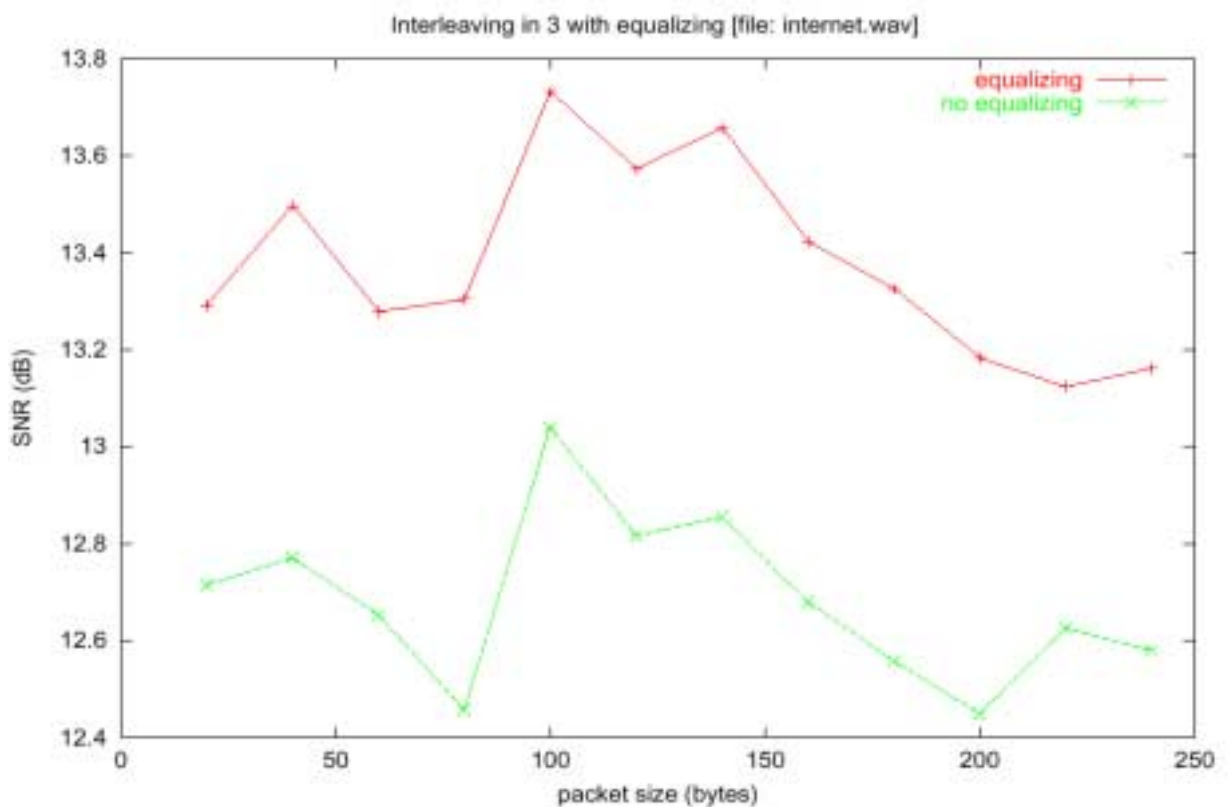
Εικόνα 3.6.10 Σύγκριση διαφύλλωσης στα 3 και μετασχηματισμού (milfish.wav)



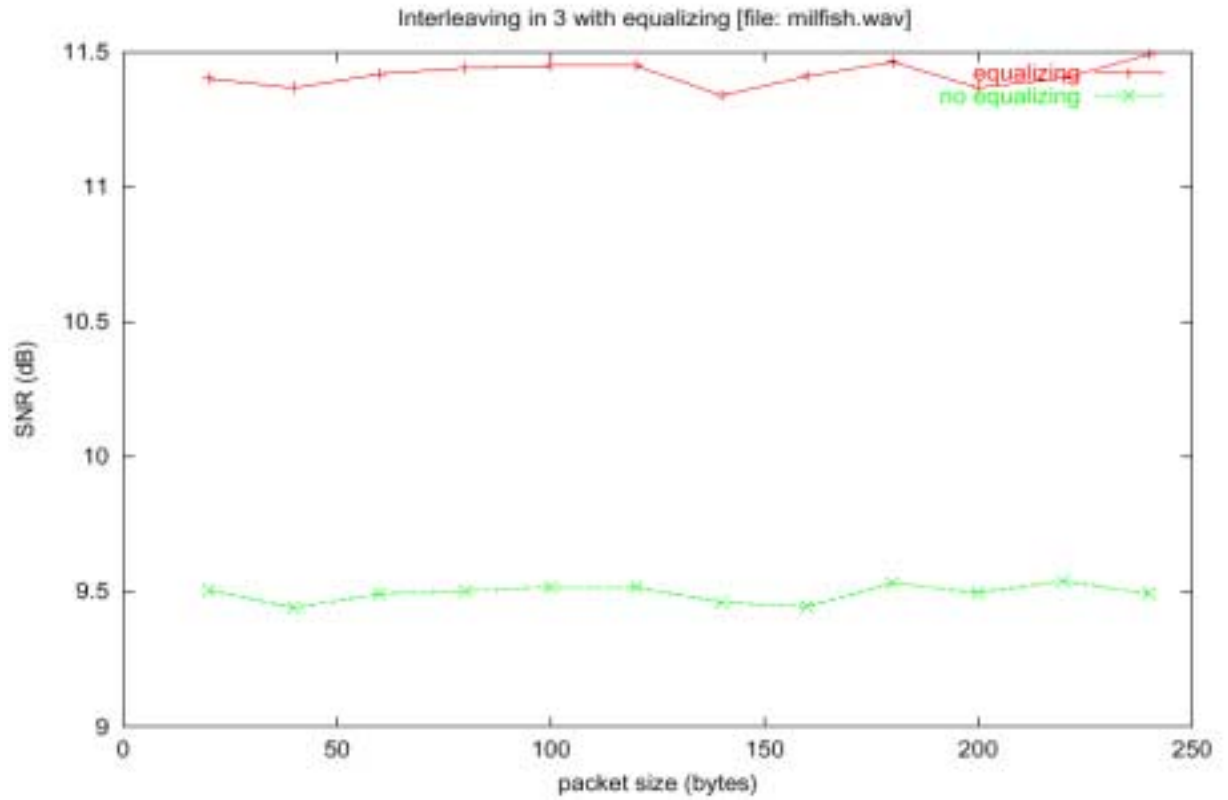
Εικόνα 3.6.11 Σύγκριση διαφύλλωσης στα 3 και μετασχηματισμού (milfish.wav)



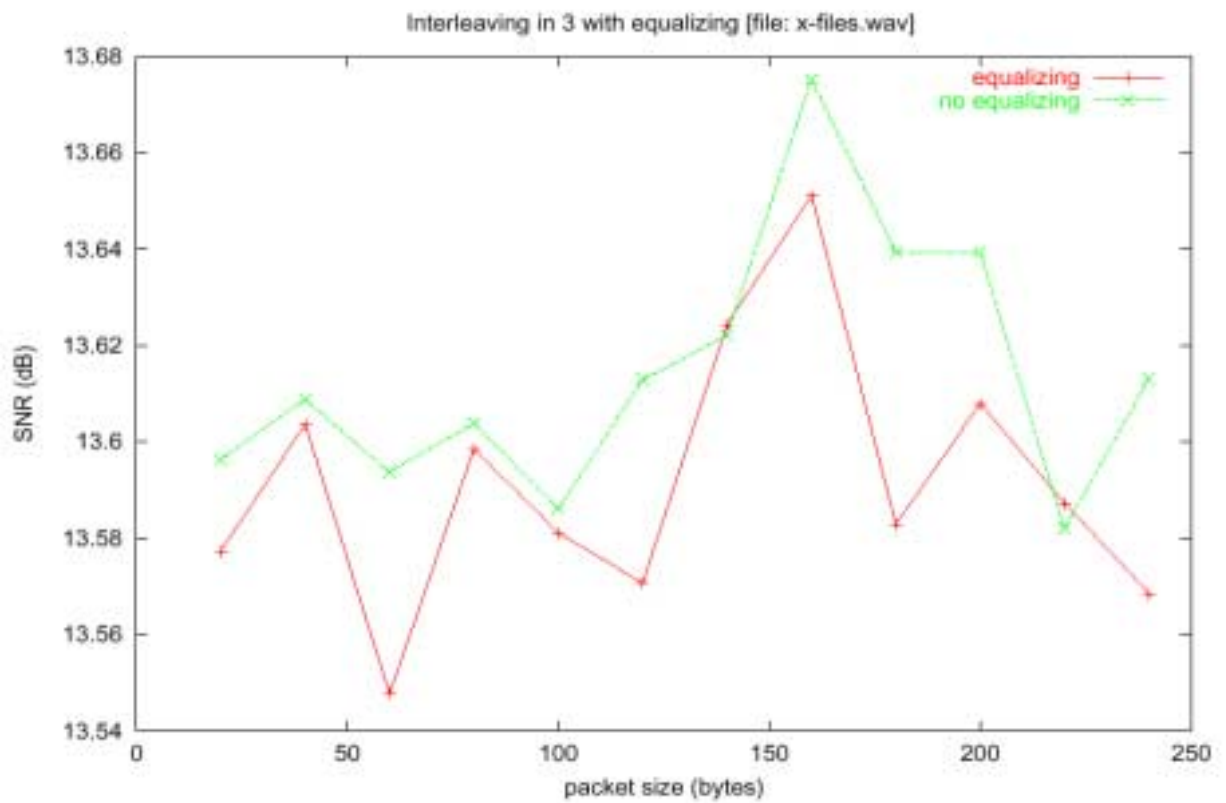
Εικόνα 3.6.12 Χρήση LMS equalizer (computers.wav)



Εικόνα 3.6.13 Χρήση LMS equalizer (internet.wav)



Εικόνα 3.6.14 Χρήση LMS equalizer (milfish.wav)



Εικόνα 3.6.15 Χρήση LMS equalizer (x-files.wav)

3.7 Συμπεράσματα-Προτάσεις

Στις προηγούμενες παραγράφους αναφέραμε αλλά και αναλύσαμε αρκετές μεθόδους ανακατασκευής. Αρχικά παρατηρήσαμε ότι στις περισσότερες περιπτώσεις η απόδοση μιας τεχνικής είναι αντιστρόφως ανάλογη της πολυπλοκότητας και της υπολογιστικής ισχύς που απαιτείται είτε από τον αποστολέα είτε από τον παραλήπτη. Στην συνέχεια επισημάνθηκε το γεγονός ότι στην προσπάθεια επανόρθωσης πρέπει να συμμετέχουν τόσο ο αποστολέας όσο και ο παραλήπτης ώστε να επιτύχουμε τα μεγαλύτερα δυνατά αποτελέσματα. Μια μορφή μιας τέτοιας συνεργασίας παρουσιάστηκε στις περιπτώσεις χρήσης μετασχηματισμών κατά την μετάδοση των δεδομένων. Η συνεργασία αυτή περιοριζόταν, σε μια «συμφωνία» μεταξύ των δυο σχετικά με τα δυνατά μεγέθη πακέτων που θα μεταδιδόταν μέσω του δικτύου. Αυτό είναι απαραίτητο γιατί το μέγεθος των πινάκων μετασχηματισμών είναι ανάλογο με το μέγεθος των πακέτων. Γίνεται εύκολα αντιληπτό, ότι για δυναμικά μεταβλητό μέγεθος πακέτων θα πρέπει ο αποστολέας αλλά και ο παραλήπτης να υπολογίζουν κάθε φορά τους αντίστοιχους πίνακες μετασχηματισμών και στην συνέχεια να τους εφαρμόζουν στα δεδομένα. Όπως καταλαβαίνουμε κάτι τέτοιο εκτός του ότι είναι υπολογιστικά απαιτητικό, εισάγει πολύ μεγάλες καθυστερήσεις ειδικά στην κατάσταση υπολογισμού των πινάκων, γεγονός που καθιστά απαγορευτική την χρήση του. Αρά η μόνη λύση ώστε να έχουν εφαρμογή οι τεχνικές των μετασχηματισμών είναι τόσο ο αποστολέας αλλά και ο παραλήπτης να έχουν συμφωνήσει σε ένα σταθερό εύρος δυνατών μεγεθών των πακέτων ώστε οι πίνακες μετασχηματισμών να έχουν υπολογιστεί off-line. Κατά αυτόν τον τρόπο η μόνη καθυστέρηση που προκύπτει είναι αυτή της εφαρμογής των μετασχηματισμών. Οι οποίοι με την σειρά τους προκαλούν βελτίωση της ποιότητας ανακατασκευής σε βαθμό που είναι αρκετός για να εξισορροπήσει το παραγόμενο κόστος σε καθυστέρηση και υπολογιστική ισχύ.

Στα πλαίσια αυτής της εργασίας προτείνεται ένα σχήμα ανακατασκευής αφενός βασισμένο στην δυναμική συνεργασία του αποστολέα-παραλήπτη και αφετέρου προσαρμόσιμο στις συνθήκες συμφόρησης του δικτύου.

Αρχικά, πρέπει να υπενθυμίσουμε ότι η μετάδοση των πολυμεσικών δεδομένων (φωνή ή/και video), γίνεται με την χρήση του πρωτοκόλλου IETF Real Time Protocol-RTP [47,48]. Βασικές λειτουργίες του πρωτοκόλλου αποτελούν η παροχή χρονικών σφραγίδων (timestamping), διαδοχική αρίθμηση (sequence numbering) και γενικά η παροχή μηχανισμών χρονισμού των πακέτων. Μέσα από αυτούς τους μηχανισμούς το RTP παρέχει μια υπηρεσία μεταφοράς, από άκρη σε άκρη (end-to-end) των πολυμεσικών δεδομένων πραγματικού χρόνου.

Επίσης το RTP σχεδιάστηκε έτσι ώστε να συνυπάρχει με ένα συμπληρωματικό πρωτόκολλο ελέγχου το IETF Real Time Control Protocol-RTCP [47,48]. Κατά την

διάρκεια μιας RTP συνόδου (RTP session), οι συμμετέχοντες σε αυτό (αποστολείς και παραλήπτες), ανταλλάσσουν κατά τακτά χρονικά διαστήματα RTCP πακέτα μεταφέροντας αναδραστική πληροφορία για την ποιότητα παράδοσης των πολυμεσικών δεδομένων.

Αναλυτικότερα, έχουν οριστεί δυο συγκεκριμένα RTCP πακέτα ελέγχου για την μεταφορά αυτής της πληροφορίας:

- *Αναφορά Παραλήπτη (Receiver Report)*: Στέλνονται από τους παραλήπτες και περιέχουν πληροφορία σχετικά με την ποιότητας λήψης των πολυμεσικών δεδομένων όπως το μέγιστο αριθμό πακέτων που έχουν λάβει, τον αριθμό πακέτων που χάθηκαν, την διακύμανση της καθυστέρησης (jitter) καθώς και χρονικές σφραγίδες (timestamps) για τον υπολογισμό της συνολικής καθυστέρησης (round-trip delay).
- *Αναφορά Αποστολέα (Sender Report)*: Στέλνονται από τους αποστολείς και εκτός των δεδομένων που περιέχονται και στην αναφορά του παραλήπτη, περιλαμβάνει πληροφορία εσωτερικού συγχρονισμού των πολυμεσικών δεδομένων. π.χ-lip synchronization σε περίπτωση τηλεσυνδιάσκεψης

Εκμεταλλευόμενοι τις δυνατότητες του πρωτοκόλλου RTP και στηριζόμενοι στα αποτελέσματα των μετρήσεων που έχουν διεξαχθεί, είμαστε σε θέση να προτείνουμε ένα δυναμικά προσαρμόσιμο σχήμα ανακατασκευής.

Πιο συγκεκριμένα: Αρχικά, εγκαθίσταται μια RTP σύνοδος (RTP session) μεταξύ των αποστολέων και παραληπτών. Κατά την διάρκεια μετάδοσης των πολυμεσικών δεδομένων, όλοι οι συμμετέχοντες της συνόδου έχουν γνώση της κατάστασης του δικτύου, μέσω των αντίστοιχων αναφορών. Εκμεταλλευόμενοι της γνώσης αυτής, μπορούν να αλλάζουν κάθε φορά τον τρόπο μετάδοσης και επεξεργασίας των δεδομένων έτσι ώστε σε περίπτωση εμφάνισης κάποιου σφάλματος, να έχουμε βέλτιστη ανακατασκευή στον παραλήπτη.

Κατά αυτόν τον τρόπο, σε κατάσταση χαμηλής συμφόρησης του δικτύου ο αποστολέας διαφυλλώνει τα δεδομένα σε τρία μέρη και τα αποστέλλει. Ο παραλήπτης λόγω του χαμηλού βαθμού συμφόρησης του δικτύου, έχει να αντιμετωπίσει μεμονωμένα λάθη το οποίο επιτυγχάνει κάνοντας χρήση της μεθόδου ανακατασκευής μέσου όρου και παράλληλης εφαρμογής του equalizing (βλ παράγραφο §3.6.2).

Αντίστοιχα σε καταστάσεις «εχθρικού καναλιού», όπου είναι πολύ πιθανό να συμβούν συσσωρευμένα λάθη, ο αποστολέας θα χρησιμοποιήσει μια διαφύλλωση δυο επιπέδων με παράλληλη χρήση μετασχηματισμού(βλ παράγραφο §3.4.2). Βέβαια όπως έχει επισημανθεί θα πρέπει να προκαθοριστεί το εύρος των μεγεθών των πακέτων ώστε να υπολογιστούν οι πίνακες των αντίστοιχων μετασχηματισμών off-line. Αυτό είναι εφικτό γιατί το μέγεθος των πακέτων που μεταφέρουν πολυμεσικά δεδομένα είναι συνήθως μικρό και δεν παρουσιάζει μεγάλες διακυμάνσεις [21,29].

Τέλος οι παράγοντες που θα χαρακτηρίζουν την κατάσταση του δικτύου είναι οι τιμή της συνολικής καθυστέρησης (round trip delay), η μεταβλητότητα της καθυστέρησης (jitter) και ο ρυθμός εμφάνισης λαθών στο κανάλι.

ΚΕΦΑΛΑΙΟ 4

Επισκόπηση εργασίας – Μελλοντικές επεκτάσεις

Στα πλαίσια αυτής της εργασίας, μελετήθηκαν αλλά και προτάθηκαν νέες τεχνικές βελτίωσης της παρεχόμενης ποιότητας υπηρεσίας των εφαρμογών μετάδοσης φωνής μέσω του διαδικτύου. Ουσιαστικά η επιδιωκόμενη βελτίωση επιτυγχάνεται αντιμετωπίζοντας δυο βασικά χαρακτηριστικά του διαδικτύου: την διακύμανση της καθυστέρησης (jitter) και τις απώλειες πακέτων.

Στο πρώτο μέρος της εργασίας παρουσιάστηκαν αλλά και υλοποιήθηκαν μια σειρά από τους πιο γνωστούς αλγορίθμους εξάλειψης της διακύμανσης της καθυστέρησης. Στην συνέχεια μέσα από μια σειρά πειραμάτων εντοπίστηκαν οι αδυναμίες τους και τέλος προτάθηκε ένας καινούργιος αλγόριθμος ο οποίος αποδίδει σαφώς καλύτερα.

Όσον αφορά το δεύτερο μέρος της εργασίας, ακολουθήθηκε παρόμοια ροή. Αρχικά παρουσιάστηκαν οι ευρέως χρησιμοποιούμενες μέθοδοι ανακατασκευής χαμένων πακέτων φωνής. Συνεχίζοντας, εκτελώντας μια σειρά πειραμάτων παρατέθηκαν τα πλεονεκτήματα αλλά και τα μειονεκτήματα αυτών, ώστε τελικά να καταλήξουμε στην πρόταση ενός καινούργιου προσαρμοστικού σχήματος ανακατασκευής.

Τα πειράματα και στα δυο μέρη της εργασίας έγιναν με την χρήση κατάλληλων προσομοιωτών οι οποίοι αναπτύχθηκαν στα πλαίσια αυτής της εργασίας. Ως φυσική εξέλιξη αυτής της εργασίας θα αποτελούσε η ενσωμάτωση αυτών των αλγορίθμων σε μια πραγματική εφαρμογή μετάδοσης φωνής μέσω του διαδικτύου. Πιο συγκεκριμένα χρησιμοποιώντας το Java Media Framework API (JMF), παρέχονται οι κατάλληλοι μηχανισμοί ώστε να υλοποιηθεί από την αρχή μια εφαρμογή μετάδοσης φωνής μέσω του διαδικτύου. Κατά αυτόν τον τρόπο γίνεται δυνατή η ενσωμάτωση των προαναφερθέντων αλγορίθμων ώστε να εξαχθούν μετρήσιμα συμπεράσματα για την ταυτόχρονη εφαρμογή τους.

Παράρτημα Α

Σε αυτό το κεφάλαιο παρατίθεται η απόδειξη ύπαρξης αντίστροφων για τους πίνακες μετασχηματισμών που συναντήθηκαν στα κεφάλαια §3.4 και §3.5. Πιο συγκεκριμένα θα δείξουμε την αντιστρεψιμότητα ενός μόνο πίνακα (βλ. Σχέση 3.7) και εκμεταλλευόμενοι το γεγονός ότι όλοι οι υπόλοιποι, είναι της ίδιας μορφής ακολουθώντας την ίδια διαδικασία μπορούμε εύκολα να το αποδείξουμε την ύπαρξη αντίστροφου και για αυτούς.

Υπενθυμίζουμε τον μετασχηματισμό που ορίζεται από την σχέση 3.7 :

$$\mathbf{T} = \mathbf{A}^{-1} \times \mathbf{B} = \begin{pmatrix} 1 & 1/5 & 0 & \dots & 0 & 0 & 0 \\ 1/6 & 1 & 1/6 & \dots & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & \dots & 1/6 & 1 & 1/6 \\ 0 & 0 & 0 & \dots & 0 & 1/6 & 1 \end{pmatrix}^{-1} \begin{pmatrix} 4/5 & 2/5 & 0 & 0 & \dots & 0 & 0 & 0 & 0 & 0 \\ 0 & 1/3 & 2/3 & 1/3 & \dots & 0 & 0 & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & \dots & 1/3 & 2/3 & 1/3 & 0 & 0 \\ 0 & 0 & 0 & 0 & \dots & 0 & 0 & 1/3 & 2/3 & 1/3 \end{pmatrix}$$

Η παραπάνω εξίσωση έχει μοναδική λύση μόνο όταν ο πίνακας \mathbf{A} αντιστρέφεται. Ικανή και αναγκαία συνθήκη η οποία εξασφαλίζει την ύπαρξη αντιστρόφου για ένα πίνακα, αποτελεί η μη-μηδενική τιμή της ορίζουσας του.

Πιο συγκεκριμένα:

$$\det(\mathbf{A}) = \begin{vmatrix} 1 & 1/5 & 0 & \dots & 0 & 0 & 0 \\ 1/6 & 1 & 1/6 & \dots & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & \dots & 1/6 & 1 & 1/6 \\ 0 & 0 & 0 & \dots & 0 & 1/6 & 1 \end{vmatrix}_N \quad (1)$$

Αρχικά εφαρμόζουμε στην πρώτη γραμμή του \mathbf{A} την ιδιότητα *Laplace expansion*:

$$\det(\mathbf{A}) = \begin{vmatrix} 1 & 1/6 & 0 & \dots & 0 & 0 & 0 \\ 1/6 & 1 & 1/6 & \dots & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & \dots & 1/6 & 1 & 1/6 \\ 0 & 0 & 0 & \dots & 0 & 1/6 & 1 \end{vmatrix}_{N-1} - \frac{1}{5} \begin{vmatrix} 1/6 & 1/6 & 0 & \dots & 0 & 0 & 0 \\ 0 & 1 & 1/6 & \dots & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & \dots & 1/6 & 1 & 1/6 \\ 0 & 0 & 0 & \dots & 0 & 1/6 & 1 \end{vmatrix}_{N-1} \quad (2)$$

Στην συνέχεια εφαρμόζουμε την ίδια ιδιότητα και στους δυο πίνακες στο δεξί μέρος της σχέσης (2):

$$\det(\mathbf{A}) = \begin{vmatrix} 1 & 1/6 & 0 & \dots & 0 & 0 & 0 \\ 1/6 & 1 & 1/6 & \dots & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & \dots & 1/6 & 1 & 1/6 \\ 0 & 0 & 0 & \dots & 0 & 1/6 & 1 \end{vmatrix}_{N-1} - \frac{1}{30} \begin{vmatrix} 1 & 1/6 & 0 & \dots & 0 & 0 & 0 \\ 1/6 & 1 & 1/6 & \dots & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & \dots & 1/6 & 1 & 1/6 \\ 0 & 0 & 0 & \dots & 0 & 1/6 & 1 \end{vmatrix}_{N-2} \quad (3)$$

Όλες οι ορίζουσες που προκύπτουν είναι για τον τριδιαγώνιο πίνακα \mathbf{T} και μπορούν εύκολα να υπολογιστούν χρησιμοποιώντας την εξίσωση:

$$\det(\mathbf{T}_N) = \frac{\beta^{N+1} - \alpha^{N+1}}{\beta - \alpha} \quad (4)$$

Όπου τα α και β αποτελούν τις ρίζες της εξίσωσης $x^2 - x - \left(\frac{1}{6}\right)^2 = 0$

Συνδυάζοντας τις (3) και (4), είναι εύκολο να αποδείξουμε ότι η $\det(\mathbf{A})$ είναι πάντα μη-μηδενική, γεγονός που εξασφαλίζει την ύπαρξη του \mathbf{A}^{-1} .

Παράρτημα Β

Σε αυτή την παράγραφο θα δείξουμε πως μπορούμε προσαρμόζοντας την τεχνική δημιουργίας πυρήνων Gauss, να υπολογίσουμε τους συντελεστές a , b που χρησιμοποιήθηκαν στην παράγραφο §3.5 για την ανακατασκευή των χαμένων δειγμάτων. Ουσιαστικά αυτό που θέλουμε είναι οι τιμές των a , b να ακολουθούν την κατανομή Gauss.

Η σχέση που μας δίνει την κατανομή Gauss, θεωρώντας την μέση τιμή της ίση με μηδέν είναι:

$$f(x) = \frac{1}{\sqrt{2\pi}} e^{-\frac{x^2}{2\sigma^2}} \quad (1)$$

Για την δημιουργία ενός πυρήνα Gauss, πρέπει να εφαρμοστεί δειγματοληψία στην συνεχή κατανομή (βλ. Σχέση 1), αφού πρώτα αποφασιστεί το πλάτος του πυρήνα που θέλουμε να παραχθεί.

Αυτό αντιστοιχεί με το να επιλέξουμε μια τιμή για το σ . Την σχέση μεταξύ του σ και του πλάτους του πυρήνα (το οποίο θα το ονομάζουμε w), μπορούμε να την δούμε σαν το ποσοστό κάλυψης της επιφάνειας κάτω από την καμπύλη Gauss.

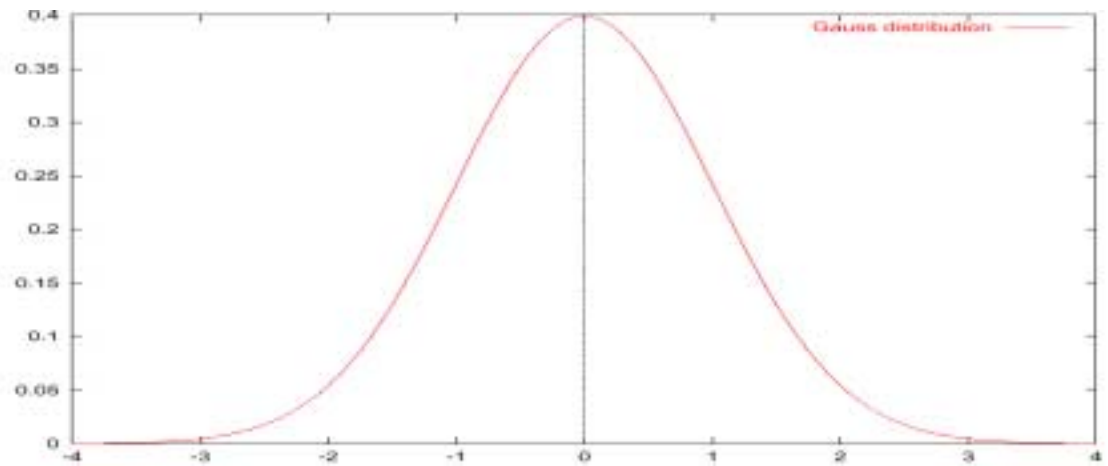
Μια επαρκής επιλογή είναι $w = 5\sigma$, η οποία αντιστοιχεί σε κάλυψη του 98,76% της επιφάνειας Gauss[♦].

Έτσι έχουμε $\sigma = \frac{w}{5}$ (2) και αν επιλέξουμε πυρήνα πλάτους $w = 5$, τότε η (2) γίνεται

$$\sigma = 1. \text{ Άρα η (1) γίνεται } f(x) = \frac{1}{\sqrt{2\pi}} e^{-\frac{x^2}{2}} \quad (3)$$

Η γραφική αναπαράσταση της (3) φαίνεται στο παρακάτω σχήμα:

[♦] Αν ακολουθήσουμε ακριβώς την ίδια διαδικασία, αλλά θέλαμε έναν πυρήνα πλάτους 3, θα χρησιμοποιούσαμε την σχέση $w = 3\sigma$ και θα καταλήγαμε σε $a = 0.5$. Δηλαδή ουσιαστικά η τεχνική ανακατασκευής μέσου όρου, είναι μια ειδική μορφή ανακατασκευής Gauss.



και αναλυτικότερα έχουμε:

$$f(1) = f(-1) = \frac{1}{\sqrt{2\pi}} e^{-\frac{1}{2}} \quad \text{και} \quad f(2) = f(-2) = \frac{1}{\sqrt{2\pi}} e^{-2}$$

Όμως θέλουμε τα a, b κανονικοποιημένα ώστε το άθροισμα τους να δίνει 1, άρα

$$a = \frac{f(1)}{f(1) + f(-1) + f(2) + f(-2)} \quad \text{και} \quad b = \frac{f(2)}{f(1) + f(-1) + f(2) + f(-2)}$$

Τέλος εκτελώντας τις πράξεις έχουμε $a = 0.4088$, $b = 0.0912$

Αναφορές

- [1] J-C. Bolot, "End-to-end packet delay and loss behaviour in the Internet", *Proc. ACM Sigcomm '93, SanFrancisco, CA*, pp. 189-199, Aug. 1993.
- [2] A. Mukherjee, "On the dynamics and significance of low frequency components of Internet load", *Journal of Internetworking: Research and Experience*, vol. 5, no. 4, pp. 163-205, Dec.1994.
- [3] Weibin Zhao, David Olshefski and Henning Schulzrinne, "Internet Quality of Service: an Overview", Columbia University, July 99.
- [4] Telecommunication Standardization Sector Of ITU. ITU-T Recommendation G.114. *Technical report, International Telecommunication Union*, March 1993.
- [5] Jayant, N., "Effects of packet loss on waveform coded speech", *Fifth Int. Conference on Computer Communications*, pp.275-280, Atlanta, GA, October 1980.
- [6] Montgomery, W., "Techniques for packet voice synchronization", *IEEE Journal on Selected Areas In Communications*, 6(1), pp.1022-1028, December 1983.
- [7] Alvarez-Cuevas, F., Bertran, M., Oller, F., Selga, J., "Voice synchronization in packet switching networks", *IEEE Networks Magazine*, 7(5), pp.20-25, September 1993.
- [8] Ramjee, R., Kurose, J., Towsley, D., Schulzrinne, H., "Adaptive playout mechanisms for packetized audio applications in wide area networks", *Proc. IEEE Infocom '94*, Montreal, Canada, April 1994.
- [9] Moon, S., Kurose, J., Towsley, D., "Packet Audio Playout Delay Adjustment: Performance Bounds and Algorithms", Technical Paper, Dept. of Computer Science, Univ. of Massachusetts at Amherst, 1995.
- [10] <http://citeseer.nj.nec.com/schulzrinne98guide.html>
- [11] <http://java.sun.com>
- [12] H. Schulzrinne, "Audio and Video over Packet Networks - Issues, Architecture and Protocols, Overheads presented at INTEROP, Paris, October 1994. URL: <http://www.fokus.gmd.de/minos/employees/hgs/rtp/interop94.tex> and <http://www-net.cs.umass.edu/~sbmoon/traces.html>.
- [13] Jon Postel, editor, "Transmission Control Protocol specification," *ARPANET Working Group Request for Comment*, (September 1981), RFC793.
- [14] V. Jacobson, "Congestion avoidance and control, " *Proc. 1988 ACM SIGCOMM Conf.*, (Aug. 1988, Stanford), pp. 314-329.

- [15] Jesus Pinto, Kenneth J. Christensen, “An Algorithm for Playout of Packet Voice based on Adaptive Adjustment for Talkspurt Silence Periods”, *Proceedings of the 24th Conference on Local Computer Networks 1998*, Department of Computer Science and Engineering, University of South Florida.
- [16] Y. J. Cho, C.K Un, “ Performance Analysis of Reconstruction Algorithms for Packet Voice Communications,” *Computer Networks and ISDN Systems*, Vol. 26, No. 11, pp. 1385 - 1408, 1994.
- [17] P. DeLeon, C.J. Sreenan, “Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)”, Vol. 6, pp. 3097-3100, March 1999
- [18] S. Haykin, *Adaptive Filter Theory*. Englewood-Cliffs, N-J.: Prentice-Hall, 3d., 1998
- [19] Colin Perkins, Orion Hodson, Vicky Hardman, “A Survey of Packet-Loss Recovery Techniques for Streaming Audio”, Department of Computer Science, University College London, August 1998
- [20] J. Rosenberg and H. Schulzrinne. “An RTP payload format for generic forward error correction”. *IETF Audio/Video Transport working group*, December 1999. RFC 2733
- [21] J. Rosenberg . “Reliability enhancements to NeVoT”, December 1996.
- [22] H.F. Mattson and G. Solomon. “A new treatment of bose-chaudhuri codes.” *Journal of the Society of Industrial and Applied Mathematics (SIAM)*, 9(4):654-669, December 1961
- [23] I.S Reed and G. Solomon. “Polynomial codes over certain finite fields.” *Journal of the Society of Industrial and Applied Mathematics (SIAM)*, 8(2):300-304. June 1960
- [24] E.R Berlekamp. *Algebraic Coding Theory*. McGraw-Hill, 1968
- [25] J.L Massey. “Shift-register synthesis and BCH decoding”. *IEEE Transactions on Information Theory*, IT-15:122-127, 1969
- [26] V. Hardman, M. A. Sasse, M. Handley, and Watson. “Reliable audio for use over the Internet.” *In Proceedings of INET’ 95*, 1995
- [27] J.-C. Bolot and A. Vega-Garcia. “The case for FEC based error control for packet audio in the Internet”. To appear in *ACM Multimedia Systems*
- [28] <http://www-sop.inria.fr/rodeo/fphone/index.html>
- [29] <http://www-mice.cs.ucl.ac.uk/multimedia/software/rat/>

- [30] C. S. Perkins, I. Kouvelas, O. Hodson, V. Hardman, M. Handley, J.-C Bolot, A. Vega-Garcia, and S. Fosse-Parisis. "RTP payload format for redundant audio data". *IETF Audio/Video Transport working group*, 1997. RFC 2198
- [31] J. L. Ramsey. "Realization of optimum interleavers". *IEEE Transactions on Information Theory*, IT-16:338-345, May 1970
- [32] G. A. Miller and J. C. R. Licklider. "The intelligibility of interrupted speech". *Journal of the Acoustical Society of America*, 22(2):167-173, 1950
- [33] P. T. Brady. "Effects of transmission delay on conversational behavior on echo-free telephone circuits". *Bell System Technical Journal*, 50:115-134, January 1971
- [34] J. Nonnenmacher, E. Biersack, and D. Towsley. "Parity-based loss recovery for reliable multicast transmission". In *Proceedings ACM SIGCOMM'97*, Cannes, France, September 1997
- [35] J. G. Gruber and L. Strawczynski. "Subjective effects of variable delay and clipping in dynamically managed voice systems". *IEEE Transactions on Communications*, COM-33(8):801-808, August 1985
- [36] N. S. Jayant and S. W. Christensen. "Effects of packet losses in waveform coded speech and improvements due to an odd-even sample-interpolation procedure." *IEEE Transactions on Communications*, COM-29(2):101-109, February 1981
- [37] R. M. Warren. *Auditory Perception*. Pergamon Press, Inc., 1982
- [38] ETSI. Recommendation GSM 6.11 - Substitution and muting of lost frames for full rate speech channels, 1992
- [39] David. J. Goodman, Gordon B. Lockhart, Ondria. J. Wasem, Wai-Choong Wong, "Waveform Substitution Techniques for Recovering Missing Speech Segments in Packet Voice Communications," *IEEE Transactions on Acoustic Speech and Signaling Processing*, vol. 34, No. 3, pp. 342-347, Dec. 1986
- [40] Ondria. J. Wasem, David. J. Goodman, Charles A. Dvorak, Howard G. Page "The Effect of Waveform Substitution on the Quality of PCM Packet Communications," *IEEE Transactions on Acoustic Speech and Signaling Processing*, vol. 36, No. 3, pp. 342-347, March 1988
- [41] H. Sanneck, A. Stenger, K. Bem Youmes, and B. Girod. "A new technique for audio packet loss concealment," In *IEEE Global Internet 1996*, pages 48-52. IEEE, December 1996

- [42] International Telecommunications Union. Recommendation G.723.1: Dual rate speech coder for multimedia communications transmitting at 5.3 and 6.3 Kbits/s, March 1996
- [43] Y. L. Chen and B. S. Chen. "Model-based multi-rate representation of speech signals and its application to recovery of missing packets." *IEEE Transactions on Speech and Audio Processing*, 15(3):220-231, May 1997
- [44] Benjamin W. Wah and Dong Lin. "Transformation-based Reconstruction for Audio Transmissions over the Internet", *Proceedings of the 17th Symposium Reliable Distributed Systems 1998*, IEEE
- [45] <http://java.sun.com/products/java-media/sound/>
- [46] Emanuele Trucco, Alessandro Verri. *Introductory Techniques for 3-D Computer Vision*. Prentice Hall, 1998
- [47] H. Schulzrinne, S. Casner, R. Frederick, V. Jacobson. "RTP: A Transport Protocol for Real-Time Applications". *IETF Network Working Group working group*, January 1996. RFC1889
- [48] H. Schulzrinne. "RTP Profile for Audio and Video Conferences with Minimal Control". *IETF Network Working Group working group*, January 1996. RFC1890
- [49] M. S. Shuster, "Diffusion of network innovation: implications for adoption of Internet services", M.S. thesis, Massachusetts Institute of Technology, Cambridge, MA, June 1998.
- [50] Postel, J., "Transmission Control Protocol", USC/Information Sciences Institute, September 1981, RFC-793
- [51] Postel, J., "User Datagram Protocol", USC/Information Sciences Institute, August 1980, RFC-768