



ΤΕΧΝΟΛΟΓΙΚΟ ΕΚΠΑΙΔΕΥΤΙΚΟ ΙΔΡΥΜΑ ΚΡΗΤΗΣ
ΣΧΟΛΗ ΤΕΧΝΟΛΟΓΙΚΩΝ ΕΦΑΡΜΟΓΩΝ
ΤΜΗΜΑ ΗΛΕΚΤΡΟΝΙΚΗΣ

Κωδικοποίηση φωνής σε πραγματικό χρόνο

Μελέτη και υλοποίηση του αλγορίθμου CELP
στον επεξεργαστή TMS320C6701 της TI

Πτυχιακή εργασία του φοιτητή
Βαμβακά Πέτρου

Επιβλέπων καθηγητής
Δρ. Γεώργιος Γλεντής

Χανιά, Νοέμβριος 2003

Σύνοψη

Η ανάγκη για καλύτερη αξιοποίηση του φάσματος έχει μεγαλώσει δραματικά τα τελευταία χρόνια, ειδικότερα μετά την καθιέρωση της κινητής τηλεφωνίας σαν μέσο καθημερινής επικοινωνίας. Μέχρι τώρα, το κύριο μερίδιο του τηλεπικοινωνιακού φόρτου αποτελεί η φωνή, και παρόλο που αυτό φαίνεται ότι θα αλλάξει στο προσεχές μέλλον, η συνεπακόλουθη αναγκαιότητα για αποτελεσματικότερη συμπίεση της φωνής είναι προφανής. Η εργασία αυτή παρουσιάζει έναν από τους ισχυρότερους αλγόριθμους κωδικοποίησης φωνής, τον CELP (Code Excited Linear Prediction).

Για την εργασία αυτή υλοποιήθηκε σε κώδικα γλώσσας C ένας πλήρης κωδικοποιητής/αποκωδικοποιητής CELP, ο οποίος εκτελέστηκε στον επεξεργαστή TMS320C6701 της πλακέτας EVM της Texas Instruments, με τελικό σκοπό τη λειτουργία του σε πραγματικό χρόνο. Κάποιες βοηθητικές προσομοιώσεις του codec έγιναν και στο περιβάλλον του Matlab.

Τα χαρακτηριστικά του αλγόριθμου που υλοποιήθηκε ήταν η χρήση πλαισίων 240 δειγμάτων φωνής, ανάλυση LPC 10^{ov} βαθμού και η εφαρμογή πρόγνωσης με κωδικοβιβλία σε δύο στάδια, τόσο για την μακροχρόνια πρόγνωση (Pitch prediction) όσο και για την εύρεση της βέλτιστης διέγερσης (Optimal excitation).

Περιεχόμενα

Σύνοψη	ii
Περιεχόμενα	iii
1. Εισαγωγή	1
2. Ανάλυση μέσω σύνθεσης	3
2.1 Εισαγωγή	3
2.2 Χρονικά μεταβαλλόμενα φίλτρα	4
2.2.1 Προγνώστης βραχέως χρόνου – Ανάλυση γραμμικής πρόβλεψης	5
2.2.1.1 Μέθοδος Αυτοσυσχέτισης	8
2.2.1.2 Μέθοδος Συμμεταβλητότητας	10
2.2.2 Μακροπρόθεσμος προγνώστης	11
2.2.2.1 LTP ανοιχτού βρόχου	13
2.2.2.2 LTP κλειστού βρόχου	14
2.3 Σήμα διέγερσης	18
2.3.1 Διέγερση με χρήση κωδικοβιβλίου	19
2.4 Διαδικασία ελαχιστοποίησης σφάλματος	22
2.5 Συνολική επισκόπηση της ανάλυσης μέσω σύνθεσης	23
3. Ο αλγόριθμος CELP	25
3.1 Εισαγωγή	25
3.2 Κωδικοποιητής	25
3.2.1 Γραμμική πρόγνωση βραχέως χρόνου	27
3.2.2 Μετατροπή των συντελεστών πρόγνωσης σε LSF	28
3.2.3 Δημιουργία υποπλαισίων για ανάλυση κωδικοβιβλίων	31
3.2.4 Ανάλυση προσαρμοστικού κωδικοβιβλίου	32
3.2.5 Ανάλυση στοχαστικού κωδικοβιβλίου	35
3.2.6 Συγκεντρωτικά χαρακτηριστικά κωδικοποιητή	37
3.3 Αποκωδικοποιητής	38
4. Θέματα σχετικά με το υλικό	41
4.1 Εισαγωγή	41
4.2 Ο επεξεργαστής TMS320C6701	41
4.2.1 Πυρήνας επεξεργαστή	42

4.2.2 Μνήμη _____	43
4.2.3 Περιφερειακά _____	44
4.3 Η κάρτα EVM _____	45
4.4 Εφαρμογή του κώδικα στην κάρτα EVM _____	46
5. Αποτελέσματα και συμπεράσματα _____	48
5.1 Υλοποίηση κωδικοποιητή – αποκωδικοποιητή σε σειρά _____	48
5.2 Λειτουργία σε πραγματικό χρόνο _____	48
5.3 Προσομοίωση σε Matlab _____	51
5.4 Μελλοντική εργασία _____	52
5.4.1 Περαιτέρω βελτίωση του κώδικα – ενσωμάτωση νέων τιμημάτων στον αλγόριθμο _____	52
5.4.2 Βελτίωση του τελικού φίλτρου για εξάλειψη θορύβου _____	52
5.4.3 Μελέτη ανοχής του κωδικοποιητή σε σφάλματα _____	53
6. Βιβλιογραφία _____	54
Παράρτημα Α : Κώδικας σε γλώσσα C _____	55

1 Εισαγωγή

Η συνεχής επέκταση της χρήσης τηλεπικοινωνιακών υπηρεσιών, τόσο ενσύρματων όσο και ασύρματων, από ολοένα και αυξανόμενες πληθυσμιακές ομάδες τα τελευταία χρόνια, αυξάνει ανάλογα και τις ανάγκες για προσφερόμενο εύρος ζώνης στις μπάντες των τηλεπικοινωνιών. Καθώς όμως το εύρος των συχνοτήτων, ιδιαίτερα με τις υπάρχουσες τεχνολογίες, δεν είναι απεριόριστο, η λύση που ακολουθείται παγκόσμια είναι η περισσότερο αποτελεσματική εκμετάλλευση του φάσματος, ώστε στο ίδιο εύρος να εξυπηρετούνται με την ίδια ή και καλύτερη ποιότητα περισσότεροι χρήστες της υπηρεσίας.

Για την επίτευξη αυτού του στόχου, χρησιμοποιείται από τους τηλεπικοινωνιακούς οργανισμούς ένας συνδυασμός τεχνικών όπως πολυπλεξία, κωδικοποίηση καναλιού, κωδικοποίηση πηγής κλπ. Η παρούσα εργασία ασχολείται με την τελευταία κατηγορία και εξετάζει έναν συγκεκριμένο κωδικοποιητή πηγής, ο οποίος χρησιμοποιείται ευρέως σήμερα σε διάφορες παραλλαγές. Ο κωδικοποιητής αυτός εφαρμόζει τον αλγόριθμο CELP (Code Excited Linear Prediction) και χρησιμοποιείται για συμπίεση αποκλειστικά σήματος φωνής, καθώς η σχεδίασή και η λειτουργία του στηρίζεται σε ορισμένες ιδιότητες της ανθρώπινης ομιλίας και ακουστικής αντίληψης. Η αποτελεσματικότητα της τεχνικής αυτής είναι φανερή από το γεγονός ότι παρέχει δυνατότητα συμπίεσης του σήματος της φωνής από 64Kbps (ο ρυθμός μετάδοσης του προτύπου PCM της τηλεφωνίας) μέχρι και τα 4.8Kbps (ο ρυθμός μετάδοσης του αλγορίθμου DoD PFS1016, που εξετάζεται στην εργασία αυτή και ο οποίος δίνει τη μέγιστη συμπίεση από τους αλγορίθμους της οικογένειας CELP). Με λίγα λόγια, το άμεσο κέρδος για τη μετάδοση φωνής είναι ο υποδεκαπλασιασμός τουλάχιστον του απαιτούμενου φασματικού εύρους σε σχέση με ένα κλασικό PCM σύστημα, χωρίς μεγάλο συμβιβασμό στην ποιότητα του μεταδιδόμενου σήματος.

Στα επόμενα κεφάλαια, θα αναλυθούν διεξοδικά οι αρχές λειτουργίας του αλγορίθμου καθώς και ο κώδικας και το υλικό (hardware) που χρησιμοποιήθηκε. Συγκεκριμένα, το κεφάλαιο 2 εξετάζει γενικά το θεωρητικό υπόβαθρο πάνω στο οποίο στηρίζεται η λειτουργία του αλγορίθμου CELP, δηλαδή την τεχνική Ανάλυσης

μέσω Σύνθεσης (Analysis-by-Synthesis). Στο κεφάλαιο 3 αναλύεται κάθε τμήμα ξεχωριστά ως προς τον ακριβή τρόπο υλοποίησής του για τον αλγόριθμο PFS1016, ενώ γίνεται συσχέτιση των τμημάτων με τις αντίστοιχες ρουτίνες κώδικα γλώσσας C με τις οποίες έγινε η υλοποίηση. Το κεφάλαιο 4 παρουσιάζει το υλικό πάνω στο οποίο εφαρμόστηκε ο κώδικας, δηλαδή την κάρτα EVM6x01 πάνω στην οποία βρίσκεται ο επεξεργαστής TMS320C6701 της Texas Instruments. Επίσης εξετάζονται θέματα υλοποίησης σχετικά με το υλικό, όπως ο τρόπος εισόδου/εξόδου του σήματος φωνής, η ψηφιοποίηση κλπ. Το κεφάλαιο 5 τέλος, παρουσιάζει τα τελικά συμπεράσματα της εργασίας αυτής. Ακολουθεί σε παράρτημα (παράρτημα A) ο πλήρης κώδικας της υλοποίησης.

2 Ανάλυση μέσω σύνθεσης

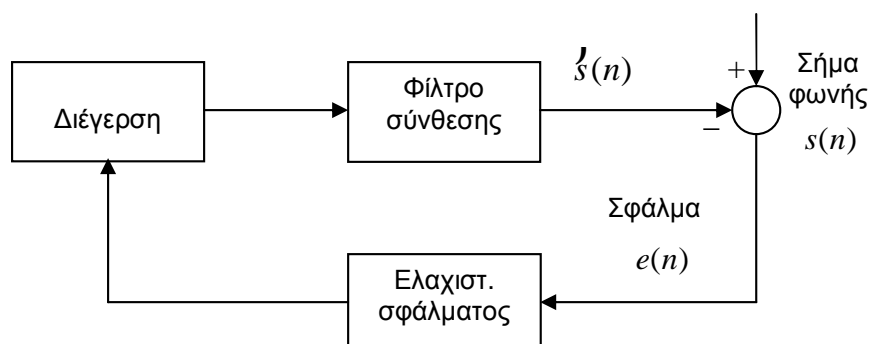
2.1 Εισαγωγή

Η βασική τεχνική πάνω στην οποία στηρίζεται ο αλγόριθμος CELP, όπως και οι περισσότεροι άλλοι αλγόριθμοι ρυθμών μετάδοσης κάτω από τα 9.6Kbps, είναι η ανάλυση μέσω σύνθεσης (Analysis by Synthesis – AbS).

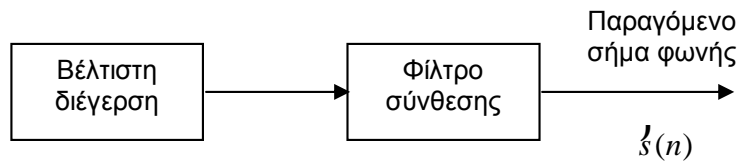
Η κύρια ιδέα πίσω από την ανάλυση μέσω σύνθεσης είναι η εξής: Αρχικά υποθέτεται ένα θεωρητικό μοντέλο παραγωγής της φωνής. Το μοντέλο αυτό ελέγχεται από μια σειρά παραμέτρων, οι οποίες καθορίζουν την απόκρισή του. Το ζητούμενο είναι, μεταβάλλοντας τις παραμέτρους, να επιτευχθεί μια απόκριση του μοντέλου τέτοια ώστε να προσομοιάζει το πραγματικό σήμα με όσο το δυνατό μικρότερο σφάλμα. Τότε, οι συγκεκριμένες παράμετροι του μοντέλου θεωρούνται σαν παράμετροι και του πραγματικού σήματος. Για την εύρεση της βέλτιστης ομάδας παραμέτρων, συνήθως εφαρμόζεται ένας αλγόριθμος δοκιμών και υπολογισμού του σφάλματος μέχρι την ελαχιστοποίησή του.

Τα βασικά τμήματα της διαδικασίας αυτής περιγράφονται από τον κλειστό βρόχο του Σχήματος 2.1, όπου απεικονίζεται ένας κωδικοποιητής φωνής που υλοποιεί την τεχνική ανάλυσης μέσω σύνθεσης. Στο Σχήμα 2.2 εμφανίζεται ο αντίστοιχος αποκωδικοποιητής. Στο Σχήμα 2.1 υπάρχουν τρεις βαθμίδες όπου οι παράμετροι είναι μεταβλητές:

- (1) Χρονικά μεταβαλλόμενο φίλτρο
- (2) Σήμα διέγερσης
- (3) Διαδικασία ελαχιστοποίησης σφάλματος



Σχήμα 2.1 Κωδικοποιητής AbS



Σχήμα 2.2 Αποκωδικοποιητής AbS

Η λειτουργία του κωδικοποιητή του σχήματος είναι, σε γενικές γραμμές, η ακόλουθη: Αρχικά, από το εισερχόμενο σήμα εξάγονται οι συντελεστές του φίλτρου, οι οποίοι οφείλουν να μεταβάλλονται παρακολουθώντας τις μεταβολές του σήματος. Το επόμενο στάδιο είναι η αναζήτηση του κατάλληλου σήματος διέγερσης, η οποία γίνεται με επαναληπτικό τρόπο. Για κάθε ακολουθία διέγερσης που δοκιμάζεται, αφού αυτή μορφοποιηθεί από το φίλτρο βάσει των συντελεστών που βρέθηκαν, υπολογίζεται το υποκειμενικό (σύμφωνα με επιλεγμένο κριτήριο) σφάλμα. Κατόπιν επιλέγεται σαν βέλτιστη η ακολουθία που ελαχιστοποιεί το σφάλμα. Οι παράμετροι αυτές (οι συντελεστές του φίλτρου και η βέλτιστη διέγερση) θεωρούνται επαρκείς ώστε με βάση αυτούς να αναπαραχθεί το σήμα φωνής στον αποκωδικοποιητή.

Παρακάτω θα επιχειρηθεί μια μελέτη καθεμιάς από τις βαθμίδες στη γενική της μορφή, ενώ μια πιο συγκεκριμένη ανάλυση, όσον αφορά τα ακριβή μέρη που απαρτίζουν τον αλγόριθμο CELP καθώς και τον τρόπο υλοποίησής τους, θα συζητηθούν στο επόμενο κεφάλαιο.

2.2 Χρονικά μεταβαλλόμενα φίλτρα

Καθώς είναι αναγκαίο να μεταβάλλονται οι συντελεστές του φίλτρου σύνθεσης σύμφωνα με τις μεταβολές του εισερχόμενου σήματος φωνής, στις περισσότερες περιπτώσεις κωδικοποιητών η βαθμίδα του φίλτρου όπως εμφανίζεται στο Σχήμα 2.1 αποτελείται από δύο γραμμικούς προγνώστες, οι οποίοι είναι: Ο προγνώστης βραχέως χρόνου STP (Short-term predictor) ή, κατά την ορολογία που χρησιμοποιείται περισσότερο στη βιβλιογραφία, προγνώστης LPC (Linear Predictive Coding) και ο μακροπρόθεσμος προγνώστης LTP (Long-term predictor) ή προγνώστης θεμελιώδους συχνότητας (Pitch predictor).

Σε αρκετές περιπτώσεις κωδικοποιητών, ο δεύτερος προγνώστης δεν εμφανίζεται σαν ξεχωριστή μονάδα, καθώς η λειτουργία του είναι δυνατό να θεωρηθεί σαν μέρος της βαθμίδας διέγερσης.

2.2.1 Προγνώστης βραχέως χρόνου – Ανάλυση γραμμικής πρόβλεψης

Ο προγνώστης STP εκμεταλλεύεται τη συσχέτιση βραχέως χρόνου του εισερχόμενου σήματος φωνής (short-term correlation) για να εξάγει συντελεστές για το φίλτρο τέτοιους ώστε το φίλτρο να διαμορφώνει μια βασική φασματική περιβάλλουσα του σήματος.

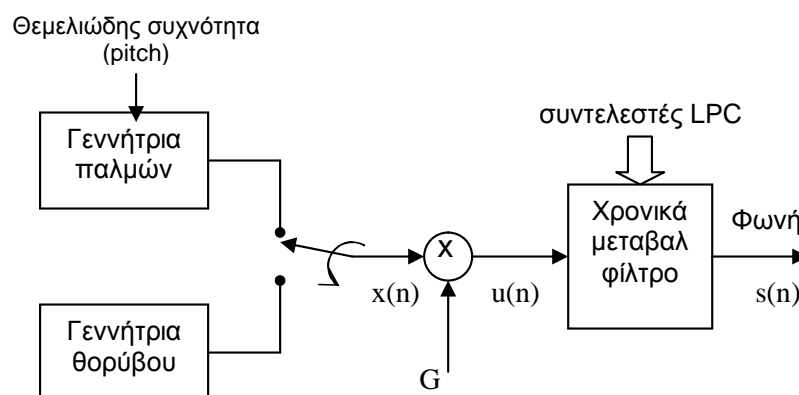
Η λειτουργία του STP στηρίζεται στην ανάλυση γραμμικής πρόβλεψης ή ανάλυση LPC. Για να επιτευχθεί η ανάλυση LPC, πρέπει να γίνουν αρχικά δύο παραδοχές:

A) Υπάρχει ένα θεωρητικό μοντέλο για την παραγωγή της φωνής και είναι γνωστά τα χαρακτηριστικά του. Στις περισσότερες περιπτώσεις, χρησιμοποιείται το ψηφιακό μοντέλο παραγωγής της φωνής (source-filter model), του οποίου ένα απλοποιημένο διάγραμμα φαίνεται στο Σχήμα 2.3. Σύμφωνα με το μοντέλο αυτό, η διέγερση μπορεί να παρασταθεί είτε σαν ακολουθία κρουστικών παλμών (όταν η ομιλία αποτελείται από εύφωνους ήχους) είτε σαν θόρυβος κανονικής κατανομής (όταν η ομιλία αποτελείται από άφωνους ήχους). Η συνδυασμένη συνεισφορά στη διαμόρφωση του ήχου της γλωττικής ροής, του φωνητικού σωλήνα και της ακτινοβολίας στα χείλη, παρίσταται με ένα χρονικά μεταβαλλόμενο φίλτρο με συνάρτηση μεταφοράς που προσεγγίζεται από την ακόλουθη:

$$H(z) = \frac{G}{A(z)} \quad (2.1)$$

όπου

$$A(z) = 1 - \sum_{j=1}^p a_j z^{-j} \quad (2.2)$$



Σχήμα 2.3 Το ψηφιακό μοντέλο παραγωγής της φωνής

Μετασχηματίζοντας την 2.2 στο πεδίο του διακριτού χρόνου, προκύπτει η διαφορική εξίσωση της ανάλυσης γραμμικής πρόβλεψης:

$$s(n) = Gx(n) + \sum_{j=1}^p a_j s(n-j) \quad (2.3)$$

Η εξίσωση 2.3 δηλώνει ότι η τιμή της παρούσας εξόδου $s(n)$ μπορεί να προβλεφθεί αθροίζοντας τη σταθμισμένη παρούσα είσοδο $Gx(n)$ και ένα άθροισμα p – προηγούμενων σταθμισμένων τιμών εξόδου, όπου ο παράγοντας G παριστάνει το κέρδος που πρέπει να εφαρμοστεί ώστε η ισχύς του σήματος του μοντέλου να γίνει ίση με την ισχύ του πραγματικού σήματος.

B) Η δεύτερη παραδοχή που πρέπει να γίνει για να είναι δυνατή η χρήση της ανάλυσης LPC, είναι ότι το σήμα φωνής που χρησιμοποιείται στο παραπάνω μοντέλο είναι χρονικά στάσιμο. Προφανώς, κάτι τέτοιο δεν ισχύει, αφού η ομιλία είναι σήμα συνεχώς μεταβαλλόμενο στο χρόνο, με φαινομενικά τυχαίο τρόπο. Για το λόγο αυτό, επιλέγεται να γίνει επεξεργασία του εισερχόμενου σήματος κατά τμήματα τέτοιου μεγέθους που να ικανοποιούν την απαίτηση για στασιμότητα.

Με βάση αυτές τις παραδοχές, η ανάλυση γραμμικής πρόβλεψης καλείται να υπολογίσει τους συντελεστές a_j , $j = 1, \dots, p$ της εξίσωσης 2.3 με δεδομένες τις τιμές του σήματος $s(n)$.

Η διαδικασία εκκινείται με την εκτίμηση των συντελεστών, οι οποίοι είναι στη γενική περίπτωση διαφορετικοί από τους πραγματικούς συντελεστές του μοντέλου. Αν \bar{a}_j είναι η εκτίμηση των πραγματικών συντελεστών, τότε το σφάλμα δίνεται από την

$$e(n) = s(n) - \sum_{j=1}^p \bar{a}_j s(n-j) \quad (2.4)$$

όπου το άθροισμα παριστάνει την εκτίμηση για το σήμα που γίνεται βάσει της εκτίμησης των συντελεστών. Οι πραγματικοί συντελεστές προσεγγίζονται από την εκτίμηση με χρήση κάποιου κριτηρίου ελαχιστοποίησης σφάλματος. Το πιο διαδεδομένο είναι το κριτήριο *ελαχιστοποίησης του τετραγωνικού σφάλματος* (Least Mean Squared Error Criteria – LMSE). Το συνολικό τετραγωνικό σφάλμα είναι:

$$E = E\{e^2(n)\} = E\left\{\left[s(n) - \sum_{j=1}^p \bar{a}_j s(n-j)\right]^2\right\} \quad (2.5)$$

Για την ελαχιστοποίησή του αναζητούνται τα ακρότατα, μηδενίζοντας τις μερικές παραγώγους του ως προς \bar{a}_j για $j = 1, \dots, p$, δηλ.

$$\frac{\partial E}{\partial \bar{a}_j} = 0, i = 1, \dots, p$$

Η παράγωγος ως προς τον i -οστό συντελεστή θα είναι

$$E\left\{2\left[s(n) - \sum_{j=1}^p \bar{a}_j s(n-j)\right] \cdot [-s(n-i)]\right\} = 0$$

ή, τελικά

$$E\left\{\left[s(n) - \sum_{j=1}^p \bar{a}_j s(n-j)\right] \cdot s(n-i)\right\} = 0 \quad (2.6)$$

για $i = 1, \dots, p$.

Λόγω της ορθογωνικότητας των $e(n)$ ως προς τις τιμές του σήματος $s(n-i)$ για $i = 1, \dots, p$, μπορεί να ξαναγραφεί η 2.6 ως εξής

$$\sum_{j=1}^p \bar{a}_j f_n(i, j) = f_n(i, 0) \quad \text{για } i = 1, \dots, p \quad (2.7)$$

όπου

$$f_n(i, j) = E\{s(n-i)s(n-j)\} \quad (2.8)$$

Για την εξαγωγή της 2.7, πάρθηκε υπόψη η δεύτερη παραδοχή, ότι δηλαδή το σήμα θεωρείται χρονικά αμετάβλητο, ή αλλιώς ότι για την ανάλυση χρησιμοποιούνται μικρά τμήματα του εισερχόμενου σήματος. Ουσιαστικά δηλαδή, η σχέση 2.8 μπορεί να αντικατασταθεί από αθροίσματα πεπερασμένων όρων,

$$f_n(i, j) = \sum_m s_n(m-i)s_n(m-j), \text{ για } i = 1, \dots, p, j = 0, \dots, p \quad (2.9)$$

όπου $s_n(m)$ τμήμα καθορισμένου μήκους του εισερχόμενου σήματος.

Για την επίλυση της LPC ανάλυσης, την εύρεση δηλαδή των εκτιμήσεων \bar{a}_j , υπάρχουν από το σημείο αυτό δύο επικρατούσες μέθοδοι, η μέθοδος Αυτοσυσχέτισης (Auto-correlation) και η μέθοδος Συμμεταβλητότητας (Covariance) από τις οποίες θα αναλυθεί εκτενώς η πρώτη, η οποία χρησιμοποιείται εκτενέστερα τόσο στη βιβλιογραφία όσο και στην υλοποίηση που παρουσιάζεται στην εργασία αυτή.

2.2.1.1 Μέθοδος Αυτοσυσχέτισης

Όπως αναφέρθηκε παραπάνω, η ανάλυση γραμμικής πρόβλεψης γίνεται πάνω σε τμήματα του εισερχόμενου σήματος φωνής. Η μέθοδος αυτοσυσχέτισης υποθέτει ότι αν N είναι το μήκος του εισερχόμενου τμήματος του σήματος (το πλήθος των δειγμάτων του τμήματος) τότε το σήμα είναι μηδενικό σε όλο το μήκος του εκτός από το επιλεγμένο τμήμα για το οποίο γίνεται η ανάλυση. Ή με άλλα λόγια, τα δείγματα του σήματος είναι μηδενικά παντού εκτός από το διάστημα $0 \leq m \leq N - 1$.

Το ζητούμενο αυτό ικανοποιείται με την εφαρμογή μιας ολισθαίνουσας συνάρτησης «παραθύρου» (window function) στο εισερχόμενο σήμα. Στην ιδανική περίπτωση, η απόκριση συχνότητας μιας τέτοιας συνάρτησης έχει έναν άπειρα στενό κύριο λοβό, ώστε η ανάλυση του παραθύρου να είναι η μέγιστη ενώ ταυτόχρονα δεν έχει καθόλου δευτερεύοντες λοβούς. Οι πραγματικές συναρτήσεις αποτελούν ένα συμβιβασμό μεταξύ των δύο παραπάνω χαρακτηριστικών. Μερικές από τις συναρτήσεις αυτές είναι οι ακόλουθες:

Τετραγωνικό παράθυρο:

$$w(n) = \begin{cases} 1, & 0 \leq n \leq N - 1 \\ 0, & \text{αλλού} \end{cases}$$

Παράθυρο Blackman:

$$w(n) = \begin{cases} 0.42 - 0.5 \cos\left(2\pi \frac{n}{N-1}\right) + 0.08 \cos\left(2\pi \frac{2n}{N-1}\right), & 0 \leq n \leq N - 1 \\ 0, & \text{αλλού} \end{cases}$$

Παράθυρο Hamming:

$$w(n) = \begin{cases} 0.54 - 0.46 \cos\left(2\pi \frac{n}{N-1}\right), & 0 \leq n \leq N - 1 \\ 0, & \text{αλλού} \end{cases}$$

Άλλες συναρτήσεις που αναφέρονται στη βιβλιογραφία, είναι το παράθυρο Hanning, το παράθυρο Bartlett, το παράθυρο Kaiser κ.α.

Η χρήση του τετραγωνικού παραθύρου απορρίπτεται για πρακτικές εφαρμογές, καθώς εισάγει υπολογίσιμες παραμορφώσεις στο πεδίο της συχνότητας (frequency leakage) έχοντας μια απόκριση συχνότητας με στενό μεν κεντρικό λοβό αλλά και σημαντική παρουσία δευτερευόντων λοβών. Για το λόγο αυτό χρησιμοποιούνται παράθυρα με ομαλότερη φασματική απόκριση, όπως για

παράδειγμα το παράθυρο Hamming, το οποίο χρησιμοποιείται ευρέως σε υλοποιήσεις πραγματικού χρόνου.

Με την εφαρμογή του παραθύρου στο εισερχόμενο σήμα φωνής, δημιουργείται ένα πλαίσιο (frame) για την ανάλυση γραμμικής πρόβλεψης. Καθώς το παράθυρο ολισθαίνει, το προκύπτων σήμα είναι δυνητικά άπειρο σε μήκος αλλά μηδενικό παντού έξω από το παράθυρο, επιτρέποντας έτσι τον υπολογισμό της συνάρτησης αυτοσυσχέτισης του σήματος. Το i -οστό πλαίσιο ανάλυσης δίνεται από

$$s_n(n) = s(n)w_i(n) \quad (2.10)$$

όπου το $w_i(n)$ παριστά το i -οστό παράθυρο.

Μπορεί τότε να οριστεί η συνάρτηση αυτοσυσχέτισης βραχέως χρόνου, ως εξής:

$$f_n(i, j) = R_n(|i - j|), \text{ για } i = 1, \dots, p, j = 0, \dots, p \quad (2.11)$$

όπου

$$R_n(j) = \sum_{m=0}^{N-1-j} s_n(m)s_n(m+j) \quad (2.12)$$

Χρησιμοποιώντας τις παραπάνω σχέσεις, η 2.7 μπορεί να εκφραστεί σαν

$$\sum_{j=1}^p \bar{a}_j R_n(|i - j|) = R_n(i), \quad 1 \leq i \leq p \quad (2.13)$$

Η 2.13 περιγράφει ένα σύστημα εξισώσεων, οι οποίες είναι γνωστές και σαν κανονικές εξισώσεις και οι οποίες μπορούν να γραφτούν σε μορφή πινάκων όπως παρακάτω:

$$\begin{bmatrix} R_n(0) & R_n(1) & \mathbf{L} & R_n(p-1) \\ R_n(1) & \mathbf{L} & \mathbf{L} & R_n(p-2) \\ \mathbf{M} & \mathbf{M} & \mathbf{O} & \mathbf{M} \\ R_n(p-1) & \mathbf{L} & \mathbf{L} & R_n(0) \end{bmatrix} \begin{bmatrix} \bar{a}_1 \\ \bar{a}_2 \\ \mathbf{L} \\ \bar{a}_n \end{bmatrix} = \begin{bmatrix} R_n(1) \\ R_n(2) \\ \mathbf{L} \\ R_n(p) \end{bmatrix} \quad (2.13)$$

Ο $p \times p$ πίνακας των R_n (συντελεστών της αυτοσυσχέτισης – autocorrelation coefficients), είναι συμμετρικός και τα στοιχεία κάθε διαγωνίου είναι ίσα. Κάθε πίνακας με αυτές τις ιδιότητες αναφέρεται ως πίνακας *Toeplitz*. Για την εύρεση των ζητούμενων συντελεστών \bar{a}_j , πρέπει προφανώς να επιλυθεί το σύστημα των κανονικών εξισώσεων. Ένας από τους περισσότερο διαδεδομένους τρόπους είναι ο αλγόριθμος των Levinson-Durbin. Πρόκειται για μια αναδρομική διαδικασία, η οποία περιγράφεται στον Πίνακα 2.1.

Οι συντελεστές k_i που υπολογίζονται κατά τη διάρκεια της αναδρομής, περιέχουν την ίδια πληροφορία με τους συντελεστές αυτοσυσχέτισης (R_n) και

αναφέρονται στη βιβλιογραφία σαν *συντελεστές ανάκλασης (reflection coefficients)* ή *μερικοί συντελεστές συσχέτισης (partial correlation coefficients – PARCORs)*.

2.2.1.2 Μέθοδος Συμμεταβλητότητας

Κατά τη μέθοδο συμμεταβλητότητας, η συνάρτηση παραθύρου δεν εφαρμόζεται απ' ευθείας στο εισερχόμενο σήμα φωνής, αλλά στο σήμα σφάλματος πρόβλεψης $e(n)$, με αποτέλεσμα οι προκύπτουσες εξισώσεις να μην καταλήγουν σε συμμετρικό πίνακα Toeplitz. Έτσι, η λύση δεν μπορεί να εξαχθεί με το ίδιο αποτελεσματικά όσο με τη χρήση του αλγορίθμου Levinson-Durbin για τη μέθοδο αυτοσυσχέτισης. Παρ' όλα αυτά, έχουν αναπτυχθεί ικανές μέθοδοι όπως η *Ανάλυση Cholesky* (Cholesky Decomposition).

Το πλεονέκτημα της μεθόδου αυτής έναντι της μεθόδου αυτοσυσχέτισης είναι ότι εφ' όσον η συνάρτηση παραθύρου δεν εφαρμόζεται στο ίδιο το σήμα αλλά στο προκύπτων σφάλμα, το σήμα εισέρχεται χωρίς παραμόρφωση στο στάδιο της ανάλυσης, με συνεπακόλουθο μια πιθανή καλύτερη απόδοση της ανάλυσης γραμμικής πρόβλεψης.

Πίνακας 2.1 Η αναδρομή Levinson - Durbin

1. Αρχικοποίηση του αλγορίθμου

$$E_n^0 = R_n^0$$

2. Για $i = 1, 2, \dots, p$, αναδρομική επίλυση των

$$\text{i) } k_i = \left[R_n(i) - \sum_{j=1}^{i-1} \bar{a}_j^{i-1} R_n(i-j) \right] / E_n^{i-1}$$

$$\text{ii) } \bar{a}_i^i = k_i$$

$$\text{iii) } \bar{a}_j^i = \bar{a}_j^{i-1} - k_i \bar{a}_{i-j}^{i-1} \quad 1 \leq j \leq i-1$$

$$\text{iv) } E_n^i = (1 - k_i^2) E_n^{i-1}$$

3. Οι ζητούμενοι συντελεστές είναι

$$\bar{a}_j = \bar{a}_j^p$$

2.2.2 Μακροπρόθεσμος προγνώστης

Αν επιχειρηθεί να γίνει η αντίστροφη διαδικασία από αυτήν που περιγράφηκε στην προηγούμενη παράγραφο, να γίνει δηλαδή εφαρμογή στο εισερχόμενο σήμα φωνής του αντίστροφου φίλτρου γραμμικής ανάλυσης που περιγράφεται από τη σχέση

$$H^{-1}(z) = 1 - \sum_{j=1}^p a_j z^{-j} \quad (2.14)$$

που είναι η αντίστροφη της 2.1, η παρατήρηση θα ήταν ότι από το σήμα έχουν αφαιρεθεί οι συσχετίσεις βραχέως χρόνου μεταξύ των γειτονικών δειγμάτων του σήματος. Ή ισοδύναμα, ότι έχει αφαιρεθεί η φασματική περιβάλλουσα του σήματος.

Παρ' όλα αυτά, η μορφή του σήματος θα απείχε αρκετά από το να είναι επίπεδη φασματικά, να έχει ομοιόμορφη φασματική πυκνότητα ισχύος. Θα παρατηρούνταν ειδικότερα αιχμηροί περιοδικοί παλμοί, που οφείλονται προφανώς στο είδος της διέγερσης που έχει υποθεθεί (περιοδικοί κρουστικοί παλμοί και λευκός θόρυβος). Τα παραπάνω συμβαίνουν καθώς η ανάλυση LPC είναι σχεδιασμένη να επεξεργάζεται ικανοποιητικά μόνο τις βραχυχρόνιες συσχετίσεις του σήματος. Στο σήμα όμως υπάρχουν και μακροχρόνιες συσχετίσεις. Άρα, για την μοντελοποίηση της περιοδικής (μακροπρόθεσμης) δομής του σήματος απαιτείται μια ακόμα βαθμίδα προγνώστη, ο μακροπρόθεσμος προγνώστης LTP, ο οποίος εκμεταλλεύεται την συσχέτιση μεταξύ δειγμάτων του σήματος που απέχουν μία ή περισσότερες περιόδους της θεμελιώδους συχνότητας (*pitch*).

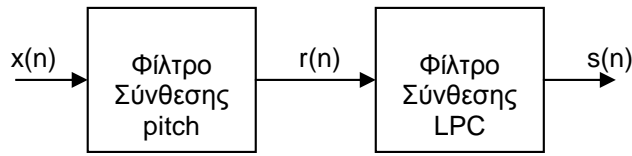
Όπως και στην ανάλυση LPC, η μακροχρόνια πρόγνωση μπορεί να παρασταθεί σαν συνάρτηση μεταφοράς «μόνο πόλων» ενός φίλτρου:

$$P(z) = \frac{1}{1 - \sum_{j=-1}^I b_j z^{-(j+T)}} \quad (2.15)$$

όπου T είναι η θεμελιώδης περίοδος (*pitch*) ή υστέρηση (*lag / delay*), b_j είναι οι συντελεστές κέρδους της θεμελιώδους συχνότητας, που αντανakλούν την συσχέτιση μεταξύ των μακρινών δειγμάτων του σήματος και I ο αριθμός που δίνει τα taps του φίλτρου (π.χ. για $I = 1$ το φίλτρο θα έχει 3 taps).

Λαμβάνοντας υπόψη και την ανάλυση LPC και ειδικότερα τη σχέση 2.3, το συνδυασμένο μοντέλο ανάλυσης στο πεδίο του διακριτού χρόνου, όπως εμφανίζεται στο Σχήμα 2.4, θα δίνεται από τη διαφορική εξίσωση

$$s(n) = Gx(n) + \sum_{j=-1}^I b_j r(n-T-j) + \sum_{j=1}^p a_j s(n-j) \quad (2.16)$$



Σχήμα 2.4 Συνδυασμένο μοντέλο ανάλυσης

όπου $r(n)$ είναι οι παρελθούσες τιμές της εισόδου, δηλ. του σήματος διέγερσης $x(n)$. Με χρήση παρόμοιας μεθοδολογίας όπως και για την ανάλυση LPC, το ζητούμενο εδώ είναι να βρεθούν οι εκτιμήσεις (β_j, D, \bar{a}_j) για τις παραμέτρους (b_j, T, a_j) . Το σφάλμα σε αυτήν την περίπτωση θα είναι

$$e(n) = s(n) - \sum_{j=1}^l b_j r(n - D - j) - \sum_{j=1}^p \bar{a}_j s(n - j) \quad (2.17)$$

Λόγω της ύπαρξης της υστέρησης D , η λύση για ελαχιστοποίηση του μέσου τετραγωνικού σφάλματος δεν είναι τόσο απλή όσο για την περίπτωση της ανάλυσης LPC. Αναφέρονται δύο προσεγγίσεις:

- (a) *Βελτιστοποίηση σε έναν κύκλο (One-Shot Optimization) ή LTP ανοιχτού βρόχου.* Αρχικά υπολογίζονται μέσω LPC οι συντελεστές a_j . Με την υπόθεση ότι το σήμα $r(n)$ που προκύπτει από το αντίστροφο φίλτρο LPC περιέχει παραπλήσια πληροφορία περιοδικότητας με το εισερχόμενο σήμα $s(n)$, είναι δυνατή η εξαγωγή των ζητούμενων (β_j, D) από το ενδιάμεσο $r(n)$. Η διαδικασία βελτιστοποιείται όταν η υστέρηση D είναι μεγαλύτερη από το μήκος του πλαισίου ανάλυσης, δηλ. $D > L$.
- (b) *Επαναληπτική προσέγγιση (Iterative Sequential Approach) ή LTP κλειστού βρόχου.* Αρχικά εκτελείται μια διαδικασία ίδια με την βελτιστοποίηση σε έναν κύκλο, αλλά στις επόμενες επαναλήψεις η βραχυχρόνια πρόγνωση βελτιστοποιείται βάσει των συντελεστών που υπολογίστηκαν στην προηγούμενη επανάληψη. Με τον τρόπο αυτό, το $r(n)$ μεταβάλλεται σε κάθε επανάληψη, ώστε η πληροφορία περιοδικότητας που περιέχει να προσεγγίζει αυτήν του πραγματικού σήματος. Η διαδικασία συνεχίζεται μέχρι την επίτευξη κάποιας προκαθορισμένης τιμής κατωφλίου ή για μέχρι τη συμπλήρωση ορισμένου αριθμού επαναλήψεων.

Σε κάθε περίπτωση πάντως, το πλαίσιο ανάλυσης που χρησιμοποιείται για την LTP είναι τμήμα του πλαισίου (υποπλαίσιο – sub frame) της STP πρόγνωσης καθώς η LTP είναι ιδιαίτερα επίπονη υπολογιστικά και η τμηματική εκτέλεσή της μειώνει τον υπολογιστικό φόρτο.

Είναι σημαντικό επίσης να ειπωθεί ότι το μήκος του παραθύρου (σε δείγματα) που χρησιμοποιείται για την μακροπρόθεσμη πρόγνωση να είναι αρκετά μεγαλύτερο από το μήκος του υποπλασίου ανάλυσης, σε αντίθεση με την πρόγνωση βραχέως χρόνου όπου είναι ίσα. Αυτό οφείλεται στο ότι η υστέρηση D μπορεί να μεταβάλλεται μεταξύ ενός ελάχιστου (D_{\min} , περίπου 16 δείγματα) και ενός μέγιστου (D_{\max} , περίπου 150 δείγματα). Άρα εδώ το παράθυρο θα πρέπει να έχει μήκος $L + D_{\max}$, όπου L το μήκος του υποπλασίου, έτσι ώστε να περιέχει τουλάχιστον μια πλήρη περίοδο της θεμελιώδους αρμονικής.

2.2.2.1 LTP ανοιχτού βρόχου

Η μέθοδος βελτιστοποίησης σε έναν κύκλο μπορεί να συνοψιστεί ως εξής:

Αφαιρώντας την επίδραση της πρόγνωσης βραχέως χρόνου, η 2.17 γίνεται

$$e(n) = r(n) - \sum_{j=-1}^I b_j r(n-D-j) \quad (2.18)$$

οπότε οι εκτιμήσεις β_j μπορούν να εξαχθούν με τη χρήση του κριτηρίου ελαχιστοποίησης του τετραγωνικού σφάλματος

$$E = E\{e^2(n)\} = E\left\{\left[r(n) - \sum_{j=-1}^I b_j r(n-D-j)\right]^2\right\} \quad (2.19)$$

Αντικαθιστώντας με άθροισμα πεπερασμένων όρων, καθώς όπως και πριν η ανάλυση γίνεται τμηματικά πάνω στο εισερχόμενο σήμα (ανά πλαίσιο):

$$E = \sum_m e_n^2(m) = \sum_m \left[r_n(m) - \sum_{j=-1}^I b_j r_n(m-D-j) \right]^2 \quad (2.20)$$

Μηδενίζοντας τις μερικές παραγώγους ως προς τους συντελεστές β_j , προκύπτει τελικά για τον i -οστό όρο

$$\sum_{j=-1}^I b_j V(i, j) = R(D+i, 0) \quad -I \leq i \leq I \quad (2.21)$$

όπου

$$R(D+i, 0) = \sum_{m=0}^{L-1} r(m-D-i)r(m) \quad (2.22)$$

και

$$V(i, j) = \sum_{m=0}^{L-1} r(m-D-i)r(m-D-j), \quad -I \leq i \leq I, \quad -I \leq j \leq I \quad (2.23)$$

Η επίλυση μπορεί να γίνει γράφοντας την 2.21 σε μορφή πινάκων, και αντιστρέφοντας τον $[V(i, j)]$.

Για να είναι δυνατή όμως η χρήση της παραπάνω μεθοδολογίας απαιτείται να βρεθεί αρχικά η υστέρηση τ . Για την ανίχνυσή της αναφέρονται διάφοροι αλγόριθμοι όπως *Αυτοσυσχέτισης*, *Cepstrum*, *Μέγιστης πιθανότητας* (Maximum Likelihood) κ.α.

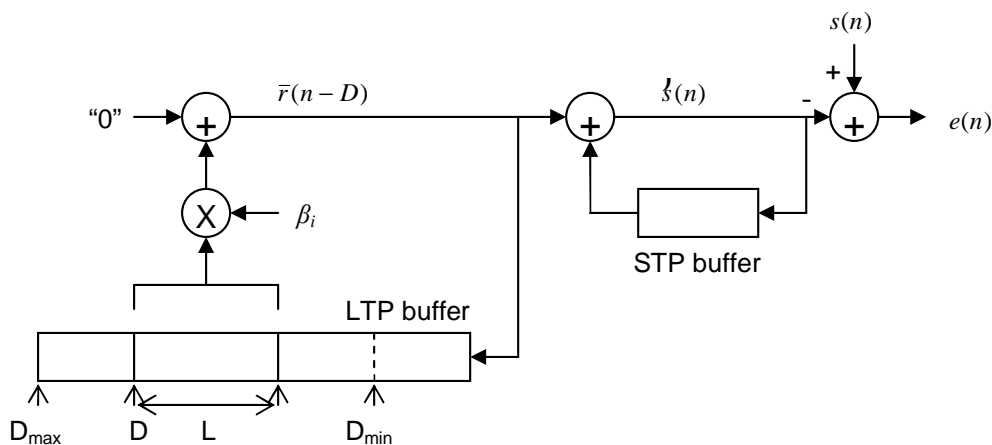
2.2.2.2 LTP κλειστού βρόχου

Στην LTP κλειστού βρόχου (Closed-loop) το κυρίως ζητούμενο δεν είναι η ελαχιστοποίηση του σφάλματος πρόβλεψης $r(n)$ του προγνώστη όπως έγινε παραπάνω, αλλά του σφάλματος μεταξύ του σταθμισμένου εισερχόμενου σήματος και της προκύπτουσας συνθετικής φωνής. Δηλ. αναφορικά με το Σχήμα 2.5, αυτό που ενδιαφέρει είναι η ελαχιστοποίηση του σφάλματος $e(n)$. Ένας τρόπος για να επιτευχθεί αυτό, θα ήταν η εξαντλητική αναζήτηση των ζητούμενων συντελεστών της πρόγνωσης ταυτόχρονα με τη βέλτιστη διέγερση. Κάτι τέτοιο όμως θα επιβάρυνε υπερβολικά μια υλοποίηση σε υπολογιστικό φόρτο. Για το λόγο αυτό, αναφέρονται υποβέλτιστες λύσεις, όπως η διαίρεση της αναζήτησης σε δύο διαδοχικά στάδια.

Υποθέτοντας λοιπόν ότι για το στάδιο αυτό η διέγερση είναι μηδενική και έχουν υπολογιστεί οι συντελεστές της LPC ανάλυσης που ελαχιστοποιούν το $e(n)$, αναζητούνται οι συντελεστές της LTP. Θα είναι

$$\hat{s}(n) = \sum_{i=-I}^I b_i \sum_{k=0}^n \bar{r}(n-k-D-i)h_w(k) \quad (2.24)$$

όπου $h_w(k)$ είναι η σταθμισμένη κρουστική απόκριση του φίλτρου LPC, D η υστέρηση (pitch delay) και L το μήκος του πλαισίου ανάλυσης. Οι συντελεστές που υπολογίστηκαν μέσω της LPC ανάλυσης έχουν σταθμιστεί για λόγους καλύτερης προσαρμογής στην ανθρώπινη ακουστική αντίληψη, με χρήση κάποιου κριτηρίου όπως η *ελαχιστοποίηση βάσει της αντιληπτικότητας* (perceptually based minimisation), το οποίο θα παρουσιαστεί εκτενέστερα στην παράγραφο 2.4.



Σχήμα 2. 5 LTP κλειστού βρόχου

Από την 2.24 εξάγεται το σταθμισμένο τετραγωνικό σφάλμα

$$E_w(D) = \sum_{n=0}^{L-1} e^2(n) = \sum_{n=0}^{L-1} [s(n) - \hat{s}(n)]^2 \quad (2.25)$$

όπου $s(n)$ είναι το σταθμισμένο σήμα εισόδου από το οποίο έχει αφαιρεθεί η συνεισφορά μηδενικής εισόδου του φίλτρου LPC. Μηδενίζοντας τις μερικές παραγώγους όπως προηγουμένως, προκύπτει

$$2 \left[\sum_{n=0}^{L-1} s(n) - \sum_{i=-I}^I b_i \sum_{k=0}^n \bar{r}(n-k-D-i)h_w(k) \right] \left[- \sum_{k=0}^n \bar{r}(n-k-D-j)h_w(k) \right] = 0 \quad (2.26)$$

Αν τεθεί

$$Z_i(n) = \sum_{k=0}^n \bar{r}(n-k-D-i)h_w(k)$$

τότε

$$\sum_{n=0}^{L-1} s(n)Z_j(n) - \sum_{i=-I}^I b_i \sum_{n=0}^{L-1} Z_i(n)Z_j(n) = 0 \quad -I \leq j \leq I \quad (2.27)$$

Υποθέτοντας τη χρήση ενός φίλτρου LTP με 3 taps (όπου δηλ. $I = 1$), η 2.27 μπορεί να επιλυθεί ως προς τους β_i αφού γραφεί σε μορφή πινάκων, ως εξής

$$\begin{bmatrix} b_{-1} \\ b_0 \\ b_1 \end{bmatrix} = \begin{bmatrix} f(-1,-1) & f(0,-1) & f(1,-1) \\ f(-1,0) & f(0,0) & f(1,0) \\ f(-1,1) & f(0,1) & f(1,1) \end{bmatrix}^{-1} \begin{bmatrix} B(-1) \\ B(0) \\ B(1) \end{bmatrix} \quad (2.28)$$

όπου

$$f(i, j) = \sum_{n=0}^{L-1} Z_i(n)Z_j(n)$$

$$B(i) = \sum_{n=0}^{L-1} s(n)Z_i(n)$$

Για φίλτρο 1-tap, με $I = 0$, η λύση δίνεται από

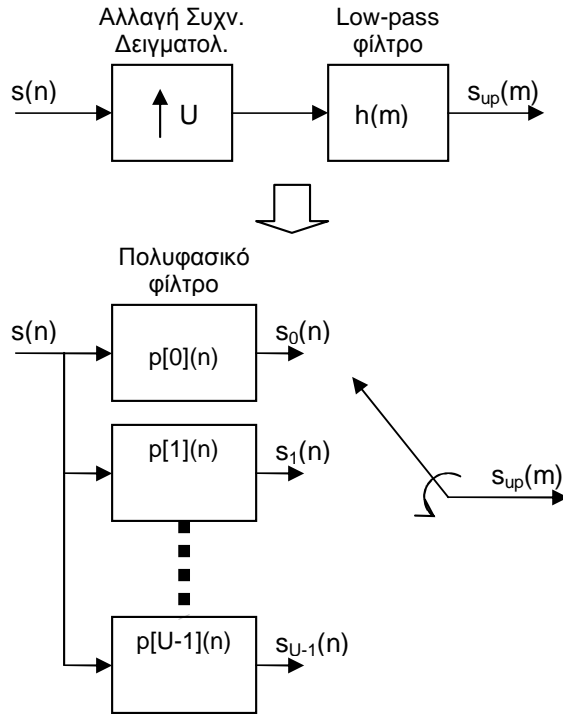
$$b_0 = \frac{B(0)}{f(0,0)} \quad (2.29)$$

Όπως υπονοεί και το Σχήμα 2.5, μετά την εύρεση των συντελεστών β_i , αυτοί εισάγονται ξανά στην 2.25 και η διαδικασία συνεχίζει για τον υπολογισμό της βέλτιστης υστέρησης D για την οποία ελαχιστοποιείται το σφάλμα.

Ένα σημαντικό πρόβλημα που προκύπτει κατά τη διαδικασία αυτή, είναι όταν η υστέρηση είναι μικρότερη το μήκος του πλαισίου ανάλυσης, δηλαδή $D < L$. Τότε η συνάρτηση του μέσου τετραγωνικού σφάλματος γίνεται μη-γραμμική και οι μερικές παράγωγοι $\partial E_w / \partial b_i$ παίρνουν τη μορφή πολυωνύμων τρίτου βαθμού ως προς τους συντελεστές β_i . Μία λύση για το πρόβλημα αυτό είναι η χρήση *προσαρμοστικού κωδικοβιβλίου* (adaptive codebook) με περιεχόμενα τις παρελθούσες αποκρίσεις του LTP φίλτρου. Τότε, το ελλιπές τμήμα του εξεταζόμενου διανύσματος υστέρησης (του διανύσματος D) μπορεί να συμπληρωθεί κατάλληλα με τιμές που αναζητούνται μέσα στο κωδικοβιβλίο, ώστε να προσεγγιστεί το απαιτούμενο μήκος. Σε κάθε επανάληψη, το κωδικοβιβλίο ενημερώνεται για την βέλτιστη λύση που έχει μέχρι τώρα προσεγγιστεί.

Ένα ακόμα πρόβλημα της LTP πρόγνωσης όπως αναλύθηκε παραπάνω είναι αυτό της ακρίβειας, που επηρεάζεται από τη συσχέτιση διακριτών σημάτων. Η ιδανική περίπτωση θα ήταν η συσχέτιση να αναζητείται σε σήματα συνεχούς χρόνου ώστε να είναι όσο το δυνατόν ακριβέστερη. Εφ' όσον όμως η υστέρηση D περιορίζεται αποκλειστικά σε ακέραιες τιμές δειγμάτων του πραγματικού σήματος, είναι αναπόφευκτο κατά την ανάλυση να στρογγυλοποιούνται προς ακέραιοι οι μη-ακέραιες τιμές που θα ήταν οι βέλτιστες για την περιγραφή της περιοδικότητας του σήματος. Συνεπακόλουθα θα μειώνεται η ακρίβεια και η ποιότητα του εξαγόμενου συνθετικού σήματος. Η λύση εδώ, για την αύξηση της αναλυτικής ικανότητας είναι μια *διαδικασία υπερδειγματοληψίας* (up-sampling procedure).

Η διαδικασία υπερδειγματοληψίας περιγράφεται από το Σχήμα 2.6. Για την αντικατάσταση της υστέρησης D (που είναι μια ακέραια τιμή δειγμάτων) από μια πραγματική τιμή D_r , είναι απαραίτητη η μετατροπή του σήματος διακριτού χρόνου



Σχήμα 2.6 Πολυφασική δομή της διαδικασίας up-sampling

$s(n)$ στο αντίστοιχο συνεχούς χρόνου $s_c(t)$. Αυτό μπορεί να γίνει με τη χρήση ενός ιδανικού μη-αιτιοκρατικού χαμηλοπερατού φίλτρου ανασύστασης της μορφής

$$s_c(t) = \sum_{k=-\infty}^{\infty} \frac{s(k) \sin[p(t-k)]}{p(t-k)} \quad (2.30)$$

εφ' όσον για την αρχική δειγματοληψία του σήματος πληρείται η προϋπόθεση του κριτηρίου Nyquist.

Καθώς όμως το ενδιαφέρον εστιάζεται μόνο στα υποπολλαπλάσια της υστέρησης D , δεν απαιτείται στην πραγματικότητα η ανασύσταση του $s_c(t)$ αλλά ένα νέο σήμα διακριτού χρόνου $s_{up}(m)$ που προκύπτει από δειγματοληψία με υψηλότερο ρυθμό του $s_c(t)$. Το ιδανικό φίλτρο της 2.30 αντικαθίσταται από ένα φίλτρο πεπερασμένου μήκους. Στο Σχήμα 2.6 φαίνεται η πολυφασική δομή του φίλτρου ανασύστασης που χρησιμοποιείται, όπου U είναι ο παράγοντας υπερδειγματοληψίας. Με χρήση της τεχνικής αυτής, μια μη-ακέραια τιμή υστέρησης $(D + d/U)$ στην αρχική συχνότητα δειγματοληψίας f_s αντιστοιχεί πλέον σε μια ακέραια τιμή $(UD + d)$, όπου $d = 0, 1, \dots, U-1$ στη νέα συχνότητα δειγματοληψίας Uf_s . Άρα, για την επίτευξη μη ακεραίων τιμών υστέρησης, αρκεί να επιλεγθεί ο κατάλληλος κλάδος του πολυφασικού φίλτρου.

Μια απλή και αποτελεσματική υλοποίηση του φίλτρου μπορεί να γίνει χρησιμοποιώντας μια συνάρτηση sinc (δηλ. $\sin(x) / x$) σε συνδυασμό με παράθυρο Hamming για την αποφυγή του φαινομένου aliasing.

2.3 Σήμα Διέγερσης

Η σημασία του σήματος διέγερσης είναι μεγάλη για την κωδικοποίηση βάσει της τεχνικής της ανάλυσης μέσω σύνθεσης. Εκτός από το να προσφέρει την αρχική πληροφορία για τη λειτουργία του LTP προγνώστη, η διέγερση επίσης περιγράφει χαρακτηριστικά του σήματος ομιλίας τα οποία δεν αντιπροσωπεύονται στο μοντέλο των χρονικά μεταβαλλόμενων φίλτρων, όπως περιοδικές δομές και συσχετίσεις που εκτείνονται σε μεγάλο χρονικό εύρος και δεν μπορούν να ανιχνευθούν από την LTP, ή τυχαίες μεταβολές του σήματος που δεν είναι δυνατό να προβλεφθούν και να οριστούν ντετερμινιστικά. Αναφέρονται διάφορα είδη διεγέρσεων, τα οποία ποικίλουν από ακολουθίες κρουστικών παλμών μέχρι τυχαίο θόρυβο. Μερικά από αυτά είναι:

- *Πολλαπλών παλμών* (Multi-pulse excitation, MPE): Το σήμα διέγερσης αποτελείται από ακολουθίες σταθερού αριθμού κρουστικών παλμών με διαφορετικά πλάτη και σε μη-ομοιόμορφες αποστάσεις. Κατά την κωδικοποίηση, αυτό που ουσιαστικά πρέπει να καθοριστεί είναι η θέση και το πλάτος κάθε παλμού για την ελαχιστοποίηση του σφάλματος.
- *Παλμών σε κανονικά διαστήματα* (Regular-pulse excitation, RPE): Είναι το Multi-pulse με μειωμένη ποιότητα κωδικοποίησης σαν αντιστάθμισμα για το μικρότερο ρυθμό μετάδοσης που παράγει, καθώς στο MP είναι απαραίτητος μεγάλος αριθμός bits για την περιγραφή των θέσεων των παλμών. Εδώ οι παλμοί είναι σε καθορισμένες θέσεις σταθερής απόστασης και μεταβάλλεται μόνο το πλάτος τους.
- *Αυτοδιέγερσης* (Self-excitation, SE): Το σήμα διέγερσης εξάγεται από προηγούμενες τιμές του, κατά παρόμοιο τρόπο με την LTP κλειστού βρόχου. Ουσιαστικά πρόκειται για ένα είδος προσαρμοστικού κωδικοβιβλίου (adaptive codebook) στο οποίο οι αρχικές τιμές κατά την εκκίνηση της διαδικασίας είναι τυχαίες, ενώ κάθε νέα ακολουθία διέγερσης που επιλέγεται ως βέλτιστη μέσα από το κωδικοβιβλίο, επανεισάγεται σε αυτό, εκτοπίζοντας την παλαιότερη (πρώτη στη σειρά) ακολουθία.

- *Κωδικοβιβλίου* (Codebook excitation, CE): Η ακολουθία διέγερσης επιλέγεται από ένα σύνολο προκαθορισμένων ακολουθιών σε συνδυασμό με τη χρήση ενός κατάλληλου παράγοντα κέρδους. Εφ' όσον το κωδικοβιβλίο είναι πεπερασμένου μήκους, οι ακολουθίες που περιέχει θα πρέπει να έχουν εκλεχθεί με προσοχή ώστε να είναι όσο το δυνατόν αντιπροσωπευτικότερες.

Παρακάτω θα αναλυθεί εκτενέστερα η τελευταία μέθοδος, η οποία εφαρμόζεται στην υλοποίηση που παρουσιάζεται στην παρούσα εργασία.

2.3.1 Διέγερση με χρήση κωδικοβιβλίου

Όπως φαίνεται από το Σχήμα 2.6, μετά από αντίστροφο φιλτράρισμα του σήματος από τα STP και LTP φίλτρα (όταν δηλαδή αφαιρεθούν από το σήμα οι δομές που αυτά μοντελοποιούν), το υπόλοιπο του σήματος μοιάζει με τυχαίο θόρυβο κανονικής κατανομής. Για το λόγο αυτό, τα περισσότερα είδη κωδικοβιβλίων που προτείνονται, περιέχουν ακολουθίες διέγερσης με *Γκαουσιανά* (Gaussian) χαρακτηριστικά και χαρακτηρίζονται ως *στοχαστικά* κωδικοβιβλία.

Ένα κωδικοβιβλίο περιέχει έναν σχετικά μεγάλο αριθμό *κωδικολέξεων* (codewords), από τις οποίες επιλέγεται κάθε φορά η βέλτιστη. Οι κωδικολέξεις παρίστανται σαν διανύσματα c_i (c_1, c_2, \dots, c_k , όπου τυπικές τιμές του k είναι 256, 512 και 1024) μήκους όσο το μήκος σε δείγματα του υποπλαισίου ανάλυσης, άρα συνολικά το κωδικοβιβλίο έχει τη μορφή ενός $k \times L$ πίνακα, αν L το μήκος του υποπλαισίου ανάλυσης. Οι αρχικές μορφές κωδικοβιβλίων που προτάθηκαν παρουσίαζαν το πρόβλημα της υπερβολικής απαίτησης σε αποθηκευτική μνήμη συστήματος καθώς περιείχαν k διαφορετικά διανύσματα, άρα απαιτούσαν $k \times L$ θέσεις αποθήκευσης. Μια λύση που εφαρμόζεται ευρέως είναι αυτή των κωδικοβιβλίων με *υπέρθεση* (overlapping codebooks). Με τον τρόπο αυτό, το κωδικοβιβλίο παρίσταται σαν μονοδιάστατος πλέον πίνακας, όπου τα περισσότερα από τα L στοιχεία δύο γειτονικών διανυσμάτων είναι κοινά. Για τη δημιουργία κάθε επόμενου διανύσματος ένα ή δύο στοιχεία εξέρχονται στο τέλος του προηγούμενου διανύσματος και αντίστοιχα ένα ή δύο νέα εισάγονται στην αρχή του. Τα διανύσματα δηλαδή προκύπτουν με ολίσθηση πάνω στον πίνακα κατά 1 ή 2 στοιχεία.

Η αναζήτηση της βέλτιστης διέγερσης γίνεται συνήθως χρησιμοποιώντας το κριτήριο ελαχιστοποίησης του τετραγωνικού σφάλματος. Το ζητούμενο συγκεκριμένα είναι ο προσδιορισμός του πίνακα διέγερσης \mathbf{X} (που στην περίπτωση

του κωδικοβιβλίου με υπέρθεση είναι μονοδιάστατο διάνυσμα όπως περιγράφηκε παραπάνω αλλά θα μελετηθεί στη γενικότερη περίπτωση σαν πίνακας) και του αντίστοιχου κέρδους \mathbf{g} , έτσι ώστε ο \mathbf{gX} να οδηγεί στην σύνθεση σήματος που θα ελαχιστοποιεί το σφάλμα $e(n)$, δηλ.

$$\mathbf{e}_k = \mathbf{s} - \hat{\mathbf{s}}_k \quad (2.31)$$

όπου \mathbf{e}_k το σφάλμα σε μορφή πίνακα, \mathbf{s} το σήμα εισόδου και $\hat{\mathbf{s}}_k$ το παραγόμενο σήμα λόγω του k -οστού πίνακα διέγερσης.

Αν \mathbf{H} ένας $L \times L$ πίνακας ο οποίος περιέχει τη συνδυασμένη κρουστική απόκριση των φίλτρων LTP και STP ως εξής

$$\mathbf{H} = \begin{bmatrix} h(0) & h(1) & \mathbf{L} & h(L-1) \\ 0 & h(0) & \mathbf{L} & h(L-2) \\ \mathbf{M} & \mathbf{M} & \mathbf{O} & \mathbf{M} \\ 0 & 0 & \mathbf{L} & h(0) \end{bmatrix} \quad (2.32)$$

δηλ. η πρώτη γραμμή περιέχει την απόκριση για κρουστική διέγερση $\delta(n)$, η δεύτερη για $\delta(n-1)$, η j -οστή για $\delta(n-j)$ κ.ο.κ., και \mathbf{s}_m η έξοδος μηδενικής εισόδου των LTP και STP φίλτρων, τότε το σήμα στο οποίο θα πρέπει να προσεγγίζει το παραγόμενο (συνθετικό) σήμα θα είναι

$$\mathbf{s}_r = \mathbf{s} - \mathbf{s}_m \quad (2.33)$$

και

$$\mathbf{e}_k = \mathbf{s}_r - \mathbf{g}_k \mathbf{X}_k \mathbf{H} = \mathbf{s}_r - \mathbf{g}_k \hat{\mathbf{s}}_k \quad (2.34)$$

όπου

$$\hat{\mathbf{s}}_k = \mathbf{X}_k \mathbf{H}$$

Το τετραγωνικό σφάλμα προκύπτει ως εξής

$$E_k = \mathbf{e}_k \mathbf{e}_k^T \quad (2.35)$$

όπου το T δηλώνει τον ανάστροφο (transpose) του πίνακα. Το βέλτιστο κέρδος \mathbf{g}_k για την εξεταζόμενη διέγερση μπορεί να υπολογιστεί από τις 2.34, 2.35, απαιτώντας την ορθογωνικότητα του σφάλματος \mathbf{e}_k ως προς το παραγόμενο $\hat{\mathbf{s}}_k$, δηλ.

$$\mathbf{e}_k \hat{\mathbf{s}}_k^T = 0 \quad (2.36)$$

Συνεπώς

$$(\mathbf{s}_r - \mathbf{g}_k \hat{\mathbf{s}}_k) \cdot \hat{\mathbf{s}}_k^T = 0 \quad (2.37)$$

και λύνοντας ως προς το ζητούμενο \mathbf{g}_k ,

$$\mathbf{g}_k = \mathbf{s}_r \hat{\mathbf{s}}_k^T (\hat{\mathbf{s}}_k \hat{\mathbf{s}}_k^T)^{-1} \quad (2.38)$$

Αντικαθιστώντας την παραπάνω στην 2.34, το τετραγωνικό σφάλμα της 2.35 μπορεί να ξαναγραφεί με τη μορφή

$$E_k = \mathbf{s}_r [\mathbf{I} - \hat{\mathbf{s}}_k^T (\hat{\mathbf{s}}_k \hat{\mathbf{s}}_k^T)^{-1} \hat{\mathbf{s}}_k] \mathbf{s}_r^T \quad (2.39)$$

όπου \mathbf{I} ο μοναδιαίος πίνακας.

Για την περίπτωση τώρα που οι πίνακες διέγερσης είναι μονοδιάστατοι (διανύσματα) όπως στην περίπτωση του overlapping κωδικοβιβλίου, η 2.39 απλοποιείται αρκετά, καθώς

$$\hat{\mathbf{s}}_k \hat{\mathbf{s}}_k^T = \sigma \quad (\text{βαθμωτό μέγεθος}) \quad (2.40)$$

$$\mathbf{g}_k = \mathbf{s}_r \hat{\mathbf{s}}_k^T / \sigma = g_k \quad (\text{επίσης βαθμωτό}) \quad (2.41)$$

Τότε

$$\begin{aligned} E_k &= \mathbf{s}_r \mathbf{s}_r^T - g_k \hat{\mathbf{s}}_k \mathbf{s}_r^T \\ &= \mathbf{s}_r \mathbf{s}_r^T - \mathbf{Q}_k \end{aligned} \quad (2.42)$$

Ξαναγράφοντας τις 2.41 και 2.42 στο πεδίο του διακριτού χρόνου

$$g_k = \frac{\sum_{i=0}^{L-1} s_r(i) s_k(i)}{\sum_{i=0}^{L-1} s_k^2(i)} \quad (2.43)$$

$$E_k = \sum_{i=0}^{L-1} s_r^2(i) - g_k \sum_{i=0}^{L-1} s_k(i) s_r(i) \quad (2.44)$$

και με αντικατάσταση του g_k της 2.43 στην 2.44

$$E_k = \sum_{i=0}^{L-1} s_r^2(i) - \frac{\left[\sum_{i=0}^{L-1} s_k(i) s_r(i) \right]^2}{\sum_{i=0}^{L-1} s_k^2(i)} \quad (2.45)$$

ενώ ο παράγοντας \mathbf{Q}_k που αναφέρθηκε παραπάνω δίνεται από

$$\mathbf{Q}_k = \frac{\left[\sum_{i=0}^{L-1} s_k(i) s_r(i) \right]^2}{\sum_{i=0}^{L-1} s_k^2(i)} \quad (2.46)$$

Από τα παραπάνω προκύπτει ότι το κέρδος g_k εκφράζει τη συσχέτιση του σήματος εισόδου με την απόκριση των φίλτρων LTP και STP στο σήμα διέγερσης προς το άθροισμα τετραγώνων της απόκρισης. Επίσης, το τετραγωνικό σφάλμα E_k προκύπτει ως η διαφορά μεταξύ της ενέργειας του σήματος εισόδου και του τετραγώνου της συσχέτισης του σήματος εισόδου με την απόκριση στη διέγερση προς το άθροισμα τετραγώνων της απόκρισης. Πρακτικά, για την εύρεση της βέλτιστης διέγερσης μεγιστοποιείται ο παράγοντας \mathbf{Q}_k .

2.4 Διαδικασία ελαχιστοποίησης σφάλματος

Για την ελαχιστοποίηση του σφάλματος μεταξύ του πραγματικού σήματος $s(n)$ και του παραγόμενου από τη διέγερση μέσω των φίλτρων πρόγνωσης σήματος $\hat{s}(n)$ χρησιμοποιείται όπως αναλύθηκε παραπάνω το κριτήριο του μέσου τετραγωνικού σφάλματος. Όμως, στους χαμηλούς ρυθμούς κωδικοποίησης που επιτυγχάνονται από τον αλγόριθμο CELP, ένα τυπικό κριτήριο αυτού του είδους δεν είναι απολύτως επαρκές, καθώς ο αριθμός των bits που αντιστοιχούν σε κάθε δείγμα φωνής είναι λίγα. Έτσι, το κριτήριο τροποποιείται με βάση την ανθρώπινη ακουστική αντίληψη, με τη μορφή ενός φίλτρου στάθμισης $W(z)$

$$\begin{aligned} W(z) &= \frac{A(z)}{A(z/g)} \\ &= \frac{1 - \sum_{i=1}^p a_i z^{-i}}{1 - \sum_{i=1}^p a_i g^i z^{-i}}, \quad 0 \leq g \leq 1 \end{aligned} \quad (2.47)$$

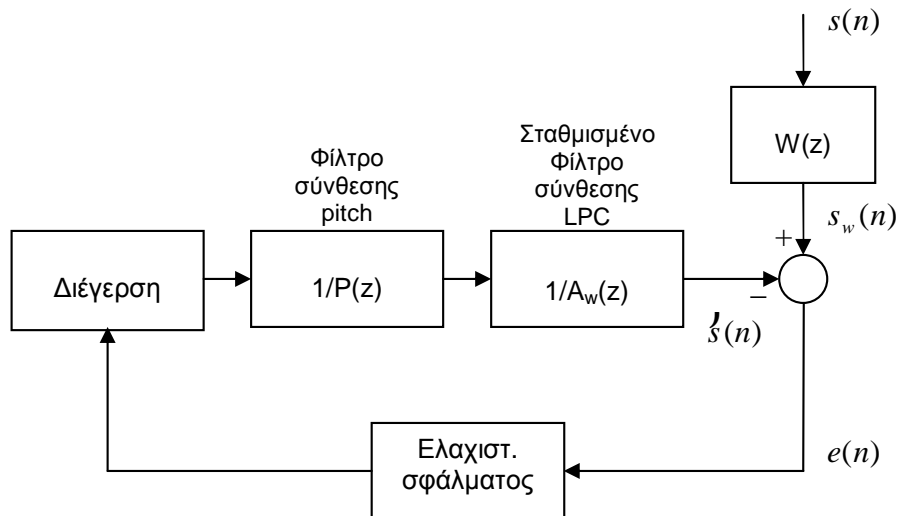
όπου g είναι ο παράγοντας στάθμισης με τιμές που συνήθως κυμαίνονται σε περίπου 0.8 – 0.9, καταλήγοντας στο κριτήριο ελαχιστοποίησης σφάλματος βάσει της αντιληπτικότητας (Perceptually based minimisation criterion). Η επίδραση του φίλτρου είναι να μειώνει την ισχύ του σήματος στις περιοχές των formants και να την ενισχύει έξω από αυτές (στις λεγόμενες φασματικές κοιλάδες – spectral valleys), με αποτέλεσμα την ακουστική κάλυψη του θορύβου στις περιοχές που είναι περισσότερο αντιληπτός.

Μαθηματικά, το φίλτρο δε μεταβάλλει τις συχνότητες των formants, αλλά οδηγεί σε μια διαστολή του εύρους τους (bandwidth expansion) κατά Δf ,

$$\Delta f = -\frac{f_s}{p} \ln g \quad (2.48)$$

όπου f_s η συχνότητα δειγματοληψίας του σήματος.

Η εφαρμογή του φίλτρου στάθμισης μπορεί να γίνει στην κανονική του θέση, δηλαδή στο σήμα σφάλματος που προκύπτει από την αφαίρεση του $\hat{s}(n)$ από το $s(n)$. Προτιμάται όμως μια άλλη προσέγγιση, που οδηγεί σε μείωση του υπολογιστικού όγκου, και προβλέπει την μετακίνηση της διαδικασίας στάθμισης στους δύο κλάδους που συνεισφέρουν στην αφαίρεση, όπως φαίνεται στο Σχήμα 2.7 (που αποτελεί μια τροποποίηση του Σχήματος 2.1). Έτσι, τα εισερχόμενα δείγματα του σήματος φωνής



Σχήμα 2.7 Κωδικοποίηση AbS με τροποποιημένο φίλτρο στάθμισης σήματος

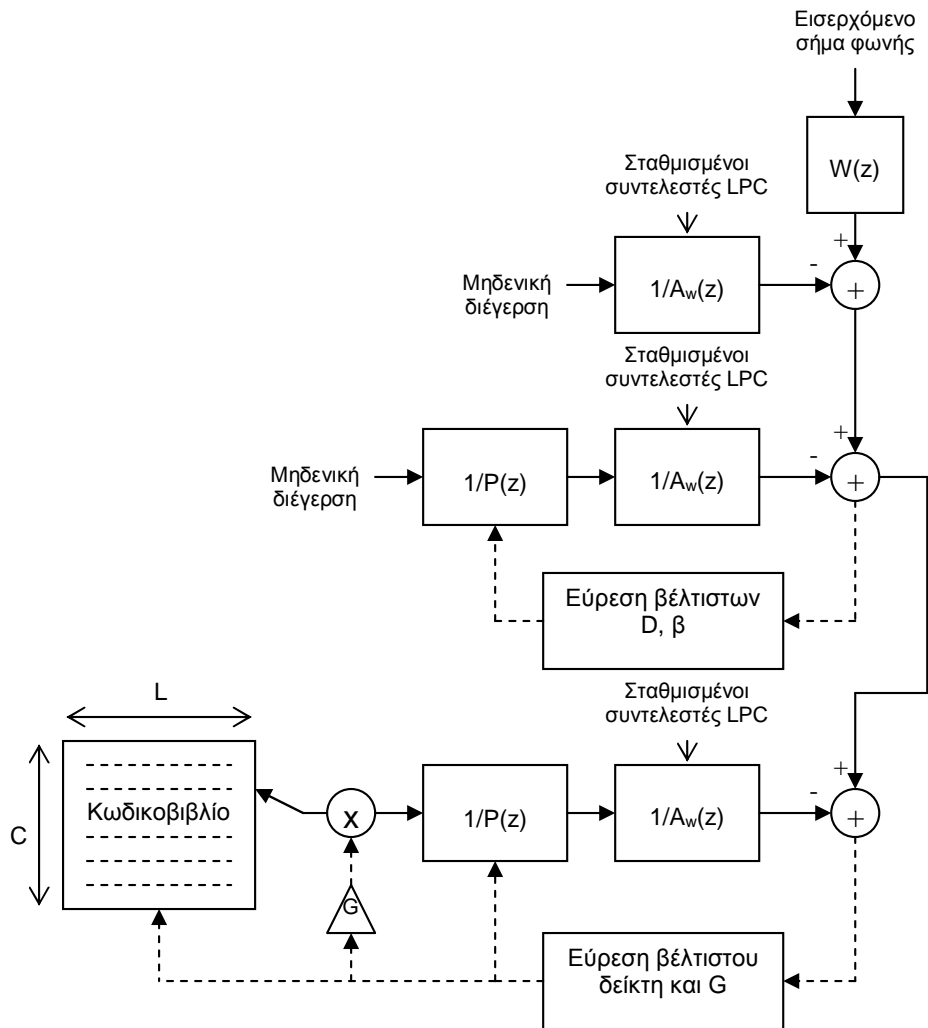
σταθμίζονται μία φορά ανά πλαίσιο, ενώ ταυτόχρονα στον άλλο κλάδο το $W(z)$ σε συνδυασμό με το LPC φίλτρο δίνουν το τροποποιημένο φίλτρο μόνο πόλων της 2.49.

$$\frac{1}{A_w(z)} = \frac{1}{A(z)} W(z) = \frac{1}{1 - \sum_{i=1}^p a_i g^i z^{-i}} \quad (2.49)$$

2.5 Συνολική επισκόπηση της ανάλυσης μέσω σύνθεσης

Ενοποιώντας όλα όσα αναφέρθηκαν στις παραπάνω παραγράφους μπορεί πλέον να δοθεί μια αναλυτικότερη περιγραφή της διαδικασίας ανάλυσης μέσω σύνθεσης, καθώς και το αναλυτικότερο διάγραμμα του Σχήματος 2.8, ως προς τον αλγόριθμο CELP όπου η διέγερση προέρχεται από κωδικοβιβλίο. Τα βήματα της διαδικασίας είναι τα ακόλουθα:

- (a) Το εισερχόμενο σήμα φωνής διαιρείται σε πλαίσια κατάλληλου μήκους και εκτελείται ανάλυση βραχέως χρόνου (STP) σε κάθε πλαίσιο για την εξαγωγή σειράς συντελεστών (συντελεστές LPC) που μοντελοποιούν τη φασματική περιβάλλουσα του σήματος.
- (b) Οι αρχικές συνθήκες του φίλτρου STP πρόγνωσης διαγράφονται ώστε στο επόμενο στάδιο ανάλυσης η STP να εκκινηθεί χωρίς μνήμη.
- (c) Το σήμα διαιρείται σε επιμέρους τμήματα (υποπλαίσια). Σε κάθε ένα από αυτά εκτελείται η μακροπρόθεσμη ανάλυση (LTP), με αλγόριθμο είτε ανοιχτού, είτε κλειστού βρόχου, με τη χρήση ή όχι προσαρμοστικού



Σχήμα 2.8 Αναλυτικότερο διάγραμμα AbS κωδικοποιητή με διέγερση κωδικοβιβλίου (CELP).

κωδικοβιβλίου, ώστε να επιλεγθούν οι βέλτιστοι συντελεστές D (υστέρηση) και β (παράγοντας κέρδους) που μοντελοποιούν την περιοδική δομή του σήματος.

- (d) Αφού καθοριστούν οι συντελεστές των φίλτρων STP και LTP, αναζητείται η βέλτιστη ακολουθία διέγερσης μεταξύ των διανυσμάτων που περιέχονται σε ένα κωδικοβιβλίο το οποίο μπορεί να είναι M -διάστατο ή μονοδιάστατο. Από την ανάλυση προκύπτει ένας δείκτης προς το βέλτιστο διάνυσμα καθώς και ο αντίστοιχος παράγοντας κέρδους G που ελαχιστοποιεί το σφάλμα. Η ανάλυση γίνεται, όπως παραπάνω, ανά υποπλαίσια.

3 Ο αλγόριθμος CELP

3.1 Εισαγωγή

Ο αλγόριθμος CELP είναι ίσως ο περισσότερο υποσχόμενος και ο ευρύτερα χρησιμοποιούμενος για κωδικοποίηση φωνής σε ρυθμούς κάτω από τα 8 Kbps. Για το λόγο αυτό έχουν αναφερθεί πολυάριθμες παραλλαγές και υλοποιήσεις από το 1982 που ο αλγόριθμος προτάθηκε για πρώτη φορά από τον Bishnu Atal. Η υλοποίηση που παρουσιάζεται παρακάτω στηρίζεται κατά κύριο λόγο στην πρόταση των Campbell, Tremain, Welch, που είναι γνωστή ως *PFS-1016* (Proposed Federal Standard), η οποία έγινε για λογαριασμό του Υπουργείου Άμυνας των Η.Π.Α. το 1990 και καθιερώθηκε σαν *DoD 4.8Kbps CELP*. Άλλες ιδιαίτερα γνωστές παραλλαγές του αλγορίθμου είναι αυτές που εφαρμόζονται στους κωδικοποιητές *GSM EFR* (Enhanced Full Rate) της ETSI, *Low-Delay CELP*, που θεσμοθετήθηκε σαν πρόταση G728 της CCITT και *ACELP* (Algebraic CELP), που χρησιμοποιείται στην υλοποίηση του συστήματος TETRA.

Στο παρόν κεφάλαιο, θα αναλυθεί η ακριβής δομή και λειτουργία του αλγορίθμου που υλοποιήθηκε, καθώς και η διάρθρωση της υλοποίησης κάθε επιμέρους τμήματος σε κώδικα.

3.2 Κωδικοποιητής

Ο αλγόριθμος που προτείνεται στο PFS-1016, αφορά την κωδικοποίηση φωνής σε δεδομένα με ρυθμό 4.8 Kbps. Σε γενικές γραμμές, εφαρμόζεται η τεχνική της ανάλυσης μέσω σύνθεσης, όπως αναφέρθηκε στο προηγούμενο κεφάλαιο, με σκοπό

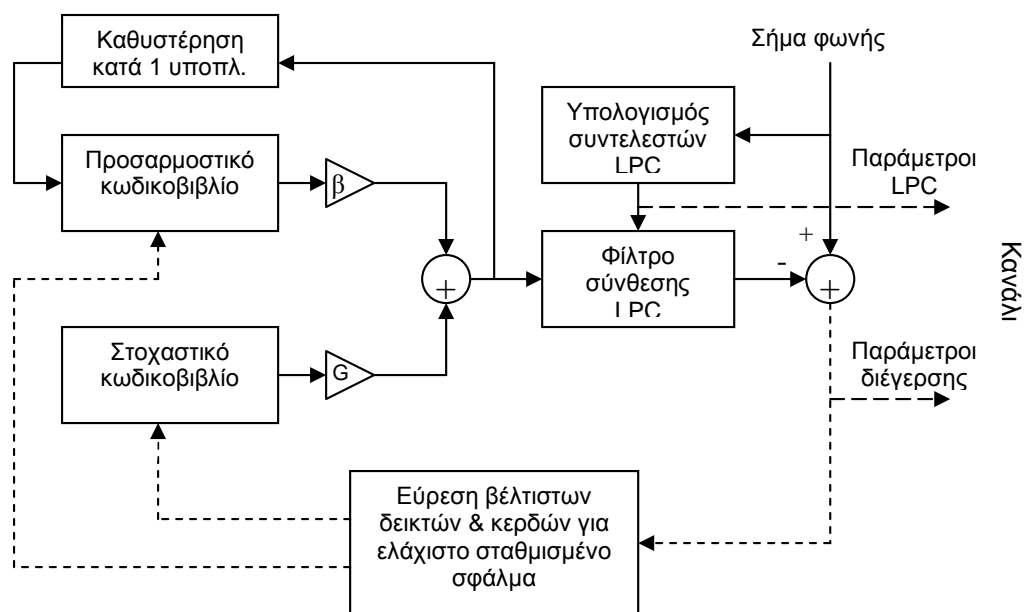
Πίνακας 3.1 Παράμετροι που αποστέλλονται από τον κωδικοποιητή

Παράμετροι LPC ανάλυσης	10 LSPs (Line Spectral Pairs)
Παράμετροι προσαρμοστικού κωδικοβιβλίου	Υστέρηση D , Κέρδος β
Παράμετροι στοχαστικού κωδικοβιβλίου	Δείκτης πίνακα (<i>index</i>), Κέρδος G

να εξαχθούν οι παράμετροι που θα επιτρέψουν την ικανοποιητική ανασύσταση του σήματος φωνής από τον αποκωδικοποιητή. Οι παράμετροι που τελικά αποστέλλονται συνοψίζονται στον πίνακα 3.1.

Το Σχήμα 3.1 παρουσιάζει το λειτουργικό διάγραμμα του κωδικοποιητή CELP. Το εισερχόμενο σήμα φωνής χωρίζεται σε τμήματα 240 δειγμάτων και εκτελείται πρόγνωση βραχέως χρόνου για την εύρεση των συντελεστών LPC. Οι συντελεστές αυτοί σταθμίζονται βάσει της ανθρώπινης ακουστικής αντίληψης (Perceptually-based criterion). Η ανάλυση LTP γίνεται σε υποπλαίσια των 60 δειγμάτων, με μηδενική διέγερση, χρησιμοποιώντας προσαρμοστικό κωδικοβιβλίο, ενώ η αναζήτηση αρχικής διέγερσης γίνεται στα ίδια υποπλαίσια μέσα από στοχαστικό κωδικοβιβλίο με υπέρθεση. Για την ελαχιστοποίηση της διαφοράς μεταξύ πρόβλεψης και πραγματικού σήματος εφαρμόζεται το κριτήριο σταθμισμένου μέσου τετραγωνικού σφάλματος.

Ο σκελετός του κωδικοποιητή υλοποιείται από την ρουτίνα **Encoder**. Η δειγματοληψία του σήματος γίνεται στα 8000 Hz, και στη ρουτίνα εισάγονται πλαίσια 240 δειγμάτων, άρα η διάρκεια κάθε πλαισίου ανάλυσης είναι 30 ms. Ο A/D μετατροπέας μετά την κβάντιση αποδίδει τα δείγματα σε μορφή 16-bit ακεραίων (όσο δηλαδή είναι και το μέγιστο δυναμικό εύρος του κβαντιστή), που εισερχόμενα στον κώδικα μετατρέπονται σε 32-bit floating point για διευκόλυνση των



Σχήμα 3.1 Κωδικοποιητής CELP κατά PFS-1016

υπολογισμών και μεγαλύτερη ακρίβεια. Εφαρμόζεται ένα φίλτρο διέλευσης ζώνης 100-3800 Hz (3 dB εξασθένηση στα σημεία αυτά) για τη διασφάλιση ότι το σήμα είναι voice-band. Ακόλουθα, καλείται η ρουτίνα **LpcAnalysis** που εκτελεί την ανάλυση LPC (βραχέως χρόνου).

3.2.1 Γραμμική πρόγνωση βραχέως χρόνου

Η γραμμική πρόγνωση εκτελείται ανά πλαίσιο, χρησιμοποιώντας τη μέθοδο αυτοσυσχέτισης. Η απαίτηση για χρονική στασιμότητα του σήματος ικανοποιείται με τον πολλαπλασιασμό του πλαισίου ανάλυσης με ένα σταθερό παράθυρο Hamming που ορίζεται από τη σχέση

$$w(n) = \begin{cases} 0.54 - 0.46 \cos\left(2p \frac{n}{N-1}\right) & 0 \leq n \leq N-1 \\ 0, & \text{αλλού} \end{cases}$$

Το παράθυρο αυτού του τύπου έχει αποδειχτεί αποτελεσματικό για εφαρμογές κωδικοποίησης φωνής και είναι το ευρύτερα χρησιμοποιούμενο. Λαμβάνεται με μήκος όσο και του πλαισίου ανάλυσης (240 δείγματα / 30ms).

Το επόμενο βήμα είναι η αναζήτηση των συντελεστών πρόγνωσης a_j (predictor coefficients) για το σχηματισμό του φίλτρου σύνθεσης

$$H(z) = \frac{1}{1 - \sum_{j=1}^p a_j z^{-j}} \quad (3.1)$$

όπου $p = 10$, που είναι ο βαθμός της LPC ανάλυσης που χρησιμοποιείται στη εφαρμογή που παρουσιάζεται εδώ. Έχει βρεθεί ότι ανάλυση αυτού του βαθμού είναι επαρκής για την ακριβή αναπαράσταση της φασματικής περιβάλλουσας σημάτων φωνής (παρ' όλο που για παράδειγμα το πρότυπο G728 της CCITT προτείνει $p = 50$, χωρίς όμως χρήση LTP προγνώστη στη συνέχεια).

Για την εύρεση των a_j , υπολογίζεται η αυτοσυσχέτιση βραχέως χρόνου του σήματος με την εύρεση των συντελεστών αυτοσυσχέτισης $C(i)$ (auto-correlation coefficients).

$$C(i) = \sum_{k=i+1}^{240} s_n(k) s_n(k-i), \quad i = 0, 1, \dots, 10 \quad (3.2)$$

όπου $s_n(k)$ το διάνυσμα μήκους 240 δειγμάτων που περιέχει το εισερχόμενο τμήμα του σήματος φωνής, μετά την εφαρμογή της συνάρτησης παραθύρου Hamming. Το προκύπτον σύστημα των κανονικών εξισώσεων επιλύεται με τον αλγόριθμο

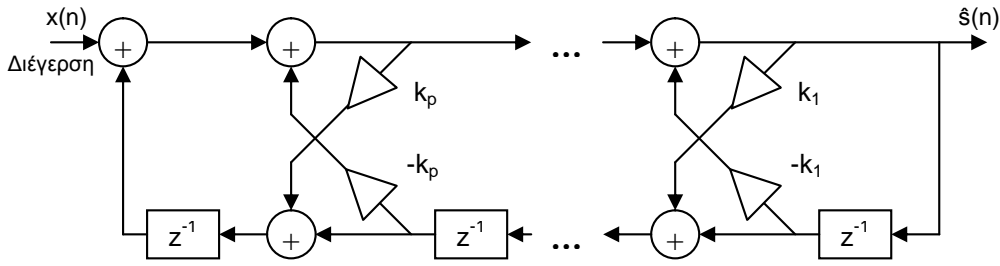
Levinson-Durbin, όπως περιγράφηκε στο προηγούμενο κεφάλαιο, στην υπορουτίνα **DurbinRecur**. Τέλος, οι συντελεστές πρόγνωσης που προκύπτουν από την επίλυση της αναδρομής Levinson-Durbin διοχετεύονται στη συνάρτηση **PercWeight**, ώστε να γίνει η στάθμισή τους βάσει της ανθρώπινης ακουστικής αντίληψης (human hearing perception). Η στάθμιση γίνεται κατά έναν παράγοντα $\gamma = 0.994$, και καταλήγει στην διεύρυνση των formants στους οποίους αντιστοιχούν οι συντελεστές κατά 15 Hz. Η στάθμιση των συντελεστών αντιστοιχεί, κατ' αναλογία με τη μεθοδολογία της παραγράφου 2.4, σε απ' ευθείας στάθμιση του εισερχόμενου σήματος, ενώ κατά τη διάρκεια της LTP πρόγνωσης και της αναζήτησης διέγερσης οι συντελεστές σταθμίζονται και πάλι με διαφορετικό γ , ώστε να προκύψει το νέο φίλτρο $1/A(z\gamma^{-1})$ που δίνει το σταθμισμένο κριτήριο ελαχιστοποίησης σφάλματος.

3.2.2 Μετατροπή των συντελεστών πρόγνωσης σε LSFs

Το κυριότερο πρόβλημα της πρόγνωσης βραχέως χρόνου είναι η κβάντιση των συντελεστών πρόγνωσης a_j ώστε να είναι δυνατή η εκπομπή τους σε μορφή δυαδικών δεδομένων. Η απ' ευθείας κβάντιση δεν είναι αποτελεσματική, καθώς στους χαμηλούς ρυθμούς μετάδοσης που θέτει σαν στόχο ο αλγόριθμος CELP, ο αριθμός των bits που διατίθενται για τη μετάδοση των συντελεστών είναι μικρός. Συνεπώς, υπάρχει πιθανότητα σημαντικών σφαλμάτων κβάντισης, τα οποία θα προκαλούσαν απόκλιση από την επιθυμητή συμπεριφορά του φίλτρου LPC σύνθεσης του αποκωδικοποιητή. Παρουσιάζεται λοιπόν η ανάγκη μετατροπής των συντελεστών πρόγνωσης σε κάποια μορφή η οποία θα εγγυάται μεγαλύτερη σταθερότητα του φίλτρου. Στην παρούσα εφαρμογή χρησιμοποιείται η αναπαράσταση των συντελεστών με τη μορφή των *LSPs* (Line Spectrum Pairs) που προκύπτουν από τις *LSFs* (Line Spectrum Frequencies).

Για την μετατροπή σε LSFs, χρησιμοποιούνται οι μερικοί συντελεστές συσχέτισης (PARCORs) που αναφέρθηκαν στην παράγραφο 2.2.1.1. Όπως ειπώθηκε, οι συντελεστές PARCOR μεταφέρουν την ίδια πληροφορία με τους συντελεστές πρόγνωσης και είναι δυνατή η αναπαράσταση του φίλτρου σύνθεσης μέσω αυτών με υλοποίηση lattice όπως δείχνει το Σχήμα 3.2, όπου k_i είναι οι PARCOR. Το φίλτρο θα είναι ευσταθές για $-1 < k_i < 1$, ενώ οι ακραίες συνθήκες $k_{p+1} = 1$ και $k_{p+1} = -1$ περιγράφουν τις υποθετικές περιπτώσεις ενός τέλεια ανοιχτού και ενός τέλεια

κλειστού φωνητικού σωλήνα (vocal tract), αντίστοιχα, στο επίπεδο της γλωττικής διέγερσης.



Σχήμα 3.2 Υλοποίηση φίλτρου σύνθεσης μέσω των PARCOR

Ο στόχος της μετατροπής σε LSFs είναι η εύρεση δύο πολωνύμων $Q(z)$ και $P(z)$ τέτοιων ώστε οι ρίζες τους e^{jq_n} και e^{jp_n} να βρίσκονται στον μοναδιαίο μιγαδικό κύκλο αν και μόνο αν και οι ρίζες r_i (με $r_i = -k_{p-i}$, $i = 0, \dots, p-1$) της $A(z)$ του φίλτρου LPC σύνθεσης βρίσκονται μέσα στον μοναδιαίο μιγαδικό κύκλο, όπου το φίλτρο LPC σύνθεσης μπορεί να γραφεί κατά τα γνωστά ως εξής

$$H(z) = \frac{1}{A_p(z)} \quad (3.3)$$

με

$$A_p(z) = 1 + \sum_{k=1}^p a_k z^{-k} \quad (3.4)$$

και $p = 10$.

Οι ζητούμενες LSFs θα είναι οι συχνότητες των πόλων και μηδενικών της 3.5 (ή αντίστοιχα, οι γωνιακές θέσεις των ριζών των $Q(z)$ και $P(z)$).

$$Z_T(z) = \frac{Q(z)}{P(z)} \quad (3.5)$$

Για την εύρεσή τους, η $A_p(z)$ μπορεί να αναλυθεί σε δύο συναρτήσεις μεταφοράς, μία με άρτια συμμετρία και μία με περιττή, αν ληφθεί το άθροισμα και η διαφορά αντίστοιχα της $A_p(z)$ και των συζυγών της, $B_p(z) = z^{-(p+1)} A_p(z^{-1})$ συμπεριλαμβάνοντας τις ακραίες συνθήκες k_{p+1} , όπως παρακάτω:

$$k_{p+1} = 1, \quad P_{p+1}(z) = A_p(z) - B_p(z) \quad (3.6)$$

$$k_{p+1} = -1, \quad P_{p+1}(z) = A_p(z) + B_p(z) \quad (3.7)$$

Αντικαθιστώντας τα $B_p(z)$ στις παραπάνω σχέσεις,

$$P_{p+1}(z) = A_p(z) - z^{-(p+1)} A_p(z^{-1})$$

$$= 1 + (a_1 - a_p)z^{-1} + \mathbf{K} + (a_p - a_1)z^{-p} - z^{-(p+1)}$$

που καταλήγει στο γινόμενο

$$P_{p+1}(z) = -z^{-(p+1)} \prod_{i=0}^{p+1} (z + a_i) \quad (3.8)$$

Ομοίως,

$$Q_{p+1}(z) = -z^{-(p+1)} \prod_{i=0}^{p+1} (z + b_i) \quad (3.9)$$

Γνωρίζοντας ότι οι ρίζες $k_{p+1} = \pm 1$ των πολυωνύμων $P_{p+1}(z)$ και $Q_{p+1}(z)$ υπάρχουν, οι τελευταίες σχέσεις μπορούν να γραφούν

$$P(z) = \frac{P_{p+1}(z)}{(1-z)} \quad (3.10)$$

και

$$Q(z) = \frac{Q_{p+1}(z)}{(1+z)} \quad (3.11)$$

Η επίλυση των 3.10 και 3.11, για την εύρεση των ριζών των πολυωνύμων $P(z)$ και $Q(z)$ γίνεται με τη μέθοδο των *σειρών Chebyshev* (Chebyshev Series Method). Όπως ειπώθηκε παραπάνω οι LSF's προκύπτουν ως οι γωνιακές θέσεις των ριζών των $P(z)$ και $Q(z)$ με τις εξής ιδιότητες:

- (a) Οι ρίζες των πολυωνύμων βρίσκονται στον μοναδιαίο μιγαδικό κύκλο.
- (b) Οι ρίζες των πολυωνύμων εναλλάσσονται στον μοναδιαίο μιγαδικό κύκλο, ώστε $0 \leq w_{q,0} < w_{p,0} < w_{q,1} < w_{q,1} \mathbf{K} \leq p$, όπου w_i οι γωνιακές θέσεις των ριζών.

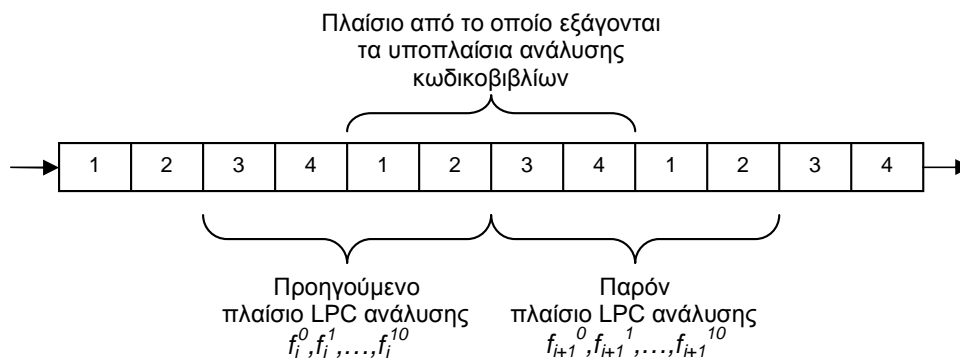
Η μετατροπή των συντελεστών πρόγνωσης σε LSPs (που αποτελούνται από τα ζεύγη των LSFs) καθώς και η κβάντισή τους, γίνεται με τη διοχέτευση των συντελεστών πρόγνωσης στη ρουτίνα **PcCoeffsToLsps**. Η κβάντιση ειδικότερα, γίνεται σε 4 bits για 4 από τις 10 LSFs που, όπως έχει αποδειχθεί, αντιστοιχούν σε περιοχές συχνοτήτων με μεγαλύτερη βαρύτητα για την ανθρώπινη ακοή και άρα είναι περισσότερο ευαίσθητες σε σφάλματα κβάντισης, ενώ οι υπόλοιπες 6 LSFs κβαντίζονται σε 3 bits. Έτσι τελικά, οι παράμετροι της LPC ανάλυσης καταλαμβάνουν 34 bits της κωδικοποιημένης εξόδου, ή αντίστοιχα ένα εύρος $34 \text{ bits} / 30 \text{ msec} = 1.133 \text{ kbps}$.

3.2.3 Δημιουργία υποπλαίσιων για ανάλυση κωδικοβιβλίων

Το μήκος του πλαισίου της LPC ανάλυσης είναι 240 δείγματα ή 30 msec. Για τα ακόλουθα στάδια (την LTP πρόγνωση και την αναζήτηση βέλτιστης διέγερσης) το πλαίσιο χωρίζεται σε τέσσερα επιμέρους υποπλαίσια ίσου μήκους. Έτσι, κάθε υποπλαίσιο αντιστοιχεί σε 60 δείγματα ή 7.5 msec σήματος φωνής.

Ο χωρισμός σε υποπλαίσια δε γίνεται απ' ευθείας, αλλά με μια διαδικασία που συνδυάζει γραμμική παρεμβολή των παραμέτρων της LPC ανάλυσης μέσα σε κάθε πλαίσιο (intra-frame) με ταυτόχρονη παρεμβολή μεταξύ διαδοχικών πλαισίων (inter-frame). Ο λόγος για τον οποίο γίνεται αυτό, είναι για να αντιμετωπιστούν οι μεταβατικές περιοχές (transition regions, όπου για παράδειγμα ένας εύφωτος ήχος διαδέχεται έναν άφωνο) του εισερχόμενου σήματος, που δεν προβλέπονται από το μοντέλο που έχει υποτεθεί στο προηγούμενο κεφάλαιο. Κατά τις μεταβατικές περιοχές, αν δε γινόταν χρήση της παρεμβολής, οι συντελεστές που θα υπολογίζονταν από την ανάλυση θα αντιπροσώπευαν μόνο μια μέση τιμή της μεταβολής των φασματικών χαρακτηριστικών του σήματος.

Το σχήμα 3.3 απεικονίζει τον τρόπο με τον οποίο γίνεται η γραμμική παρεμβολή για την δημιουργία των τεσσάρων υποπλαίσιων. Η ανάλυση κωδικοβιβλίων δεν γίνεται στο ίδιο πλαίσιο με την LPC ανάλυση, αλλά σε ένα που προκύπτει από υστέρηση $\frac{1}{2}$ μήκους πλαισίου. Επίσης, οι LPC παράμετροι που είναι απαραίτητες για τη δημιουργία του φίλτρου σύνθεσης κατά την αναζήτηση μέσα από τα κωδικοβιβλία δεν εξάγονται απ' ευθείας από τις LSFs που αντιστοιχούν στο κάθε υποπλαίσιο, αλλά από μια σταθμισμένη εκδοχή τους. Αν δηλαδή οι $f_i^0, f_i^1, \mathbf{K}, f_i^{10}$ είναι οι μη κβαντισμένες LSF που αντιστοιχούν στο i -οστό πλαίσιο LPC ανάλυσης (προηγούμενο), και $f_{i+1}^0, f_{i+1}^1, \mathbf{K}, f_{i+1}^{10}$ οι LSF που αντιστοιχούν στο $(i+1)$ -οστό



Σχήμα 3.3 Γραμμική παρεμβολή για τη δημιουργία των υποπλαίσιων

πλαίσιο (σε αυτό που μόλις έχει διεξαχθεί η LPC ανάλυση), τότε οι LSF για το κάθε υποπλαίσιο προκύπτουν από τη σταθμισμένη συνεισφορά των δύο πλαισίων:

$$\text{LSFs του υποπλαισίου 1: } f_i^{j,in} = (7/8)f_i^j + (1/8)f_{i+1}^j, \quad j = 0,1,\mathbf{K},10$$

$$\text{LSFs του υποπλαισίου 2: } f_i^{j,in} = (5/8)f_i^j + (3/8)f_{i+1}^j, \quad j = 0,1,\mathbf{K},10$$

$$\text{LSFs του υποπλαισίου 3: } f_i^{j,in} = (3/8)f_i^j + (5/8)f_{i+1}^j, \quad j = 0,1,\mathbf{K},10$$

$$\text{LSFs του υποπλαισίου 4: } f_i^{j,in} = (1/8)f_i^j + (7/8)f_{i+1}^j, \quad j = 0,1,\mathbf{K},10$$

Η καθυστέρηση της ανάλυσης κατά $1/2$ μήκος πλαισίου γίνεται απλώς με τη χρήση ενός διανύσματος που δημιουργείται σε κάθε επανάληψη και περιέχει το δεύτερο μισό του τμήματος του σήματος φωνής που επεξεργάστηκε στην προηγούμενη επανάληψη και το πρώτο μισό του τμήματος του παρόντος σήματος. Η συνάρτηση που εκτελεί τη γραμμική παρεμβολή μεταξύ των LSFs είναι η **Analysis Interpolation**, ενώ στη συνέχεια οι προκύπτουσες LSFs μετατρέπονται πάλι σε συντελεστές πρόγνωσης, για τη δημιουργία του φίλτρου σύνθεσης. Η διαδικασία έρευνας των κωδικοβιβλίων εκτελείται από τη ρουτίνα **Codebook Search**, η οποία εκτελείται τέσσερις φορές ανά πλαίσιο, δηλ. για κάθε υποπλαίσιο.

3.2.4 Ανάλυση προσαρμοστικού κωδικοβιβλίου

Για την μοντελοποίηση της περιοδικής δομής του σήματος ο προτεινόμενος από το PFS-1016 αλγόριθμος χρησιμοποιεί προσαρμοστικό κωδικοβιβλίο και η διαδικασία αναζήτησης γίνεται μέσω κλειστού βρόχου βάσει του σταθμισμένου κριτηρίου ελαχιστοποίησης του τετραγωνικού σφάλματος. Το κωδικοβιβλίο περιέχει 256 διανύσματα που αντιστοιχούν τόσο σε ακέραιες όσο και σε μη ακέραιες υστερήσεις, 20 έως 147 δειγμάτων. Είναι προσαρμοστικό ως προς το σήμα που προκύπτει από τη βέλτιστη συνδυασμένη έξοδο $r(n)$ και των δύο κωδικοβιβλίων (προσαρμοστικού και στοχαστικού) για την ανάλυση του προηγούμενου υποπλαισίου, όπως παρουσιάστηκε και στο Σχήμα 3.1. Η σχέση μεταξύ της προηγούμενης και της παρούσας τιμής του $r(n)$ προκύπτει από το φίλτρο της LTP πρόγνωσης

$$H(z) = \frac{1}{1 - \beta z^{-D}} \quad (3.12)$$

όπου D η υστέρηση και β ο παράγοντας κέρδους. Ξαναγράφοντας στο πεδίο του διακριτού χρόνου, με $r_{prev}(n)$ την προηγούμενη τιμή του $r(n)$, προκύπτει η 3.13.

$$r(n) = r_{prev}(n) + b r(n - D) \quad (3.13)$$

Ο αλγόριθμος αναζητά την βέλτιστη υστέρηση D ανάμεσα στα 256 υποψήφια διανύσματα του κωδικοβιβλίου, καθώς και το αντίστοιχο κέρδος β έτσι ώστε να οδηγήσει σε ελαχιστοποίηση του μέσου τετραγώνου του σφάλματος στην παρακάτω έκφραση

$$\begin{aligned} \sum_n e^2(n) &= \sum_n [h_w(n) * (s(n) - \hat{s}(n))]^2 \\ &= \sum_n \left[h_w(n) * \left(s(n) - b r(n - D) - \sum_{j=1}^{10} a_j \hat{s}(n - j) \right) \right] \end{aligned} \quad (3.14)$$

όπου το σύμβολο $*$ εκφράζει τη συνέλιξη και το $h_w(n)$ εκφράζει την κρουστική απόκριση του φίλτρου στάθμισης του σφάλματος.

Οι 256 υποψήφιες κωδικολέξεις που απαρτίζουν το κωδικοβιβλίο σχηματίζονται με βάση ένα διάνυσμα μήκους 147 στοιχείων της μορφής $(l_{-147}, l_{-146}, l_{-145}, \dots, l_{-1})$ που ανανεώνεται με την ολίσθηση κατά 60 στοιχεία με τέτοιο τρόπο ώστε να αφαιρεθούν από το διάνυσμα τα 60 παλαιότερα στοιχεία και να προστεθούν 60 νέα του σήματος $r(n)$ της αμέσως προηγούμενης επανάληψης της ανάλυσης, δηλ.

$$l_{-61} = l_{-1}, l_{-62} = l_{-2}, \dots, l_{-147} = l_{-87} \quad \text{και} \quad l_{-1} = r_{59}, l_{-2} = r_{58}, \dots, l_{-60} = r_0$$

Έτσι, οι πρώτες 88 κωδικολέξεις μήκους 60 στοιχείων σχηματίζονται ως εξής

$$\begin{array}{cccc} l_{-147}, & l_{-146}, & \mathbf{K}, & l_{-88} \\ l_{-146}, & l_{-145}, & \mathbf{K}, & l_{-87} \\ \mathbf{M} & \mathbf{M} & \mathbf{O} & \mathbf{M} \\ l_{-60}, & l_{-59}, & \mathbf{K}, & l_{-1} \end{array}$$

και αντιπροσωπεύουν ακέραιες τιμές της υστέρησης D . Οι επόμενες 40 κωδικολέξεις, που επίσης αντιπροσωπεύουν ακέραιες τιμές υστέρησης, σχηματίζονται ακόλουθα.

$$\begin{array}{cccc} l_{-59}, & l_{-58}, & \mathbf{K}, & l_{-3}, l_{-2}, l_{-1}, l_{-59}, \\ l_{-58}, & l_{-57}, & \mathbf{K}, & l_{-2}, l_{-1}, l_{-58}, l_{-57}, \\ \mathbf{M} & \mathbf{M} & \mathbf{O} & \mathbf{M} \quad \mathbf{M} \quad \mathbf{M} \quad \mathbf{M} \\ l_{-20}, & l_{-19}, & \mathbf{K}, l_{-1}, l_{-20}, l_{-19}, \mathbf{K}, l_{-1}, l_{-20}, l_{-19}, \mathbf{K}, & l_{-4}, l_{-3}, l_{-2}, l_{-1}, \end{array}$$

Οι υπόλοιπες 128 κωδικολέξεις αντιπροσωπεύουν μη-ακέραιες τιμές για την υστέρηση D , και σχηματίζονται από την παρεμβολή των δύο πλησιέστερων στη βέλτιστη κωδικολέξεων κατά την πορεία της ανάλυσης.

Καθώς η πλήρης εξέταση του προσαρμοστικού κωδικοβιβλίου θα ήταν εξαιρετικά επίπονη υπολογιστικά, ο αλγόριθμος εφαρμόζει *ιεραρχική* αναζήτηση και

επιπλέον διαφορετική μέθοδο έρευνας του κωδικοβιβλίου για τα άρτια και τα περιττά αριθμημένα υποπλάισια. Για κάθε περιττό υποπλάισιο, η βέλτιστη υστέρηση και το αντίστοιχο κέρδος αναζητούνται αρχικά ανάμεσα στα 128 πρώτα διανύσματα του κωδικοβιβλίου που περιέχουν ακέραιες υστερήσεις. Έστω ότι με D^* δηλώνεται η βέλτιστη υστέρηση που προκύπτει από το βήμα αυτό, με I^* ο δείκτης της D^* στο κωδικοβιβλίο και με m^* ένας συντελεστής (match score) άμεσα ανάλογος με το υπολογισμένο τετραγωνικό σφάλμα (ουσιαστικά το αντίθετο του μερικού μέσου τετραγωνικού σφάλματος). Τότε στο επόμενο στάδιο εξετάζονται τα υποπολλαπλάσια της υστέρησης D^* , δηλαδή οι υστερήσεις $(1/2)D^*$, $(1/3)D^*$, $(1/4)D^*$. Αν ο συντελεστής m^* που θα προκύψει για κάποια από τις παραπάνω απέχει από τον αρχικό κατά το πολύ 1 dB, τότε σαν βέλτιστη υστέρηση επιλέγεται η αντίστοιχη υποπολλαπλάσια. Το τελικό στάδιο της αναζήτησης για τα περιττά αριθμημένα υποπλάισια περιλαμβάνει την εξέταση των μη-ακέραιων υστερήσεων που εξάγονται από παρεμβολή μεταξύ των κωδικολέξεων που βρίσκονται στις θέσεις μεταξύ των $I^* \pm 3$ του κωδικοβιβλίου (γεινιάζουσες της βέλτιστης). Η παρεμβολή γίνεται με τη χρήση πολυφασικού φίλτρου, όπως αναφέρθηκε στην παράγραφο 2.2.2.2, που ισοδυναμεί με ανασύσταση ενός υπερδειγματοληπτημένου σήματος. Ο υπολογισμός μιας κωδικολέξης $r(i)$ με υστέρηση $D+f$, όπου f το κλασματικό μέρος της, γίνεται μέσω μιας αναδρομικής διαδικασίας ανασύστασης:

$$r_{D+f}(i) = \sum_{j=-4}^3 w_f(j) r_{D+f}(i-D+j) \quad (3.15)$$

όπου

$$w_f(j) = h(12(j+f)) \frac{\text{sin}((j+f)p)}{(j+f)p}, \quad j = -4, -3, \mathbf{K}, 3, \quad f = \frac{1}{4}, \frac{1}{3}, \frac{1}{2}, \frac{2}{3}, \frac{3}{4}$$

η συνάρτηση sinc σε συνδυασμό με παράθυρο Hamming, και

$$h(n) = 0.54 + 0.46 \cos\left(\frac{np}{6N}\right)$$

η συνάρτηση Hamming, με $N = 8$.

Όσον αφορά τα άρτια αριθμημένα υποπλάισια, η αναζήτηση δεν γίνεται σε όλο το κωδικοβιβλίο, αλλά μόνο σε περιοχή καθορισμένου πλήθους κωδικολέξεων γύρω από τη βέλτιστη λύση του προηγούμενου (περιττού) υποπλαισίου. Κάτι τέτοιο είναι εφικτό, εφόσον δεν αναμένεται ραγδαία αλλαγή της περιοδικής δομής του σήματος στο διάστημα ενός υποπλαισίου. Συγκεκριμένα, αν i είναι ο δείκτης στη βέλτιστη υστέρηση που βρέθηκε για το περιττό υποπλάισιο, στο άρτιο ερευνάται η

περιοχή $j = \min[\max(i - 31, 1), 193), j+1, j+2, \dots, j+63$ που αντιστοιχεί σε εύρος ± 32 κωδικολέξεων γύρω από την προηγούμενη βέλτιστη. Όπως και προηγούμενα, η εξέταση δίνει τα βέλτιστα D^* , I^* και m^* , τα οποία στο επόμενο στάδιο συγκρίνονται με τις τιμές τους που προκύπτουν από εξέταση των μη-ακέραιων τιμών υστέρησης για το διάστημα $I^* \pm 3$ και επιλέγεται η καλύτερη λύση.

Η ανάλυση προσαρμοστικού κωδικοβιβλίου εκτελείται από τη ρουτίνα **ACbSearch**, αφού έχει αφαιρεθεί η συνεισφορά του φίλτρου σύνθεσης LPC από το εισερχόμενο σήμα φωνής (αρχικό σήμα σφάλματος). Μέσα από αυτήν, καλείται η **FindACbGain** για τον υπολογισμό του συντελεστή κέρδους. Οι δύο αυτές ρουτίνες προκαλούν το δεύτερο μεγαλύτερο υπολογιστικό φορτίο για κάθε πλαίσιο (μετά την εξέταση του στοχαστικού κωδικοβιβλίου), ακόμα και για την υποβέλτιστη, όχι εξαντλητική, μορφή με την οποία εκτελούνται. Τελικά, μετά την εκτέλεσή τους, επιστρέφεται ένας δείκτης στη θέση της βέλτιστη διέγερσης, κωδικοποιημένος στα 8 bits (που αντιστοιχούν σε κωδικοβιβλίο 256 θέσεων) για τις περιττά αριθμημένες κωδικολέξεις και στα 6 bits για τις άρτια αριθμημένες (όπου κωδικοποιείται η σχετική απόσταση από την προηγούμενη λέξη μέσα σε 64 θέσεις). Επίσης επιστρέφεται ένας παράγοντας κέρδους, κβαντισμένος στα 5 bits με μη-ομοιόμορφη, γραμμική κβάντιση. Συνολικά λοιπόν, οι παράμετροι του προσαρμοστικού κωδικοβιβλίου καταλαμβάνουν για κάθε πλαίσιο $[(8+6+8+6) + 4 \times 5] = 48$ bits στην δυαδική ακολουθία εξόδου ή $30 \text{ msec} / 48 \text{ bits} = 1.6 \text{ kbps}$ εύρους.

3.2.5 Ανάλυση στοχαστικού κωδικοβιβλίου

Μετά την αφαίρεση των επιδράσεων των φίλτρων LPC και LTP σύνθεσης, το υπόλοιπο του σήματος δεν παρουσιάζει πλέον περιοδικότητα και η μορφή του μπορεί να προσεγγιστεί με τυχαίο θόρυβο. Η αναπαράσταση του θορύβου αυτού κατά τον αλγόριθμο του PFS-1016 γίνεται με τη χρήση ενός στοχαστικού κωδικοβιβλίου 512 κωδικολέξεων, οι οποίες συντίθενται με επικάλυψη από ένα μονοδιάστατο διάνυσμα μήκους 1082 στοιχείων και ολίσθηση κατά -2.

Η προκύπτουσα μορφή του κωδικοβιβλίου εμφανίζεται στον Πίνακα 3.2. Είναι φανερό ότι οι γειτονικές κωδικολέξεις περιέχουν τα ίδια στοιχεία εκτός των πρώτων και των τελευταίων δύο στοιχείων κάθε μιας. Οι τιμές που αντιστοιχούν στα δείγματα ανήκουν στην τριάδα (-1, 0, +1), ενώ τα μηδενικά καταλαμβάνουν το 77%

Πίνακας 3.2 Τρόπος σχηματισμού στοχαστικού κωδικοβιβλίου

Δείκτης	Λαμβανόμενο δείγμα από αρχικό μονοδιάστατο διάνυσμα
511	0, 1, 2, ..., 58, 59
510	2, 3, 4, ..., 60, 61
M	M
N	2(511-N), 2(511-N)+1, ..., 2(511-N)+59
M	M
1	1020, 1021, 1022, ..., 1078, 1079
0	1022, 1023, 1024, ..., 1080, 1081

του πίνακα (sparse codebook). Η υλοποίηση του κωδικοβιβλίου κατά τον τρόπο αυτό δημιουργεί ορισμένα πλεονεκτήματα, τα οποία περιγράφονται παρακάτω:

- § Απαιτούνται μόνο δύο bits για την αναπαράσταση των -1, 0, +1.
- § Πολλαπλασιασμός με -1 και +1 αντιστοιχεί σε μια απλή αλλαγή προσήμου του αποτελέσματος.
- § Πρόσθεση ενός όρου με μηδέν ισοδυναμεί με το να μείνει αναλλοίωτος ο όρος, ενώ η συχνότητα εμφάνισης μηδέν είναι μεγάλη (77%).
- § Κατά τον υπολογισμό συνέλιξης για δύο γειτονικές κωδικολέξεις, το αποτέλεσμα για τη δεύτερη μπορεί να εξαχθεί από το αποτέλεσμα για την πρώτη με την προσθαφαίρεση μόνο των όρων που αφορούν τα τέσσερα διαφορετικά στοιχεία.

Προφανώς, όλα τα παραπάνω καταλήγουν σε μεγάλη μείωση του υπολογιστικού φόρτου σε σχέση με μια άλλη μορφή κωδικοβιβλίου που θα περιείχε μη-μοναδιαίες τιμές, μικρότερη περιεκτικότητα σε μηδενικά και δεν θα χρησιμοποιούσε υπέρθεση.

Από το κωδικοβιβλίο των 512 λέξεων, ο αλγόριθμος αναζητά την κωδικολέξη εκείνη, \mathbf{x} , που θα ικανοποιεί το κριτήριο ελαχιστοποίησης του τετραγωνικού σφάλματος, σε συνδυασμό με το αντίστοιχο βαθμωτό κέρδος g_x . Έστω i ο δείκτης στη λέξη ($0 \leq i \leq 511$). Τότε ο πολλαπλασιασμός της λέξης \mathbf{x}_i με το αντίστοιχο κέρδος g_i οδηγεί στο σχηματισμό του διανύσματος διέγερσης $\mathbf{u}_i = \mathbf{x}_i \cdot g_i$. Το σήμα σφάλματος θα προκύψει αν από το αρχικό σήμα φωνής αφαιρεθεί το παραγόμενο. Το

παραγόμενο σήμα που οφείλεται στη συγκεκριμένη διέγερση και τη μορφοποίησή της από τα φίλτρα σύνθεσης θα δίνεται από την

$$\hat{\mathbf{s}}_i = \mathbf{H}(\mathbf{a} + \mathbf{u}_i) + \mathbf{s}_0 \quad (3.16)$$

όπου

$$\mathbf{H} = \begin{bmatrix} h(0) & h(1) & \mathbf{L} & h(L-1) \\ 0 & h(0) & \mathbf{L} & h(L-2) \\ \mathbf{M} & \mathbf{M} & \mathbf{O} & \mathbf{M} \\ 0 & 0 & \mathbf{L} & h(0) \end{bmatrix} \quad (3.17)$$

ο πίνακας που περιέχει την κρουστική απόκριση των συνδυασμένων φίλτρων σύνθεσης, \mathbf{a} το βέλτιστο διάνυσμα που επιλέχθηκε από το προσαρμοστικό κωδικοβιβλίο και \mathbf{s}_0 η απόκριση μηδενικής εισόδου των φίλτρων. Το σφάλμα που πρέπει να ελαχιστοποιηθεί θα είναι

$$\mathbf{e}_i = \mathbf{W}(\mathbf{s} - \hat{\mathbf{s}}_i) \quad (3.18)$$

όπου \mathbf{W} η κρουστική απόκριση του φίλτρου στάθμισης σε μορφή πίνακα. Η ελαχιστοποίηση γίνεται με κλειστό βρόχο, εφαρμόζοντας εξαντλητική εξέταση του κωδικοβιβλίου, εξετάζοντας δηλαδή το σύνολο των διανυσμάτων που περιέχει το κωδικοβιβλίο και όχι επιλεκτικά υποσύνολα όπως γίνεται για την ανάλυση προσαρμοστικού κωδικοβιβλίου.

Η ανάλυση στοχαστικού κωδικοβιβλίου γίνεται από τη ρουτίνα **SCbSearch**, ενώ μέσα από αυτήν καλείται η **FindSCbGain** για την εύρεση του κατάλληλου κέρδους για την βέλτιστη κωδικολέξη. Επιστρέφουν έναν δείκτη σε θέση του κωδικοβιβλίου, μήκους 9 bit (τα οποία αντιστοιχούν στο κωδικοβιβλίο των 512 λέξεων), καθώς και τον παράγοντα κέρδους με μη-ομοιόμορφη, γραμμική κβάντιση σε 5 bit. Άρα συνολικά καταλαμβάνουν $[(9+5) \times 4] = 56$ bits στην ψηφιοσειρά εξόδου, ή $56 \text{ bits} / 30 \text{ msec} = 1.867 \text{ kbps}$ εύρους. Οι δύο αυτές ρουτίνες δημιουργούν το μεγαλύτερο υπολογιστικό φόρτο ανά πλαίσιο κατά την εκτέλεση του αλγορίθμου. Να σημειωθεί στο σημείο αυτό ότι λόγω ακριβώς του μεγάλου υπολογιστικού φόρτου, κατά την πορεία της υλοποίησης το μέγεθος του στοχαστικού κωδικοβιβλίου μειώθηκε από τις 512 στις 256 λέξεις, όπως θα εξηγηθεί στο κεφάλαιο 5.

3.2.6 Συγκεντρωτικά χαρακτηριστικά κωδικοποιητή

Ο συνολικός αριθμός των bits που μεταδίδονται ανά πλαίσιο είναι το άθροισμα των bits που παράγονται από τα επιμέρους τμήματα, όπως αναλύθηκε στις παραπάνω

Πίνακας 3.3 Συνολικά χαρακτηριστικά υλοποιημένου κωδικοποιητή

	Γραμμική πρόγνωση	Προσαρμοστικό K/B	Στοχαστικό K/B
Διάρκεια πλαισίου	30 msec	7.5 msec	7.5 msec
Εξαγόμενες παράμετροι	10 LSPs	1 υστέρηση 1 κέρδος	1 δείκτης 1 κέρδος
Τύπος ανάλυσης	ανοικτού βρόχου 10 ^{ου} βαθμού μέθ. αυτοσυσχέτισης παράθυρο Hamming μετατροπή σε LSPs	κλειστού βρόχου λέξεις 60 στοιχείων στάθμιση κατά 0.8 ιεραρχική εξέταση υστέρηση 20-147 δ.	κλειστού βρόχου λέξεις 60 στοιχείων στάθμιση κατά 0.8 εξαντλητική εξέταση ολίσθηση κατά -2
bits ανά πλαίσιο	34 (3,4,4,4,4,3,3,3,3)	υστέρηση: 8+6+8+6 κέρδος: 5x4	δείκτης: 9x4 κέρδος: 5x4
Ρυθμός μετάδοσης	1133 bps	1600 bps	1867 bps

παραγράφους και είναι 138 bits. Ο αριθμός αυτός αντιστοιχεί σε ρυθμό μετάδοσης 4600 bps, δηλαδή 200 bps λιγότερα από αυτά που προβλέπει η πρόταση PFS-1016, καθώς για την παρούσα εργασία δεν υλοποιήθηκαν η κωδικοποίηση προστασίας σφαλμάτων (η πρόταση προβλέπει 4 bits / πλαίσιο για κωδικοποίηση Hamming FEC), η ύπαρξη 1 bit / πλαίσιο για συγχρονισμό και η ύπαρξη 1 bit / πλαίσιο για ανάγκες μελλοντικής επέκτασης.

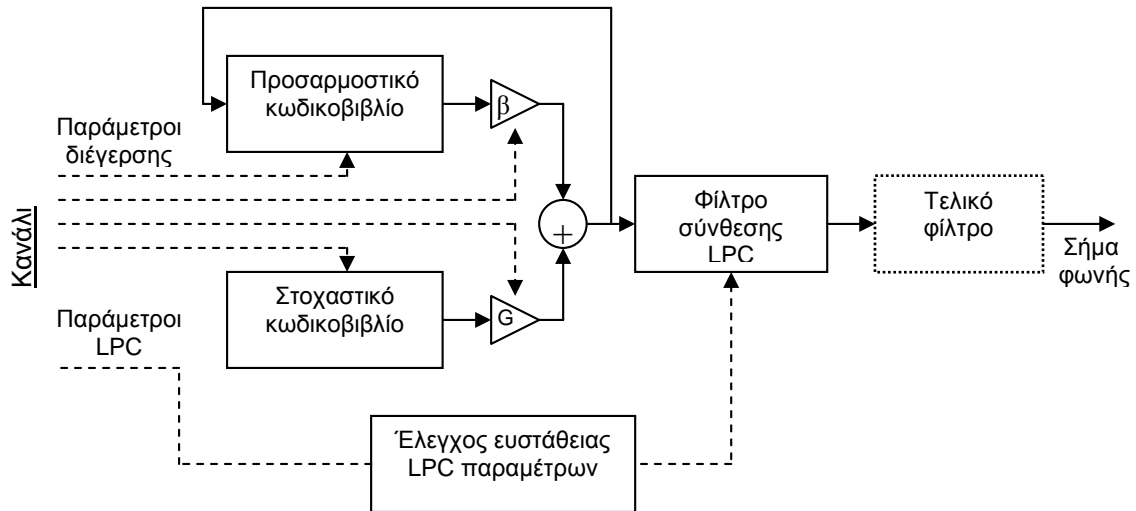
Οι παράμετροι προς μετάδοση εισάγονται στην συνάρτηση **PackIntoStream**, που δέχεται σαν είσοδο τις δεκαδικές τιμές που προκύπτουν από κάθε επιμέρους τμήμα του κώδικα και τις μετατρέπει σε μια μοναδική δυαδική ακολουθία, η οποία είναι έτοιμη για αποστολή στο κανάλι.

Τα συνολικά χαρακτηριστικά του κωδικοποιητή που υλοποιήθηκε εμφανίζονται στον πίνακα 3.3.

3.3 Αποκωδικοποιητής

Ο αποκωδικοποιητής CELP είναι όμοιος κατά μεγάλο τμήμα με τον κωδικοποιητή ή αλλιώς αποτελεί τμήμα του κωδικοποιητή. Η διαδικασία της σύνθεσης της φωνής χρησιμοποιείται τόσο κατά την κωδικοποίηση του σήματος (κατά τον κλειστό βρόχο για την ελαχιστοποίηση του τετραγωνικού σφάλματος) όσο και κατά την αποκωδικοποίηση (για την αναπαραγωγή του σήματος φωνής που μεταδόθηκε).

Το Σχήμα 3.4 παρουσιάζει το λειτουργικό διάγραμμα του αποκωδικοποιητή CELP. Η σύνθεση της φωνής γίνεται μέσω της διοχέτευσης στο LPC φίλτρο



Σχήμα 3.4 Αποκωδικοποιητής CELP κατά PFS-1016

κατάλληλου σήματος διέγερσης, το οποίο προκύπτει από τον παράλληλο συνδυασμό ενός στοχαστικού και ενός προσαρμοστικού κωδικοβιβλίου. Το στοχαστικό κωδικοβιβλίο περιέχει τις ίδιες τιμές με αυτό του κωδικοποιητή, έτσι ώστε να αναπαράγει αποτελεσματικά ένα σήμα που μοιάζει με τυχαίο θόρυβο. Το προσαρμοστικό κωδικοβιβλίο ανανεώνεται κάθε υποπλαίσιο από το συνολικό σήμα διέγερσης. Τα κωδικοβιβλία έχουν τα ίδια χαρακτηριστικά με αυτά που είχαν στον κωδικοποιητή, δηλαδή είναι αποθηκεύονται ως μονοδιάστατοι πίνακες και με τη χρήση υπέρθεσης (ολίσθησης) δημιουργούν λέξεις μήκους 60 στοιχείων.

Η υλοποίησή του αποκωδικοποιητή γίνεται στη ρουτίνα **Decoder**, μέσω της οποίας καλούνται όλες οι επιμέρους συναρτήσεις. Αρχικά, εξάγονται οι παράμετροι του μοντέλου παραγωγής της φωνής από την ακολουθία εισόδου, με τη χρήση της συνάρτησης **UnpackStream** η οποία δέχεται σαν είσοδο τη δυαδική ακολουθία και επιστρέφει τις δεκαδικές τιμές των παραμέτρων. Η σύνθεση γίνεται και εδώ ανά πλαίσιο 240 δειγμάτων.

Για κάθε πλαίσιο εξάγονται από την ακολουθία 10 LSPs, που αντιστοιχούν στους συντελεστές του φίλτρου σύνθεσης [ρουτίνα **DecodeLsps**]. Οι LSPs ελέγχονται για τη μονοτονικότητά τους (ώστε να προκύπτει δηλαδή $0 \leq w_{q,0} < w_{p,0} < w_{q,1} < w_{q,1} \mathbf{K} \leq \mathbf{p}$, όπως αναλύθηκε στην παράγραφο 3.2.2) που θα εγγυηθεί την ευστάθεια του φίλτρου. Αν διαπιστωθεί σφάλμα και δεν είναι δυνατή η αποκατάστασή του, επαναλαμβάνονται οι παράμετροι του προηγούμενου πλαισίου. Από την ψηφιακή ακολουθία εξάγονται επίσης οι παράμετροι των κωδικοβιβλίων,

δηλαδή οι δείκτες και τα αντίστοιχα κέρδη, στην δεκαδική τους μορφή [**DecodeACbIndex**, **DecodeACbGain**, **DecodeSCbIndex**, **DecodeSCbGain**] . Ακολουθεί η σύνθεση, που γίνεται ανά υποπλαίσιο 60 δειγμάτων.

Η σύνθεση εκκινείται με την εύρεση μέσα από το στοχαστικό κωδικοβιβλίο της λέξης στην οποία αντιστοιχεί ο δείκτης που λήφθηκε [**DecodeExcVec**]. Η λέξη πολλαπλασιάζεται με το κέρδος και στη συνέχεια εφαρμόζεται σαν διέγερση σε ένα φίλτρο LTP σύνθεσης (ή σύνθεσης pitch), που είναι ουσιαστικά μια συνάρτηση ανασύστασης sinc 40 σημείων, χρησιμοποιώντας τις παραμέτρους (υστέρηση και κέρδος) του προσαρμοστικού κωδικοβιβλίου [**LTPSynthesis**]. Το προκύπτων διάνυμα αποτελεί τη συνολική διέγερση, η οποία θα οδηγήσει το φίλτρο LPC σύνθεσης. Επειδή η διαδικασία εκτελείται ανά υποπλαίσιο, είναι αναγκαία η παρεμβολή στις τιμές των LSPs ώστε να δημιουργηθούν οι τιμές για όλα τα υποπλαίσια [**SynthesisInterpolation**]. Μετά την παρεμβολή, οι LSPs μετατρέπονται σε συντελεστές πρόγνωσης [**LspsToPcCoeffs**]. Χρησιμοποιώντας τους συντελεστές αυτούς σε ένα φίλτρο «μόνο πόλων» δημιουργείται το φίλτρο LPC σύνθεσης. Η διέγερσή του με το διάνυμα συνολικής διέγερσης οδηγεί στην σύνθεση ενός υποπλαισίου του αρχικού σήματος φωνής [**PoleFilter**].

4 Θέματα σχετικά με το υλικό

4.1 Εισαγωγή

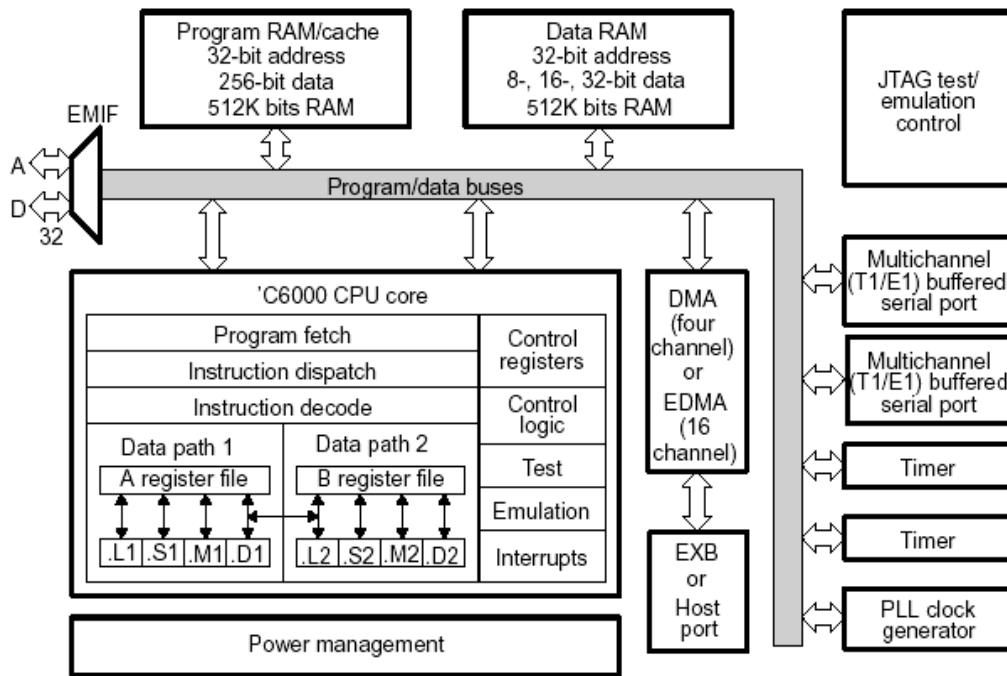
Η υλοποίηση του αλγορίθμου όπως περιγράφηκε στα προηγούμενα κεφάλαια, έγινε για την παρούσα εργασία στον επεξεργαστή TMS320C6701 της Texas Instruments, ο οποίος βρισκόταν πάνω σε μία κάρτα για θύρα PCI, την EVM (Evaluation Module) της ίδιας εταιρίας.

Η απόφαση για τη χρήση του συγκεκριμένου επεξεργαστή πάρθηκε μετά από προσεκτική εξέταση των χαρακτηριστικών όλων των διαθέσιμων επεξεργαστών του εργαστηρίου Ψηφιακής Επεξεργασίας Σήματος, και τη διαπίστωση ότι μόνο ο C6701 παρείχε ασφαλή περιθώρια υπολογιστικής ισχύος και μνήμης ώστε να είναι δυνατή η υλοποίηση ενός τόσο επίπονου υπολογιστικά αλγορίθμου χωρίς εξεζητημένες γνώσεις τεχνικών προγραμματισμού σε γλώσσα assembly (έτσι ώστε ο κώδικας να βελτιστοποιηθεί για ένα μικρότερο επεξεργαστή).

Στο παρόν κεφάλαιο, θα δοθεί μια περιληπτική παρουσίαση των χαρακτηριστικών και των δυνατοτήτων του επεξεργαστή και της κάρτας, καθώς επίσης θα αναλυθεί και ο τρόπος με τον οποίο υλοποιούνται τα ειδικά ως προς το υλικό (hardware specific) μέρη του κώδικα (κυρίως οι ρουτίνες εισόδου-εξόδου του σήματος φωνής).

4.2 Ο επεξεργαστής TMS320C6701

Ο επεξεργαστής C6701 αποτελεί το παλαιότερο αλλά και ισχυρότερο μέχρι σήμερα μέλος της οικογένειας των επεξεργαστών αριθμητικής κινητής υποδιαστολής (floating point) C67x. Η συχνότητα λειτουργίας του είναι 167MHz, που ισοδυναμεί σε χρόνο εκτέλεσης κύκλου (cycle time) της τάξης των 6ns. Η αρχιτεκτονική του επεξεργαστή παρέχει τη δυνατότητα εκτέλεσης 1G FLOPS λειτουργιών απλής ακρίβειας και 688M FLOPS λειτουργιών πολλαπλασιασμού και άθροισης. Όπως και οι συμβατικοί επεξεργαστές αποτελείται από τρία κύρια μέρη: Την κεντρική μονάδα επεξεργασίας (Central Processing Unit – CPU), τη μνήμη και τους περιφερειακούς μηχανισμούς. Τα μέρη αυτά διασυνδέονται με διαύλους (buses), όπως φαίνεται στο Σχήμα 4.1. Στις επόμενες παραγράφους θα περιγραφούν συνοπτικά τα μέρη αυτά.



Σχήμα 4.1 Λειτουργικό διάγραμμα του C6701 (σχέδιο της Texas Instr., SPRU197D)

4.2.1 Πυρήνας επεξεργαστή

Ο πυρήνας του επεξεργαστή (CPU) είναι η μονάδα που οργανώνει, αποκωδικοποιεί και εκτελεί τον κώδικα. Συγκεκριμένα, η CPU περιλαμβάνει:

- (a) Τη μονάδα ελέγχου προγράμματος, που φορτώνει τις εντολές προς εκτέλεση, τις αποκωδικοποιεί και τις κατευθύνει προς το κατάλληλο τμήμα του υλικού που θα τις επεξεργαστεί.
- (b) Τους καταχωρητές ελέγχου, που είναι 13 και χρησιμοποιούνται για τον έλεγχο των διακοπών (interrupts), τη διαχείριση μνήμης, την υποστήριξη λειτουργιών αριθμητικής κινητής υποδιαστολής και τον έλεγχο της λειτουργίας του επεξεργαστή.
- (c) Τους δρόμους δεδομένων (data paths), που υλοποιούν τη μεταφορά των δεδομένων μεταξύ των καταχωρητών της CPU, των μονάδων εκτέλεσης και της μνήμης. Υπάρχουν δύο data paths.
- (d) Τις μονάδες εκτέλεσης προγράμματος, όπου τελικά εκτελούνται οι εντολές του κώδικα. Η εκτέλεση, ανάλογα με το είδος της εντολής, μπορεί να διαρκέσει από έναν (single-cycle instruction) ως έξι κύκλους. Υπάρχουν 8 μονάδες εκτέλεσης, τέσσερις για κάθε data path. Είναι οι:
 - i. Μονάδες .L: Είναι οι Αριθμητικές και Λογικές Μονάδες (ALUs), που εκτελούν αριθμητικές πράξεις, συγκρίσεις, λογικές πράξεις.

- ii. Μονάδες *.M*: Πρόκειται για τους πολλαπλασιαστές. Λόγω του ότι ο πολλαπλασιασμός γίνεται στο επίπεδο του υλικού (hardware), η υστέρηση που εισάγει είναι μόνο δύο κύκλοι.
- iii. Μονάδες *.S*: Εκτελούν επίσης αριθμητικές και λογικές πράξεις, καθώς και διακλαδώσεις κώδικα, πράξεις ολίσθησης και ανάγνωση/εγγραφή των καταχωρητών ελέγχου.
- iv. Μονάδες *.D*: Εκτελούν για πράξεις φόρτωσης και αποθήκευσης, πρόσθεσης και αφαίρεσης, και διαχείρισης διευθύνσεων μνήμης.

Είναι σημαντικό να ειπωθεί ότι το εύρος του διαύλου της εσωτερικής μνήμης προγράμματος (program memory bus) είναι 256bit, δηλαδή ο επεξεργαστής έχει τη δυνατότητα να διαβάσει ταυτόχρονα (παράλληλα) 8 εντολές (8x32bit) αν ο κώδικας βρίσκεται αποθηκευμένος στην εσωτερική μνήμη, αυξάνοντας σημαντικά την ταχύτητα εκτέλεσης του προγράμματος.

4.2.2 Μνήμη

Το εύρος του διαύλου διευθύνσεων του C6701 είναι 32bits, δίνοντας στον επεξεργαστή δυνατότητα να προσπελάσει μέχρι 4 Gbytes μνήμης, που χωρίζεται σε 4 τμήματα. Αυτά είναι: Εσωτερική μνήμη προγράμματος, εσωτερική μνήμη δεδομένων, εσωτερική μνήμη περιφερειακών, εξωτερική μνήμη. Ανάλογα με το είδος της χαρτογράφησης (memory map) που χρησιμοποιείται, τα παραπάνω τμήματα τοποθετούνται σε συγκεκριμένες θέσεις του συνολικού διαθέσιμου χώρου.

Η εσωτερική μνήμη, η μνήμη που βρίσκεται δηλαδή στο ολοκληρωμένο του επεξεργαστή, έχει μέγεθος 128Kbytes και χωρίζεται σε μνήμη προγράμματος (64K) και μνήμη δεδομένων (64K). Όπως ειπώθηκε παραπάνω, η προσπέλαση στην εσωτερική μνήμη προγράμματος γίνεται με διάυλο 256bit ώστε να επιτυγχάνεται μέγιστος παραλληλισμός. Για το λόγο αυτό, τα κομμάτια του κώδικα που αναμένεται να προκαλούν μεγάλο φόρτο εργασίας, συνήθως φορτώνονται στην εσωτερική μνήμη. Η προσπέλαση στην εσωτερική μνήμη δεδομένων γίνεται με διάυλο 32bit, ενώ μπορεί να γίνει διαχείριση τιμών 8bit, 16bit και 32bit, δηλαδή και εδώ υπάρχει η δυνατότητα να φορτωθούν ή να αποθηκευθούν ταυτόχρονα δεδομένα (δύο αριθμοί των 16bit, ένας από κάθε data path).

Η διασύνδεση του επεξεργαστή με τυχόν υπάρχουσα εξωτερική μνήμη, γίνεται μέσω του EMIF (external memory interface), που είναι ουσιαστικά ένας

προγραμματιζόμενος διαχειριστής μνήμης. Υποστηρίζει μνήμες τύπου SDRAM, SBSRAM, ασύγχρονη μνήμη (ROM) κλπ.

4.2.3 Περιφερειακά

Ελεγκτής απευθείας προσπέλασης μνήμης

Για την ταχύτερη μεταφορά δεδομένων, ο C6701 χρησιμοποιεί απευθείας προσπέλαση μνήμης (DMA – direct memory access), δηλαδή προσπέλαση τμημάτων μνήμης χωρίς τη μεσολάβηση της CPU. Υπάρχουν τέσσερα κύρια κανάλια DMA, ανεξάρτητα προγραμματιζόμενα, καθώς και ένα βοηθητικό. Η DMA έχει τη δυνατότητα να μεταφέρει δεδομένα κατά πολλούς τρόπους, είτε ανά στοιχείο, είτε ανά πλαίσιο, είτε ανά τμήμα (συστοιχία πλαισίων).

Σειριακές θύρες

Ο επεξεργαστής C6701 παρέχει δύο σειριακές θύρες (McBSP – multichannel buffered serial ports) για επικοινωνία με εξωτερικά περιφερειακά. Οι θύρες αυτές υποστηρίζουν πλήρως αμφίδρομη επικοινωνία με μέγιστη ταχύτητα τα 40Mbits/s. Υπάρχει δυνατότητα λήψης/εκπομπής μέχρι 128 καναλιών σε συγχρονισμένα πλαίσια, ενώ το μέγεθος των στοιχείων που μεταδίδονται μπορεί να είναι 8, 12, 16, 20, 24 και 32bit με συμπίεση κατά μ-Law ή A-Law. Επίσης, οι θύρες μπορούν να επικοινωνήσουν με τη μνήμη τόσο μέσω της CPU όσο και της DMA.

Διασύνδεση με θύρα Host

Η διασύνδεση του C6701 με έναν host επεξεργαστή, ο οποίος θα έχει τη δυνατότητα απ' ευθείας προσπέλασης της μνήμης του C6701 γίνεται μέσω της διασύνδεσης με τη θύρα Host (HPI – Host port interface). Πρόκειται για μια παράλληλη θύρα για μεταφορά 16bit δεδομένων (ή 32bit με δύο αυτόματες διαδοχικές μεταφορές των 16bit). Η προσπέλαση της μνήμης του C6701 γίνεται από τον host μόνο μέσω του βοηθητικού καναλιού της DMA.

Χρονιστές

Για την δημιουργία interrupts ανά τακτά χρονικά διαστήματα καθώς και για λειτουργίες συγχρονισμού, ο C6701 περιλαμβάνει δύο 32bit χρονιστές (timers). Οι χρονιστές είναι προγραμματιζόμενοι, η μέτρηση γίνεται προς τα πάνω, ενώ η περίοδος μεταξύ δύο αυξήσεων είναι ένα τέταρτο του κύκλου της CPU.

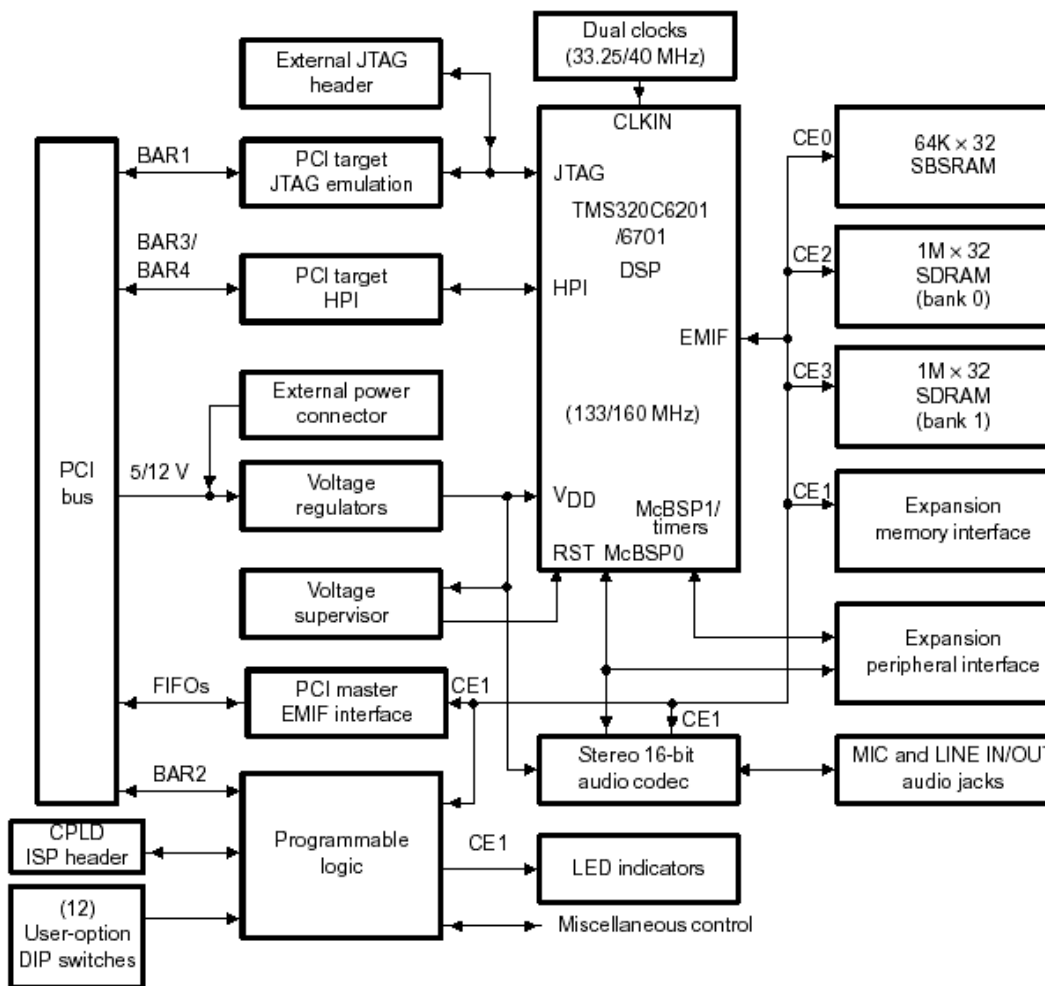
4.3 Η κάρτα EVM

Ο επεξεργαστής C6701 είναι τοποθετημένος σε μία κάρτα EVM (Evaluation Module). Πρόκειται για μια κάρτα ανάπτυξης εφαρμογών, στην οποία είναι δυνατόν ο προγραμματιστής να αξιολογήσει την απόδοση της εφαρμογής που σχεδιάζει πριν αυτή οδηγηθεί στην παραγωγή. Η κάρτα μπορεί να συνδεθεί εξωτερικά σε έναν υπολογιστή, ή εσωτερικά, σε μια θύρα PCI, όπως έγινε στην υλοποίηση που παρουσιάζεται στην εργασία αυτή.

Το λειτουργικό διάγραμμα της EVM εμφανίζεται στο Σχήμα 4.2 και περιλαμβάνει:

- § *Τον επεξεργαστή TMS320C6701.*
- § *Τους χρονιστές του επεξεργαστή (δύο ξεχωριστούς ταλαντωτές που δίνουν στον επεξεργαστή τη δυνατότητα να λειτουργήσει σε 4 διαφορετικές συχνότητες).*
- § *Εξωτερική μνήμη.* Συγκεκριμένα, η EVM παρέχει 64K λέξεων 32bit (ή 256Kbytes) μνήμης SBSRAM στα 7,5ns και 2x1M λέξεων 32bit (8Mbytes) μνήμης SDRAM στα 10ns.
- § *Διασύνδεση PCI*, για την επικοινωνία της κάρτας και του επεξεργαστή με τον host υπολογιστή.
- § *Δυνατότητες προσομοίωσης και ελέγχου σφαλμάτων.*
- § *Δυνατότητες επέκτασης μέσω εξωτερικών περιφερειακών.*
- § *Διαχείριση ήχου (audio codec)*, που περιλαμβάνει προενισχυτή εισόδου και μετατροπείς A/D και D/A με δυνατότητα δειγματοληψίας 5.5 – 48KHz και κβάντισης στα 16bit, όλα στο ολοκληρωμένο CS4231A της Crystal.
- § *Τροφοδοτικά και έλεγχο ισχύος.*

Τόσο τα περιφερειακά του επεξεργαστή όσο και η κάρτα EVM μπορούν να ελεγχθούν από βιβλιοθήκες συναρτήσεων που παρέχει η Texas Instruments, καθώς σε αρκετές περιπτώσεις η μη-αυτόματη ρύθμιση όλων των παραμέτρων που απαιτούνται για σωστή λειτουργία, μέσω των καταχωρητών ελέγχου, είναι αρκετά επίπονη. Επίσης, η Texas Instruments παρέχει τη δυνατότητα, μέσω του λογισμικού ανάπτυξης (για την περίπτωση της παρούσας εργασίας το Code Composer studio v1.02) να βελτιστοποιηθεί αυτόματα ο κώδικας (software optimization) ώστε να επιτευχθεί ταχύτερη εκτέλεση και καλύτερη διαχείριση πόρων.



Σχήμα 4.2 Λειτουργικό διάγραμμα κάρτας EVM (σχέδιο της Texas Instr., SPRU269D)

4.4 Εφαρμογή του κώδικα στην κάρτα EVM

Η εισαγωγή του σήματος φωνής προς επεξεργασία από τον αλγόριθμο, γίνεται μέσω δειγματοληψίας του σήματος που εισέρχεται από το μικρόφωνο (είσοδος φωνής) από τον μετατροπέα A/D του audio codec που βρίσκεται στην κάρτα EVM. Καθώς ο codec αυτός είναι προγραμματιζόμενος, η συχνότητα δειγματοληψίας τέθηκε στα 8000Hz, την πλέον συνηθισμένη για σήματα φωνής όπου το εύρος θεωρείται $\leq 3400\text{Hz}$ άρα η δειγματοληψία εκπληρώνει το κριτήριο Nyquist. Ορίστηκε επίσης ο codec να αποδίδει τα δείγματα σε μορφή 16bit ακεραίων χωρίς συμπίεση (υπάρχει η δυνατότητα για μ-law ή A-law συμπίεση, αν το εισερχόμενο σήμα αναμένεται να έχει μεγάλο δυναμικό εύρος). Η μεταφορά του δειγματοληπτημένου σήματος από τον codec προς επεξεργασία από τον κώδικα γίνεται μέσω του καναλιού 1 της DMA, με χρήση της τεχνικής του διπλού buffering.

Συγκεκριμένα, τα δείγματα αποθηκεύονται μέσω της DMA σε buffer της μνήμης μήκους όσο το μήκος ενός πλαισίου ανάλυσης (240 δείγματα). Μόλις ο buffer αυτός γεμίσει, ενεργοποιείται μια διακοπή (interrupt) στη CPU, που αναλαμβάνει να στείλει τον πλήρη buffer προς επεξεργασία στο κυρίως σώμα του κώδικα (στη ρουτίνα **Encoder**), ενώ ταυτόχρονα κατευθύνει την είσοδο (το κανάλι 1 της DMA) προς ένα δεύτερο buffer, έτσι ώστε να γίνεται ταυτόχρονη επεξεργασία του ενός πλαισίου και εγγραφή του επόμενου χωρίς απώλεια δεδομένων. Η ρουτίνα encoder επιστρέφει μια ακολουθία 138 bits που αντιστοιχούν στις κωδικοποιημένες παραμέτρους για το συγκεκριμένο πλαίσιο του σήματος φωνής, σύμφωνα με τη μεθοδολογία που περιγράφηκε στα προηγούμενα κεφάλαια. Η ακολουθία αυτή μπορεί πλέον να οδηγηθεί σε κατάλληλη ρουτίνα που θα αναλαμβάνει την αποστολή της στο κανάλι. Για τις ανάγκες της υλοποίησης που παρουσιάζεται εδώ, η δυαδική ακολουθία αποστέλλεται απ' ευθείας στον αποκωδικοποιητή [ρουτίνα **Decoder**].

Η decoder επιστρέφει το αποτέλεσμα της σύνθεσης φωνής, που είναι ουσιαστικά ένα πλαίσιο σήματος φωνής και στην ιδανική περίπτωση θα ήταν ταυτόσημο με το αντίστοιχο πλαίσιο που επεξεργάστηκε ο κωδικοποιητής. Προφανώς, κάθε πραγματικός κωδικοποιητής εισάγει έναν υποβιβασμό της ποιότητας του σήματος λόγω των παραδοχών που εισάγονται για τη λειτουργία του καθώς και παραμέτρων όπως σφάλμα κβάντισης, πεπερασμένη ακρίβεια αριθμητικών πράξεων κ.ο.κ. Η εξαγωγή της φωνής γίνεται με όμοιο τρόπο μέσω του καναλιού 2 της DMA, από διπλούς buffers της μνήμης προς τον codec όπου γίνεται η τελική ανασύσταση της φωνής και η εξαγωγή της στα ηχεία.

Η διαδικασία εισόδου (αλλά και εξόδου) του σήματος φωνής ελέγχεται από την ρουτίνα **InitIo**, που καλείται από την **main**. Η ρουτίνα αυτή αρχικοποιεί πρώτα τον audio codec [υπορουτίνα **EnableCodec**], θέτοντας το ρυθμό δειγματοληψίας και τη μορφή των δεδομένων (δειγμάτων) και καθορίζοντας το είδος των εισόδων/εξόδων. Στη συνέχεια γίνεται η αρχικοποίηση της σειριακής θύρας 0 [**EnableMcBSP0**] με τη ρύθμιση του καταχωρητή ελέγχου SPCR και η αρχικοποίηση του ελεγκτή DMA [**EnableDMA**], ώστε να είναι δυνατή η παραγωγή διακοπών της CPU, να τεθούν σε λειτουργία τα κανάλια 1 και 2 της DMA, να καθοριστούν οι διευθύνσεις αποθήκευσης των δεδομένων (buffers) και να οριστεί το μέγεθός τους. Τέλος, ορίζεται και ενεργοποιείται η ρουτίνα εξυπηρέτησης της διακοπής (interrupt service routine) που είναι η **DMA_ISR** και ενεργοποιείται η μεταφορά μέσω της DMA.

5 Αποτελέσματα και συμπεράσματα

5.1 Υλοποίηση κωδικοποιητή – αποκωδικοποιητή σε σειρά

Η πρωταρχική σκέψη για την υλοποίηση ήταν ως εξής: Εισαγωγή φωνής μέσω μικροφώνου, κωδικοποίηση κάθε πλαισίου φωνής από τον επεξεργαστή C6701, αποστολή των παραμέτρων (της δυαδικής ψηφιοσειράς εξόδου) μέσω της θύρας επέκτασης PCI του host υπολογιστή, προσομοίωση ενός καναλιού επικοινωνίας στον υπολογιστή, επιστροφή της ψηφιοσειράς μέσω της θύρας PCI και πάλι στον επεξεργαστή για αποκωδικοποίηση της και έξοδος της συνθετικής φωνής στα ηχεία. Όλη η διαδικασία έπρεπε να γίνεται σε πραγματικό χρόνο, δηλαδή με ελάχιστες καθυστερήσεις (που συνήθως ορίζονται στη βιβλιογραφία σε τάξεις των δεκάδων msec, ανάλογα πάντα του είδους της εφαρμογής).

Στην πορεία της υλοποίησης όμως παρουσιάστηκαν προβλήματα σε σχέση με την επικοινωνία της κάρτας EVM και του host υπολογιστή μέσω της θύρας PCI, καθώς και σε σχέση με τις καθυστερήσεις που εισήγαγε το λειτουργικό του host (Windows 95 και 98) στην διαδικασία. Έτσι, η προσέγγιση αυτή εγκαταλείφθηκε και το ζεύγος κωδικοποιητή – αποκωδικοποιητή τοποθετήθηκε σε σειρά, δημιουργώντας ένα «bypass» που αποστέλλει την έξοδο του πρώτου απ' ευθείας στην είσοδο του δεύτερου, με προοπτική να γίνει εξομοίωση ενός καναλιού επικοινωνίας πάνω στον ίδιο τον επεξεργαστή.

5.2 Λειτουργία σε πραγματικό χρόνο

Ενώ αρχικά η απαίτηση για λειτουργία του κωδικοποιητή σε πραγματικό χρόνο φαινόταν εξασφαλισμένη από τις υψηλές δυνατότητες του υλικού (επεξεργαστής), γρήγορα αποδείχτηκε ότι ο αλγόριθμος ήταν ιδιαίτερα απαιτητικός σε υπολογιστική ισχύ, ώστε να ξεπερνάει αυτές τις δυνατότητες. Πρέπει να διασαφηνιστεί σε αυτό το σημείο, ότι η επεξεργασία κάθε πλαισίου σήματος φωνής θα έπρεπε να έχει ως ανώτατο χρονικό όριο τα 30msec, αριθμός που προκύπτει από τη φυσική διάρκεια του πλαισίου ($\frac{1}{8000} \cdot 240 = 0,03\text{sec}$, όπου 8000Hz είναι η συχνότητα δειγματοληψίας και 240 ο αριθμός των δειγμάτων ανά πλαίσιο). Με άλλα λόγια, η επεξεργασία κάθε

πλαίσιου θα έπρεπε να έχει ολοκληρωθεί το πολύ σε αυτό το διάστημα, καθώς αμέσως μετά εισέρχεται ένα νέο πλαίσιο. Αν δεν έχει γίνει η επεξεργασία του προηγούμενου, αναπόφευκτα χάνονται δείγματα φωνής, γίνονται διενέξεις μεταξύ των διακοπών του επεξεργαστή και η ποιότητα του σήματος μειώνεται δραματικά. Όπως αποδείχτηκε, ο χρόνος εκτέλεσης του αλγορίθμου στην αρχική μορφή του για την επεξεργασία ενός εισερχόμενου πλαισίου προσέγγιζε τα 100msec, πολύ παραπάνω από τα 30msec που ήταν το ζητούμενο, αποκλείοντας τη λειτουργία πραγματικού χρόνου. Έτσι, έπρεπε να βρεθούν ορισμένες στρατηγικές για να υπερπηδηθεί αυτό το πρόβλημα. Ο Πίνακας 5.1 παρουσιάζει τα διάφορα στάδια πειραματισμών, σε σχέση με την βελτίωση σε χρόνο και αριθμό κύκλων στην οποία οδήγησαν.

Το πρώτο τμήμα το οποίο αναθεωρήθηκε ήταν ο κώδικας C της υλοποίησης, προχωρώντας σε μια περιορισμένη (και όχι επαρκή όπως αποδείχτηκε) βελτιστοποίηση. Για να γίνει αυτό, χρησιμοποιήθηκαν αφ' ενός τα εργαλεία λογισμικού που παρέχει η Texas Instruments για βελτιστοποίηση του, παραγόμενου μετά τη μεταγλώττιση, κώδικα assembly και αφ' ετέρου έγινε μια προσπάθεια για βελτίωση του κώδικα από άποψη περιορισμού κλήσεων σε μακρινές συναρτήσεις, ενσωμάτωσης ορισμένων συναρτήσεων σε άλλες, καλύτερης οργάνωσης και εκμετάλλευσης της μνήμης και περιορισμού ορισμένων βρόχων, όπου αυτά ήταν δυνατά. Η βελτίωση που επέφεραν αυτές οι αλλαγές ήταν σημαντική, της τάξης του 65% από άποψη ταχύτητας εκτέλεσης, πέφτοντας στα 32msec για κάθε πλαίσιο. Είναι

Πίνακας 5.1 Αλλαγές στην υλοποίηση και οι βελτιώσεις που επέφεραν

	Αριθμός κύκλων*	Χρονική διάρκεια (msec)**
Αρχική μορφή κώδικα	$\approx 15 \cdot 10^6$	≈ 90
Βελτιώσεις σε επίπεδο ποιότητας κώδικα	5.322.000	31,9
Μείωση αποτελεσματικότητας LTP	4.450.000	26,7
Μείωση μεγέθους στοχαστικού κωδικοβιβλίου	3.090.000	18,6

* Οι αριθμοί αναφέρονται στην εκτέλεση του αλγορίθμου μόνο και δεν περιλαμβάνουν τις ρουτίνες εισόδου-εξόδου, τις αρχικοποιήσεις και το λεγόμενο overhead, που ουσιαστικά έχουν μικρή συνεισφορά στον υπολογιστικό φόρτο.

** Κάθε κύκλος θεωρείται ότι διαρκεί 6nsec, από όπου υπολογίζεται η χρονική διάρκεια.

φανερό όμως ότι και πάλι δεν ήταν επαρκής.

Το επόμενο βήμα ήταν η επικέντρωση στα σημεία που προκαλούν τον μέγιστο υπολογιστικό φόρτο κατά την εκτέλεση και η προσπάθεια περιορισμού του φόρτου αυτού. Τα δύο κύρια σημεία είναι, όπως αναφέρθηκε και σε προηγούμενα κεφάλαια, η εξέταση των κωδικοβιβλίων. Έτσι, πραγματοποιήθηκαν δοκιμές απενεργοποιώντας τμήματα του αλγορίθμου και παρατηρώντας την επίδραση που θα είχε αυτό στην ταχύτητα εκτέλεσης και στην ποιότητα της εξερχόμενης φωνής. Από το τμήμα της πρόγνωσης LTP, επιλέχθηκε τελικά η απενεργοποίηση της αναζήτησης για μη-ακέραιες τιμές υστέρησης που προέβλεπε ο αρχικός αλγόριθμος PFS1016, καθώς θεωρήθηκε εξεζητημένη η υλοποίησή του σε σχέση με τον φόρτο που πρόσθετε στο σύστημα. Η ενέργεια αυτή κατέβασε το χρόνο εκτέλεσης οριακά κάτω από τα 30msec, οπότε η ποιότητα του εξερχόμενου σήματος αυξήθηκε σημαντικά, και η συνθετική φωνή έφτασε στα επίπεδα του κατανοητού. Καθώς όμως υπήρχε η υποψία ότι παράγοντες που δεν λαμβάνονται υπ' όψιν (overheads) ανέβαζαν τον πραγματικό χρόνο εκτέλεσης πάνω από τα 30msec, μειώθηκε δοκιμαστικά και το μέγεθος του στοχαστικού κωδικοβιβλίου από τις 512 λέξεις στις 256. Πράγματι, παρατηρήθηκε αρκετή βελτίωση στην ποιότητα του εξερχόμενου σήματος, ενώ ο εκτιμώμενος χρόνος εκτέλεσης του αλγορίθμου έπεσε στα 18,6msec. Ο Πίνακας 5.2 συνοψίζει τους χρόνους εκτέλεσης του αλγορίθμου τόσο συνολικά, όσο και για τα δύο επιμέρους τμήματά του. Είναι φανερό το πόσο μεγαλύτερες απαιτήσεις υπολογιστικής ισχύος έχει το τμήμα του κωδικοποιητή, που περιλαμβάνει αναζήτηση μέσα στα κωδικοβιβλία, σε σχέση με τον αποκωδικοποιητή.

Όσον αφορά τη συνολική υστέρηση ανάμεσα στην είσοδο και την έξοδο, αυτή μπορεί να υπολογιστεί αθροίζοντας την καθυστέρηση που εισάγει το υπολογιστικό μέρος (όπως μετρήθηκε παραπάνω) με την καθυστέρηση επεξεργασίας του πρώτου πλαισίου (για να εκκινηθεί η επεξεργασία, θα πρέπει να υπάρχει ένα πλήρες πλαίσιο στη μνήμη, άρα υπάρχει εξ' ορισμού μια υστέρηση ενός πλαισίου). Έτσι, η συνολική

Πίνακας 5.2 Χρόνοι εκτέλεσης για την τελική υλοποίηση

	Αριθμός κύκλων	Χρονική διάρκεια (msec)
Κωδικοποιητής	2.643.000	15,9
Αποκωδικοποιητής	472.000	2,83
Codec συνολικά*	3.090.000	18,6

* Λόγω του στατιστικού χαρακτήρα των μετρήσεων, υπάρχει μια μικρή διαφορά ανάμεσα στον χρόνο που μετρήθηκε για το συνολικό codec σε σχέση με το άθροισμα των δύο επιμέρους τμημάτων.

υστέρηση θα είναι 30msec (διάρκεια ενός πλαισίου) + 18,6msec (επεξεργασία πλαισίου) = 48,6 msec, χωρίς να λαμβάνονται υπ' όψιν οι σχετικά μικρές υστερήσεις των κυκλωμάτων A/D, D/A κλπ. Η επίδοση αυτή δεν θα χαρακτηριζόταν απολύτως πραγματικού χρόνου στη βιβλιογραφία, εντούτοις όμως κατά τη διάρκεια των δοκιμών δεν προκάλεσε ιδιαίτερη ενόχληση.

Είναι προφανές τέλος, ότι η μείωση του μεγέθους του στοχαστικού κωδικοβιβλίου κατά το ήμισυ, θα προκαλούσε αισθητή πτώση στην ποιότητα του παραγόμενου σήματος φωνής. Κάτι τέτοιο όμως θα γινόταν εμφανές μόνο σε ένα σύστημα που θα είχε ικανή ισχύ να αντεπεξέλθει στον αυξημένο φόρτο του μεγαλύτερου κωδικοβιβλίου. Στην παρούσα εργασία θεωρήθηκε ότι το κέρδος σε υπολογιστικό φόρτο αντισταθμίζει την απώλεια ποιότητας, η οποία εμφανίζεται κυρίως σε περιοχές «άφωνου» λόγου (σύμφωνα) καθώς και με τη μορφή ενός ελαφριού θορύβου στο υπόβαθρο του σήματος εξόδου, πράγμα αναμενόμενο καθώς ο στόχος του κωδικοβιβλίου είναι η μοντελοποίηση αυτών ακριβώς των τμημάτων της φωνής και όσο μικρότερο είναι το μέγεθός του, τόσο λιγότερο καλύπτει τις δυνατές ακολουθίες διέγερσης. Αυτό αποδείχθηκε και από μια περαιτέρω μείωση του μεγέθους του στοχαστικού κωδικοβιβλίου στις 128 λέξεις, όπου παρατηρήθηκε ότι προκαλείται μεγάλη απώλεια ποιότητας, ενώ το κέρδος σε υπολογιστικό φόρτο δεν είναι σημαντικό, καθώς ήδη από πριν καλυπτόταν το όριο για λειτουργία πραγματικού χρόνου.

5.3 Προσομοίωση σε Matlab

Τα παραπάνω επιβεβαιώθηκαν και από προσομοιώσεις σε Matlab, όπου έγιναν διαδοχικές δοκιμές για διαφορετικούς ομιλητές με στοχαστικά κωδικοβιβλία μεγέθους 256 και 512 λέξεων. Τα αποτελέσματα βασίζονται αποκλειστικά και μόνο σε υποκειμενικές ακουστικές δοκιμές, αφού όπως αναφέρεται εκτενώς στη βιβλιογραφία δεν έχει αναπτυχθεί ικανοποιητικό αντικειμενικό εργαλείο μέτρησης ποιότητας εξαγόμενης φωνής για τη συγκεκριμένη οικογένεια κωδικοποιητών.

Παρατηρήθηκε ότι πράγματι υπάρχει σημαντική υποβάθμιση της ποιότητας του εξερχόμενου σήματος μετά τη μείωση του μεγέθους του κωδικοβιβλίου. Επίσης, η μεταβολή δεν φάνηκε να επηρεάζει περισσότερο κάποια δείγματα φωνής από άλλα (π.χ. γυναικείες φωνές από ανδρικές), πράγμα αναμενόμενο αφού οι θεμελιώδεις συχνότητες παίζουν σημαντικό ρόλο κατά την πρόγνωση LTP και όχι κατά το στάδιο

της διέγερσης. Αντίθετα, επιβεβαιώθηκε η σημαντική συνεισφορά του στοχαστικού κωδικοβιβλίου στην περιγραφή των «άφωνων» περιοχών του λόγου. Για παράδειγμα, σε μια φράση με πολλά φωνήεντα, η διαφορά μεταξύ των δύο δοκιμών (256 και 512 λέξεων) ήταν πολύ μικρή, ενώ σε μια φράση με περισσότερα φωνήεντα, ιδιαίτερα τα συριστικά, η διαφορά ήταν εμφανής.

Για τις προσομοιώσεις αυτές χρησιμοποιήθηκε ο κώδικας Matlab των *Spanias* και *Painter*, καθώς μια πρώτη προσπάθεια που καταβλήθηκε για μεταφορά του υπάρχοντος κώδικα σε Matlab κατέληξε σε ένα υπερβολικά αργό και προβληματικό εργαλείο. Ακόμα όμως και με αυτόν τον κώδικα, αξίζει να σημειωθεί ότι η επεξεργασία μόνο 2 sec (67 πλαισίων) φωνής, διαρκεί περισσότερο από ένα λεπτό σε σύστημα με επεξεργαστή Celeron 733MHz και μνήμη 256MB.

5.4 Μελλοντική εργασία

Η παρούσα εργασία θα μπορούσε να βελτιωθεί, να συνεχισθεί και να επεκταθεί από πολλές απόψεις. Οι δύο κύριοι λόγοι για τους οποίους δεν έχουν εφαρμοστεί εδώ, είναι η έλλειψη χρόνου και η απουσία πείρας πάνω σε ορισμένα από τα αντικείμενα Παρακάτω θα δοθούν μερικές προτάσεις:

5.4.1 Περαιτέρω βελτίωση του κώδικα – ενσωμάτωση νέων τμημάτων στον αλγόριθμο

Λόγω μικρής εξοικείωσης με την assembly του συγκεκριμένου επεξεργαστή (C6701), δεν ήταν δυνατόν να γραφτούν κάποια τμήματα του κώδικα απ' ευθείας σε assembly. Κάτι τέτοιο όμως, αν εφαρμοζόταν στα τμήματα εκείνα που καταναλώνουν την περισσότερη υπολογιστική ισχύ (κυρίως στις συνελίξεις που εκτελούνται για την εύρεση του κέρδους κατά την εξέταση των κωδικοβιβλίων), θα ήταν ευεργετικό για τις επιδόσεις του αλγορίθμου.

Ένα ακόμα σημείο θα μπορούσε να είναι η διερεύνηση και ενσωμάτωση νέων τεχνικών που προτείνονται στη νεότερη βιβλιογραφία για ταχύτερη και αποτελεσματικότερη εκτέλεση της εξέτασης των κωδικοβιβλίων.

5.4.2 Βελτίωση του τελικού φίλτρου για εξάλειψη θορύβου

Είναι γνωστό ότι τα περισσότερα περιβάλλοντα απέχουν πολύ από το να μπορούν να θεωρηθούν ήσυχα. Επίσης, τα μικρόφωνα που χρησιμοποιούνται για την

πραγματοποίηση των συνομιλιών, σπάνια είναι καλής ποιότητας. Για παράδειγμα, κατά τις δοκιμές για τη συγκεκριμένη εργασία, χρησιμοποιήθηκε ένα κοινό, φθινό μικρόφωνο ηλεκτρονικού υπολογιστή. Το αποτέλεσμα των παραπάνω είναι η εισαγωγή ανεπιθύμητου θορύβου στον κωδικοποιητή, που καταρρακώνει την ποιότητα της συνθετικής φωνής στην έξοδο. Το τελικό φίλτρο εξόδου του κωδικοποιητή θα μπορούσε να βελτιωθεί ώστε να προσαρμόζεται με τέτοιο τρόπο που να ακυρώνει το θόρυβο, ή έστω ποσοστό του.

5.4.3 Μελέτη ανοχής του κωδικοποιητή σε σφάλματα

Για την πλήρη μελέτη της ανοχής του κωδικοποιητή σε σφάλματα, θα έπρεπε να προσομοιωθεί ένα ελεγχόμενο κανάλι (είτε με επικοινωνία με τον host υπολογιστή, είτε με επικοινωνία με άλλον αντίστοιχο επεξεργαστή DSP που θα τρέχει ή όχι τον ίδιο codec, είτε τέλος και με προσομοίωση του καναλιού στον ίδιο επεξεργαστή, αρκεί να έχει γίνει η κατάλληλη βελτίωση του κώδικα ώστε να είναι δυνατή η διαχείριση του επιπλέον φόρτου). Το κανάλι αυτό θα έπρεπε να έχει ρυθμιζόμενα και γνωστά χαρακτηριστικά για να μπορεί να γίνει η εκτίμηση του εισαγόμενου σφάλματος και να υπολογίζεται η απόκριση του codec. Παραπέρα βελτίωση θα μπορούσε να είναι η εισαγωγή κωδικοποίησης καναλιού (π.χ το αρχικό πρότυπο PFS1016 προτείνει κωδικοποίηση Hamming).

6 Βιβλιογραφία

- 1) Barnwell Th. P., Nayebi K., Richardson C. H., *Speech coding, a computer laboratory textbook*, John Wiley & Sons, 1996.
- 2) Campbell J., Tremain T., Welch V., “The DoD 4.8 kbps Standard (Proposed Federal Standard 1016)”, *Advances in Speech Coding*, edited by Atal et al, Kluwer Academic Publishers, 1991.
- 3) Dahnoun N., *Digital signal processing implementation using the TMS320C6000 DSP platform*, Prentice Hall, 2000.
- 4) European Telecommunications Standards Institute, *Enhanced full rate (EFR) speech transcoding (GSM 06.60 version 8.0.1 Release 1999)*, ETSI, 2000.
- 5) Hayes H. M., *Statistical digital signal processing and modelling*, John Wiley & Sons, 1996.
- 6) Kondoz A. M., *Digital speech: Coding for low bit rate communication systems*, John Wiley & Sons, 1994.
- 7) Langi A., Grieder W., Kinsner W., “Fast CELP algorithm & implementation for speech compression”, *Proc. of digital communications conf.*, University of Manitoba, Canada, 1994.
- 8) Lin C-K., *Low complexity codebook searching algorithms for FS1016*, M. Sc. Thesis at the NCT University, Taiwan, 2002.
- 9) National Communications System, “Details to assist in implementation of Federal standard 1016 CELP”, *Technical information bulletin*, NCS, 1992.
- 10) Rabiner L. R., Schafer R. W., *Digital processing of speech signals*, Prentice Hall, 1978
- 11) Texas Instruments, *TMS320C6000 Technical brief (SPRU197D)*, 1999.
- 12) Texas Instruments, *TMS320C6201/6701 Evaluation module technical reference (SPRU305)*, 1998.