Compressione della voce a 2.4 Kbit/s

Giovanni Motta

CEFRIEL

Via Emanueli 15, 20126 Milano

Tel. (02)66100083, Fax. (02)66100448

e-mail: dsp@mailer.cefriel.it

18 giugno 1993

Abstract

Il presente lavoro descrive la realizzazione di un codificatore per la voce a basso bit-rate. Dopo una descrizione generale del codificatore proposto, vengono analizzate in dettaglio le scelte innovative o poco note in letteratura, descritte le simulazioni effettuate, discussi i risultati ottenuti e gli eventuali sviluppi futuri.

Contents

Chapter 1

Introduzione

La codifica della voce a basso bit-rate è un problema particolarmente sentito in settori dove vi sia il problema di allocare più utenti in canali con forti limitazioni di banda (ad es. il radiomobile). Le proposte di codificatori a basso bit-rate note in letteratura, possono essere suddivise in due grandi famiglie; quelle che si basano sul VOCODER classico [4] e quelle che utilizzano invece schemi di Analysis-by-Syntesis come ad esempio il CELP [5]. I codec basati sul VOCODER consentono di ottenere un bit-rate molto basso con una qualità che risulta variabile e dipendente dal parlatore; i codec ispirati al CELP, invece usando un bit-rate più alto ed un modello retroazionato, ottengono una qualità migliore e più omogenea che però peggiora notevolmente a bit-rate inferiori ai 4.0 kbit/s. Il peggioramento della qualità della voce per i codec di tipo CELP a bit-rate bassi, è da attribuire principalmente alla non accurata rappresentazione dell'eccitazione, ed alla scadente quantizzazione dei parametri del filtro di ricostruzione.

Il modello proposto, ispirato ad un articolo di Atal [1], ha come obbiettivo quello di ovviare a questi inconvenienti migliorando la rappresentazione dell'eccitazione distinguendo tra tratti di voce quasi periodici (vocalizzati) e non periodici, e sintetizzando i primi con una eccitazione del tipo "single-pulse", ed i secondi usando un codebook stocastico.

In questo modo si pensa di riprodurre con naturalezza i tratti di voce quasi periodici, preservando il pitch (periodo fondamentale della voce) del parlatore. Particolare attenzione è posta nella quantizzazione dei parametri in gioco; i coefficienti del filtro sono quantizzati ricorrendo ad un quantizzatore a traliccio, uno schema *ad hoc* è invece proposto per le posizioni degli impulsi.

Chapter 2

La Struttura del CODEC

La Figura 2.1 mostra la struttura generale del codificatore proposto.

Il segnale vocale in ingresso, ottenuto campionando la voce ad un rate di 8kHz con una risoluzione di 16 bit per campione (in totale un bit-rate di 128 Kbit/s), viene suddiviso in frame di 240 campioni, a loro volta scomposte in tre subframe di 80 campioni ciascuna. Le singole subframe sono classificate come periodiche o non periodiche usando un metodo basato sul calcolo della funzione di autocorrelazione; inoltre per ciascuna di esse, viene stimato il periodo di pitch.

Il valore di pitch viene corretto al fine di aumentarne l'affidabilità e filtrato con un filtro mediano. Il valore così ottenuto viene usato come riferimento per la generazione dell'eccitazione nelle subframe periodiche.

Tre subframe consecutive (una frame), vengono analizzate con il metodo della predizione lineare allo scopo di determinare i parametri del filtro di ricostruzione. La predizione viene effettuata con il metodo del traliccio o di Burg, utilizzando un predittore di ordine dieci; i parametri del filtro vengono posti in una forma conveniente per la quantizzazione (LSF o Line Spectral Frequencies) e codificati con un quantizzatore a traliccio.

L'eccitazione per la sintesi delle singole subframe dipende dalla classificazione periodica/non periodica (P/NP) effettuata. Se la subframe è di tipo non periodico, l'eccitazione viene scelta in un codebook stocastico contenente tratti di rumore bianco gaussiano e la scelta del vettore ottimo viene effettuata minimizzando l'errore quadratico medio come nel CELP. L'indice del vettore prescelto nel codebook fornisce in questo caso la rappresentazione quantizzata dell'eccitazione. Se invece la subframe è di tipo periodico, viene determinata una sequenza di impulsi (uno per ogni periodo di Pitch) che minimizza una particolare funzione di costo.

Partendo da una sequenza di impulsi in posizioni particolarmente favorevoli, un algoritmo di programmazione dinamica, minimizza la funzione di costo, per selezionare gli impulsi che, sintetizzati, consentono non tanto di ottenere una forma d'onda il più possibile simile all'originale, quanto un suono che ne rispetti le caratteristiche spettrali mantenendo uniformità e coerenza col pitch stimato.

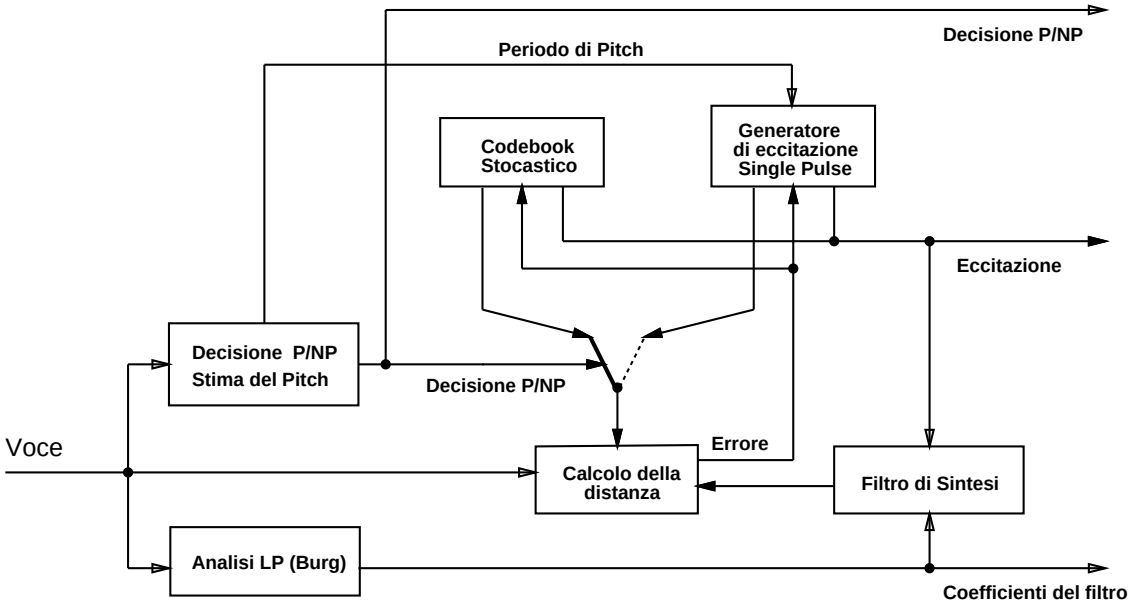

Figure 2.1: La struttura del CODEC.

Allo scopo, tale funzione contiene diversi termini che quantificano il rapporto segnale-rumore del ricostruito, la differenza di ampiezza tra due impulsi consecutivi e lo scostamento della spaziatura tra gli impulsi dal periodo di Pitch stimato.

L'algoritmo di programmazione dinamica è una versione modificata dell'algoritmo di Viterbi.

Le posizioni e le ampiezze degli impulsi vengono quantizzate con uno schema *ad hoc*. Per aumentare l'omogeneità del suono sintetizzato, vengono interpolati sia i parametri del filtro, sia le ampiezze degli impulsi.

Chapter 3

La Stima del Pitch

Il segnale vocale risulta quasi stazionario se considerato in intervalli di tempo sufficientemente limitati. L'analisi di tali intervalli porta naturalmente a distinguere due tipi di andamento:

- tratti vocalizzati (quasi periodici);
- tratti non vocalizzati.

Questa distinzione (alla base di tutti i codificatori di tipo VOCODER), viene sfruttata in fase di analisi e di sintesi, scegliendo in funzione di essa un'eccitazione opportuna. In presenza di un tratto di segnale periodico, è inoltre necessario stimare nel modo più affidabile possibile il valore del periodo di pitch (il periodo della frequenza fondamentale della voce).

La stima periodico/non periodico viene effettuata su finestre di segnale di 80 campioni, considerando anche una parte del segnale passato.

Allo scopo di ottenere decisioni più affidabili, il segnale è preelaborato limitando i valori che può assumere a $+1$, 0 e -1.

Viene poi calcolato il massimo tra i valori dell'autocorrelazione per valori di "lag" temporali compresi tra 20 e 127 campioni; tale valore, è utilizzato come un indicatore della periodicità della frame.

Il lag che ottiene il valore massimo dell'autocorrelazione, opportunamente corretto, viene usato come stima del periodo di pitch.

3.1 Il metodo dell'autocorrelazione

Diversi sono i metodi proposti in letteratura per la stima del periodo di pitch della voce, uno tra quelli che risulta essere un ragionevole compromesso tra affidabilità della stima e complessità di calcolo, utilizza la funzione di autocorrelazione del segnale.

Il metodo si basa sul calcolo di tale funzione per ogni frame di segnale da classificare :

$$R(i) = \sum_{n=i}^{N} x(n) \cdot x(n-i) \tag{3.1}$$

dove $x(n)$ è il segnale vocale, N è la lunghezza della frame e $20 < i < 127$.

La funzione $R(i)$ assumerà valori tanto più elevati, quanto più il segnale $\mathbf{x}(n)$ e la sua traslazione $\mathbf{x}(n-i)$ risultano correlati; in particolare, nel caso di un segnale periodico, essa assumerà il suo valore massimo per valori di i corrispondenti a traslazioni di lunghezza multipla del periodo del segnale. Nel caso di segnali quasi periodici (quali sono quelli di tipo vocale), è ragionevole aspettarsi il valore massimo in corrispondenza del periodi di pitch, tuttavia non è raro che tale massimo sia raggiunto in corrispondenza di un multiplo di tale periodo.

Il compito principale di un'euristica per la determinazione del periodo di pitch è dunque quello di confrontare i picchi assunti dalla funzione di autocorrelazione in corrispondenza dei multipli del pitch stimato con il suo valore massimo, per decidere se accettare o meno tale picco come rappresentante del periodo di pitch reale. Il criterio utilizzato, può essere così sintetizzato:

- La funzione di autocorrelazione viene calcolata per lag temporali compresi tra i 20 ed i 127 campioni.
- Viene determinato il suo valore massimo, sia questo R_{max}, ed il lag che lo ottiene fornisce una prima stima del pitch.
- Si confrontano i valori assunti dalla funzione nell'intorno dei multipli del pitch stimato con il valore massimo.
- Se in corrispondenza di alcuni di questi intorni la funzione assume un valore che è maggiore di $(0.85 \cdot R_{max})$, il minimo tra i lag corrispondenti costituisce la misura definitiva del pitch; se ciò non dovesse accadere, viene tenuta per buona la stima originale.

Osservando la 3.1, è evidente l'uso di una finestra rettangolare che contribuisce ad evitare le stime di periodi multipli alle basse frequenze, attenuando $R(i)$ per valori elevati di i.

La stima di periodicità di una frame viene invece effettuata calcolando l'autocorrelazione sul segnale originale con un lag pari al pitch stimato, il valore ottenuto è confrontato con una soglia S_{voiced} e per valori sopra tale soglia, la subframe è stimata P.

Allo scopo di rendere la decisione largamente indipendente dall'energia del segnale in analisi, i valori ottenuti vengono normalizzati come segue :

$$R_n(i) = \frac{R(i)}{R(0)} \tag{3.2}$$

Il valore utilizzato per la soglia di periodicità è $S_{voiced} = 0.5$.

Nonostante il metodo esposto risulti intuitivo ed efficace, talvolta la stima del periodo di pitch risulta poco affidabile a causa del rumore e delle armoniche presenti nel segnale; emerge quindi la necessità di operare delle trasformazioni sul segnale atte a facilitare il compito dell'algoritmo.

3.2 Il preprocessing con l'ANUSC

Al fine di aumentare l'affidabilità della stima del pitch con il metodo dell'autocorrelazione, il segnale viene sottoposto ad un preprocessing consistente in un "clipping" del segnale tra i valori $+1$, 0 e -1. L'algoritmo utilizzato, proposto per la prima volta in [7], prende il nome di ANUSC (acronimo di Adaptive Non Uniform Sign Clipping); esso usa una soglia che viene fatta decadere con una legge esponenziale per limitare il segnale tra i valori indicati.

Il segnale in analisi, viene scomposto in due sequenze, quella dei valori positivi e quella dei valori negativi. I valori positivi sono trattati come segue:

- La soglia di clipping viene inizializzata ad un valore opportuno (ad es. il valore assunto alla fine della frame precedente).
- Il segnale viene scandito campione per campione e quando il campione in esame supera la soglia di clip, questa viene tenuta costante fino a quando il segnale non ritorna sotto soglia.
- In questo periodo viene tenuta traccia del valore massimo assunto dal segnale sopra soglia (sia questo valore indicato con MAX).
- Quando il valore dei campioni è sotto soglia, il valore assunto dal segnale dopo il clipping è 0, altrimenti esso assume valore $+1$.
- Dal momento in cui il segnale scende sotto soglia, la soglia decade come:

$$T = MAX \cdot e^{\frac{-t}{\beta \cdot P_{av}}} \tag{3.3}$$

  dove t è il tempo, P_{av} è il pitch medio e β è una costante con valore 0.72.

- Il valore P_{av} è aggiornato frame per frame con la regola

$$P_{av-new} = \frac{(P_{av} + P)}{2} \tag{3.4}$$

  dove P assume il valore del pitch stimato.

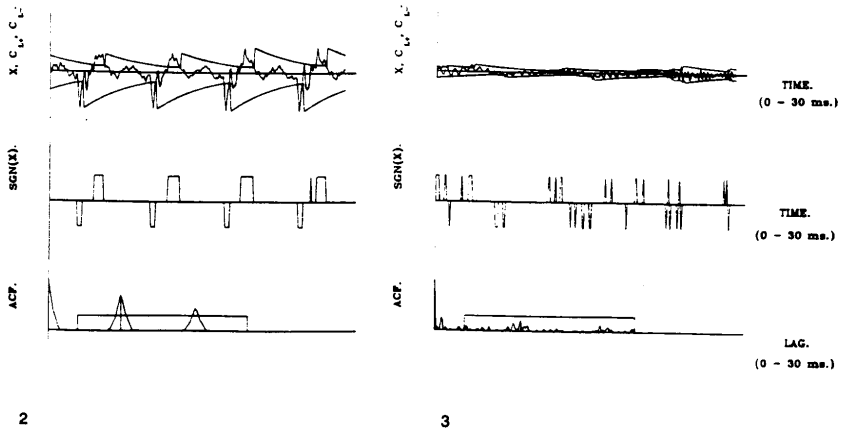

Figure 2: ANUSC on voiced speech; 3: unvoiced speech

Figure 3.1: Esempio di applicazione dell'ANUSC.

Il clipping per la parte negativa del segnale, è ottenuto allo stesso modo, assegnando però i valori 0 e -1.

La composizione (somma) campione per campione dei due risultati fornisce l'uscita dell'algoritmo. La Figura 3.1 (tratta da [7]) mostra il risultato dell'algoritmo ed i valori dell'autocorrelazione per due tratti di voce rispettivamente periodica e non periodica.

3.3 Le correzioni di pitch

Si è già accennato in precedenza alla necessità di apportare al pitch stimato ed alla decisione P/NP delle correzioni allo scopo di rendere più uniforme il segnale ricostruito eliminando eventuali errori isolati.

La tecnica utilizzata nel codec proposto consiste nell'uso di un filtro mediano a cinque, centrato sulla decisione (di pitch o di P/NP) da correggere. Con un filtro di questo tipo è possibile eliminare errori isolati ed evitare stime che differiscano troppo da quelle prese nell'intorno considerato.

È stato altresì utile implementare una correzione di pitch proposta in [6] e chiamata "Post Hoc Editing". Con questa tecnica si intende correggere la stima di pitch errata su una singola subframe. Ecco una descrizione della procedura:

- Siano P_{i-1}, P_i e P_{i+1} i periodi di pitch stimati per tre subframe consecutive (sono espressi in ms.).
- Si ponga $\theta = 0.8 + 0.1 \cdot P_i$.

- Se P_i differisce da P_{i-1} e da P_{i+1} di una quantità maggiore di θ e $|P_{i+1} - P_{i-1}| < \theta$ allora si pone P_i uguale al valore medio tra P_{i+1} e P_{i-1}.

Anche il valore dell'autocorrelazione utilizzato per la stima P/NP viene filtrato nel seguente modo prima di essere confrontato con S_{voiced}:

$$y(n) = y(n-2) \cdot 0.1 + y(n-1) \cdot 0.2 + x(n) \cdot 0.4 + x(n+1) \cdot 0.2 + x(n+2) \cdot 0.1 \quad (3.5)$$

Chapter 4

La Generazione dell'Eccitazione

L'eccitazione viene generata in questo codec in funzione della stima P/NP. Nel caso di subframe periodica, la stima di pitch interviene in modo significativo nella determinazione della sequenza di impulsi.

4.1 Il caso non periodico

Quando una subframe di 80 campioni viene stimata non-periodica, il segnale che ad essa corrisponde è generalmente di tipo rumoroso.

In questo caso l'eccitazione che viene utilizzata per il filtro di sintesi proviene da un codebook formato da vettori di rumore gaussiano a spettro bianco.

La selezione del vettore che meglio approssima il segnale, viene ottenuta come nel CELP [5], ricercando nel codebook quello che minimizza l'errore quadratico medio tra il segnale sintetico e l'originale (Figura 4.1).

Allo scopo, i vettori del codebook vengono filtrati con i parametri del filtro e confrontati con il tratto di segnale originale per il calcolo dell'errore. L'indice di quello che ottiene il minimo errore quadratico medio, viene considerato il valore quantizzato dell'eccitazione. Risulta inoltre necessario specificare il guadagno dell'eccitazione dal momento che i vettori presenti nel codebook sono normalizzati.

Il codebook utilizzato consiste di 256 vettori.

4.2 Il caso periodico

Diverso, e ben più interessante, è il caso di una subframe stimata periodica; questa viene sintetizzata usando un treno di impulsi di ampiezza opportuna distanti tra loro circa un periodo di pitch. Il calcolo della sequenza ottimale di impulsi è l'operazione chiave del codificatore e viene effettuata come segue.

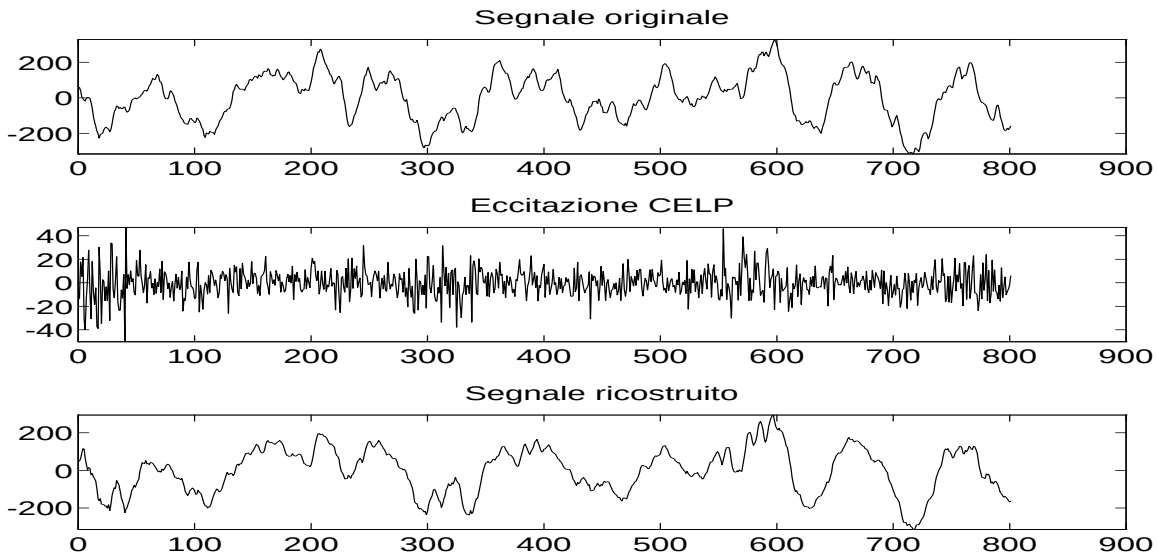

Figure 4.1: Un esempio di sintesi CELP.

Si genera una frame di riferimento (detta da ora in poi "optimization frame") più lunga degli 80 campioni stimati come periodici. Lo scopo è quello di avere un tratto di segnale che contenga più periodi di pitch.

La optimization frame viene ottenuta facendo seguire alla subframe da codificare altre subframe stimate come periodiche e prolungando questo segnale con l'evoluzione libera del filtro LPC fino ad ottenere un riferimento di 480 campioni.

Una prima selezione delle posizioni nelle quali è più conveniente posizionare gli impulsi dell'eccitazione viene effettuata ricercando i massimi di una funzione.

Con questa funzione viene stimata l'efficacia di ogni singolo impulso nella fase di ricostruzione, calcolando il massimo rapporto segnale-rumore ottenibile eccitando il filtro LPC con un singolo impulso di ampiezza ottima α posizionato all'istante di tempo n.

Sia

$$\mathbf{h} = [h(0), h(1), \ldots, h(2N-1)]^T \tag{4.1}$$

la risposta all'impulso del filtro di sintesi LP . Questo vettore è calcolato per ciascuna subframe di lunghezza $N = 80$. Il vettore di errore $\mathbf{e}_n$ è definito come la differenza tra il vettore del segnale $\mathbf{x}$ ed il vettore della risposta all'impulso ritardata $\mathbf{h}_n$ moltiplicato per l'ampiezza dell'impulso α:

$$\mathbf{e}_n = \mathbf{x} - \alpha \mathbf{h}_n, \tag{4.2}$$

dove

$$\mathbf{x} = [x(0), x(1), \ldots, x(2N-1)]^T, \tag{4.3}$$

e

$$\mathbf{h}_n = [0, \ldots, 0, h(0), h(1), \ldots, h(2N-1-n)]^T, \quad n = 0, \ldots, N-1. \tag{4.4}$$

Minimizzando l'energia dell'errore rispetto all'ampiezza dell'impulso α si ottiene

$$\min_{\alpha} \mathbf{e}_n^T \mathbf{e} = \mathbf{x}^T \mathbf{x} - \frac{(\mathbf{x}^T \mathbf{h}_n)^2}{\mathbf{h}_n^T \mathbf{h}_n}, \tag{4.5}$$

$$\alpha_{opt}(n) = \frac{\mathbf{x}^T \mathbf{h}_n}{\mathbf{h}_n^T \mathbf{h}_n}, \tag{4.6}$$

e

$$\max_{\alpha} SNR(n) = 10 \log_{10} \left(\frac{\mathbf{x}^T \mathbf{x}}{\min_{\alpha} \mathbf{e}_n^T \mathbf{e}_n} \right). \tag{4.7}$$

Ricercando i massimi della funzione $\max_{\alpha} SNR(n)$ che risultano avere un'ampiezza $\alpha_{opt}(n)$ positiva, e tenendo conto di un certo numero di posizioni nell'intorno del massimo (nel nostro caso si considera un intervallo di ampiezza cinque centrato sulle posizioni ottime), si forma l'insieme Z come

$$Z = \{ z_i, \quad i = 1, \ldots, M \mid z_i = (n_i, \quad \alpha_{opt}(n_i), \quad \max_{\alpha} SNR(n_i)) \}. \tag{4.8}$$

Gli elementi di Z sono definiti come triple consistenti nella posizione n_i dell'impulso, nell'ampiezza ottima $\alpha_{opt}(n_i)$ e nel massimo rapporto segnale-rumore $\max_{\alpha} SNR(n_i)$ di un singolo impulso nella posizione candidata.

Sotto i vincoli di un intervallo minimo e massimo tra gli impulsi, si definisce l'insieme S come l'insieme di tutti i possibili sottoinsiemi s di Z.

Il passo finale della procedura consiste nel determinare il sottoinsieme s_{opt} di S migliore rispetto ad un criterio di ottimalità che, calcolato per ogni $s \in S$, tenga conto di vari fattori, quali la massimizzazione del rapporto segnale-rumore e la minimizzazione della inconsistenza tra le ampiezze e gli intervalli degli impulsi [1].

Il sottoinsieme migliore, è

$$s_{opt} = \{ z_{q_1}, \ldots, z_{q_K} \}, \quad K \geq 1 \tag{4.9}$$

di indici

$$Q = \{ q_1, \ldots, q_K \ : \ q_k \in [1, M], \ n_{q_k} > n_{q_{k-1}} \} \tag{4.10}$$

che minimizza il costo totale di

$$\min_{S} \frac{1}{K} \left(c_{ini}(i = q_1) + \sum_{l=2}^{K} c(i = q_l, \ j = q_{l-1}) \right) \tag{4.11}$$

rispetto alla funzione di costo

$$c(i, j, k) = \frac{a}{\max_{\alpha} SNR(n_i)} + b \left| \ln \frac{\alpha_{opt}(n_i)}{\alpha_{opt}(n_j)} \right| + c \left| \ln \frac{n_i - n_j}{\overline{p}(n_i)} \right|, \quad n_i > n_j. \tag{4.12}$$

La funzione di costo 4.12, consta della somma di tre termini: il primo penalizza candidati con un basso rapporto segnale-rumore, il secondo termine penalizza le inconsistenze nelle ampiezze di due successivi candidati all'impulso, il terzo, infine, penalizza le differenze tra l'intervallo tra due candidati all'impulso ed il periodo di pitch $\overline{p}(n_i)$ stimato.

Nella prima frame di ottimizzazione che segue ad una transizione NP-P, il costo iniziale $c_{ini}(i = q_1)$ è calcolato come

$$c_{ini}(i) = \begin{cases} \frac{a}{\max_\alpha SNR(n_i)} + c \ln \frac{n_i}{\overline{p}(n_i)} + c_{fix}, & n_i > \overline{p}(n_i) \\ \frac{a}{\max_\alpha SNR(n_i)} + c_{fix}, & n_i \leq \overline{p}(n_i) \end{cases} \tag{4.13}$$

mentre nelle successive il costo iniziale $c_{ini}(i = q_1) = c(i = q_1,\ j = q_0) + c_{fix}$ dove $c(i, j)$ è ottenuta dalla 4.13 con n_j definita come la posizione dell'ultimo impulso nella precedente optimization frame. Collegando la frame attuale con la precedente in questo modo, si cerca di ottenere una certa continuità nella sintesi.

Per la ricerca del sottoinsieme ottimo di impulsi, si è utilizzato l'algoritmo di Viterbi, in cui i nodi rappresentano i candidati all'inpulso per un certo intervallo di posizioni (intorno di un generico massimo locale della funzione SNR), e la metrica associata agli archi è la funzione di costo 4.12.

Il termine c_{fix} è come i termini a, b, c una costante.

I valori per queste costanti sono stati determinati sperimentalmente come uguali a :

$$\begin{aligned} a &= 1.0 \\ b &= 0.05 \\ c &= 1.0 \\ c_{fix} &= 1.0. \end{aligned} \tag{4.14}$$

Si ricalcolano infine i valori delle ampiezze tenendo conto dell'effetto reciproco che gli impulsi hanno in una subframe, questa operazione viene portata a termine risolvendo un sistema di equazioni lineari.

La Figura 4.2 illustra il risultato di tale procedimento.

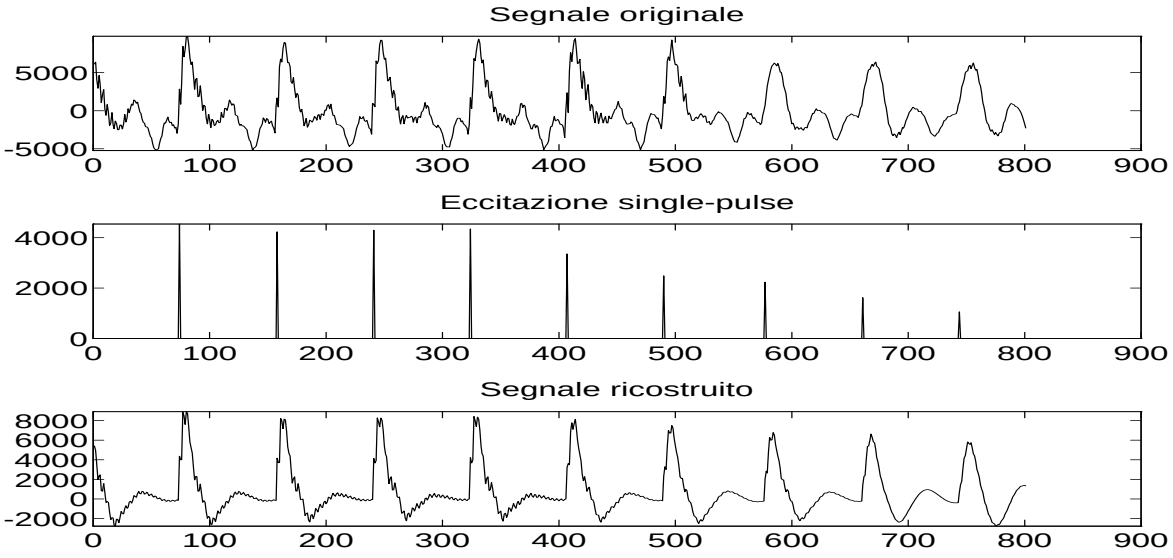

Figure 4.2: Un esempio di sintesi single-pulse.

Chapter 5

La Quantizzazione

L'analisi di predizione lineare e l'algoritmo di generazione dell'eccitazione ottima (nei casi P ed NP) forniscono i parametri che il decodificatore dovrà utilizzare in fase di sintesi.

La quantizzazione, che è già di per sè un'operazione delicata, acquista per un codificatore di questo tipo un'importanza cruciale: i bit destinati alla codifica del segnale dovranno essere distribuiti con cura tra i vari parametri.

Per ogni frame da 240 campioni si hanno a disposizione 72 bit: i parametri dell'LPC possono essere quantizzati come vedremo con $24 - 26$ bit, la stima P/NP con altri 3 bit, rimangono quindi per la quantizzazione dell'eccitazione $14 - 15$ bit per subframe.

Di seguito si descriveranno il quantizzatore per l'LPC e le proposte per la quantizzazione dell'eccitazione. Sono in corso prove per verificare la validità del quantizzatore per le posizioni degli impulsi per una subframe P.

5.1 I parametri del filtro

5.1.1 Le 2dDLSF

Le Line Spectral Frequencies (descritte nell'Appendice A) sono una rappresentazione dei parametri del filtro LPC particolarmente semplice da interpolare e da quantizzare. Esse consistono in una sequenza valori ordinati tra loro e compresi tra 0 e π; tali valori si addensano in prossimità dei picchi dello spettro del segnale ed è immediato pensare che a varazioni lente dello spettro (quali sono quelle con le quali abbiamo a che fare), corrispondano variazioni analoghe nelle LSF.

Il range limitato di valori e l'ordinamento possono essere inoltre sfruttate per controllare facilmente la stabilità del filtro nella fase di quantizzazione. La quantizzazione delle LSF viene generalmente affrontata in letteratura con un metodo differenziale, vengono cioè quantizzate le differenze tra due LSF contigue nella stessa frame; solo di recente è stato proposto un metodo più sofisticato [2] che consente di ottenere risultati migliori sfruttando non solo la dipendenza intraframe, ma anche quella interframe.

Questo metodo, detto codifica differenziale bidimensionale delle LSF (in breve 2dDLSF), può essere così sintetizzato:

Si considerino $f_0(n), f_1(n) \dots f_p(n)$ le $p+1$ LSF per la frame n (dove p è l'ordine dell'analisi LPC, nel nostro caso 10), e sia inoltre $f_0(n) = 0$ per qualsiasi n. Esse risultano avere la proprietà di ordinamento :

$$0 = f_0(n) < f_1(n) < f_2(n) < \ldots < f_p(n) < \pi \tag{5.1}$$

Allo scopo di sfruttare completamente la ridondanza presente in questi parametri, si esprima la i-ma LSF quantizzata della frame n come dipendente da $f_{i-1}(n)$ e da $f_i(n-1)$ nel seguente modo :

$$\widetilde{f}_i(n) = a_i \cdot \widetilde{f}_{i-1}(n) + b_i \cdot \widetilde{f}_i(n-1), \quad i = 1, 2, \ldots, 10 \tag{5.2}$$

dove a_i e b_i sono i coefficienti di predizione (fissi) ottenuti minimizzando l'erore quadratico medio di predizione su una sequenza di voce rappresentativa ed abbastanza lunga

$$E_i = \sum_{n=1}^{N_f} [r(n)]^2 = \sum_{n=1}^{N_f} [f_i(n) - \widetilde{f}_i(n)]^2 \tag{5.3}$$

dove N_f è il numero di frames dei dati di training. In particolare abbiamo

$$a_i = \frac{\mu_{i,i-1}(0)\sigma_i^2 - \mu_{i,i}(1)\mu_{i,i-1}(1)}{\sigma_i^2\sigma_{i-1}^2 - \mu_{i,i-1}^2(1)} \tag{5.4}$$

$$b_i = \frac{\mu_{i,i}(1)\sigma_{i-1}^2 - \mu_{i,i-1}(0)\mu_{i,i-1}(1)}{\sigma_i^2\sigma_{i-1}^2 - \mu_{i,i-1}^2(1)} \tag{5.5}$$

dove

$$\begin{cases} \sigma_i^2 & = \sum_{n=1}^{N_f} f_i^2(n) \\ \mu_{i,j}(k) & = \sum_{n=1}^{N_f} f_i(n-k) f_j(n) \end{cases} \tag{5.6}$$

Ovviamente per motivi di calcolo, nella stima degli a_i e b_i viene utilizzato il dato non quantizzato. Nella fase di codifica viene quantizzato e trasmesso il residuo o errore di predizione, cioè

$$r_i = f_i - \widetilde{f}_i \tag{5.7}$$

che risulta essere più piccolo e meno sensibile agli errori di quantizzazione.

5.1.2 Il Trellis Coded Quantizer

Il vettore $\mathbf{r} = [r_0, r_2, \ldots, r_9]$, ottenuto come codifica bidimensionale differenziale delle LSF, viene quantizzato facendo uso di un quantizzatore a trellis per ottenere $\tilde{\mathbf{r}}$, il valore effettivamente utilizzato dal decodificatore.

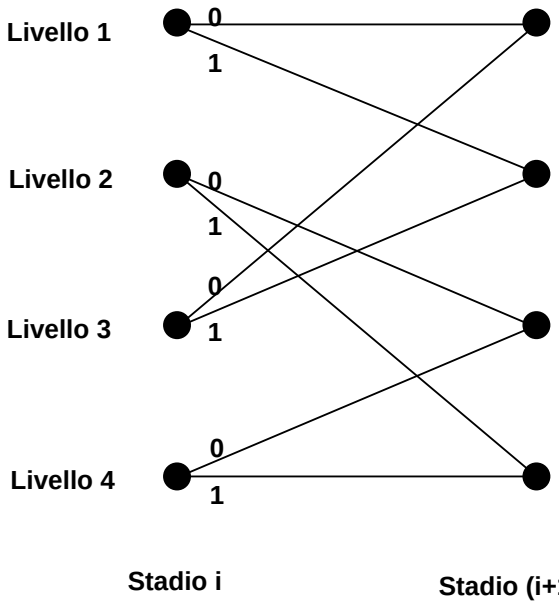

Figure 5.1: Collegamento tra due stadi successivi di un Trellis Code Quantizer.

Analogamente a quanto avviene nel caso della modulazione dei segnali con il metodo della Trellis Coded Modulation, è possibile espandere il dominio dei livelli di quantizzazione mantenendo costante il numero di bit vincolando la fase di codifica ad un trellis.

Il trellis utilizzato avrà nel caso in esame, 11 stadi numerati da 0 a 10; il numero l di nodi dello stadio i-esimo è pari al numero di livelli con i quali si vuole quantizzare l'i-mo elemento del vettore $\mathbf{r}$.

Ciascuno stadio sarà collegato al successivo con $\frac{l}{2}$ connessioni (Figura 5.1), cosicchè il valore quantizzato di r_i, può essere specificato indicando con $\log_2 \frac{i}{2} = \log_2 l - 1$ bit il cammino che connette sul trellis i livelli appartenenti a due stadi successivi.

Questo metodo consente nel nostro caso un risparmio di un bit per elemento quantizzato con un modesto sforzo computazionale; infatti la scelta del cammino sul trellis viene effettuata con l'algoritmo di Viterbi minimizzando l'errore quadratico medio (MSE) tra il vettore originale e quello quantizzato. La scelta effettuata per la metrica consente di ottenere una buona qualità mantenendo l'errore di quantizzazione entro livelli accettabili; è un problema aperto la definizione di metriche più significative da un punto di vista percettivo.

La progettazione dei livelli di quantizzazione è stata effettuata con due modalità differenti facendo riferimento ad un insieme di files costituito da frasi dette da diversi parlatori in più lingue europee (Appendice C).

Inizialmente il calcolo dei livelli è stato effettuato facendo uso dell'algoritmo Linde-Buzo-Gray [3] applicato come segue:

- si crea il Training Set per la LSF $i - ma$ utilizzando le precedenti $(i-1)$ LSF quantizzate;
- si applica l'algoritmo LBG (si veda l'Appendice B) per determinare i livelli del quantizzatore scalare;
- si itera il procedimento per la successiva LSF;
- si utilizzano i livelli trovati per la definizione del TCQ.

Il TCQ ottenuto con questo criterio, pur fornendo buone prestazioni, non risulta utilizzare i livelli in maniera equiprobabile; è evidente quindi come parte dell'informazione nel valore quantizzato fosse ridondante.

Il motivo di ciò va ricercato nel fatto che l'algoritmo LBG minimizza unicamente l'errore quadratico medio. Se il numero di bit a disposizione fosse stato maggiore, si sarebbe potuta applicare una codifica entropica al risultato della quantizzazione e progettare, a parità di bit, con un numero maggiore di livelli.

Partendo da questi presupposti, si è tentata una progettazione alternativa dei livelli di quantizzazione operando come segue:

- si crea il Training Set per la LSF $i - esima$ utilizzando le precedenti $(i-1)$ LSF quantizzate;
- il Training Set viene ordinato per determinare una serie di intervalli equiprobabili;
- in ciascuno di questi intervalli si determina un centroide che minimizza l'errore quadratico medio;
- si utilizzano i centroidi trovati per quantizzare la LSF $i - esima$ e si itera il procedimento sulla successiva.
- i livelli trovati definiscono il TCQ.

Così facendo, si lascia al trellis il compito di minimizzare l'errore globale (sulle dieci LSF) consentendogli maggiore flessibilità con i livelli equiprobabili.

I risultati ottenuti con questo metodo, misurati facendo uso della "distanza cepstrale", consentono di ottenere dei valori medi più bassi pur presentando, come è comprensibile, una varianza maggiore.

Sono state effettuate prove quantizzando i dieci parametri del filtro con 24 e 26 bit e confrontando i risultati per scegliere tra le due modalità di progetto. Le prove di ascolto hanno indicato la scelta del quantizzatore a 26 bit progettato con i livelli equiprobabili come un ragionevole compromesso tra qualità ed omogeneità del segnale ricostruito e numero di bit impiegati.

Le Figure 5.2 ed 5.3, mostrano l'istogramma delle distanze cepstrali ottenute dal quantizzatore dentro e fuori del Training Set, allocando i 26 bit per le dieci LSF nel seguente modo:

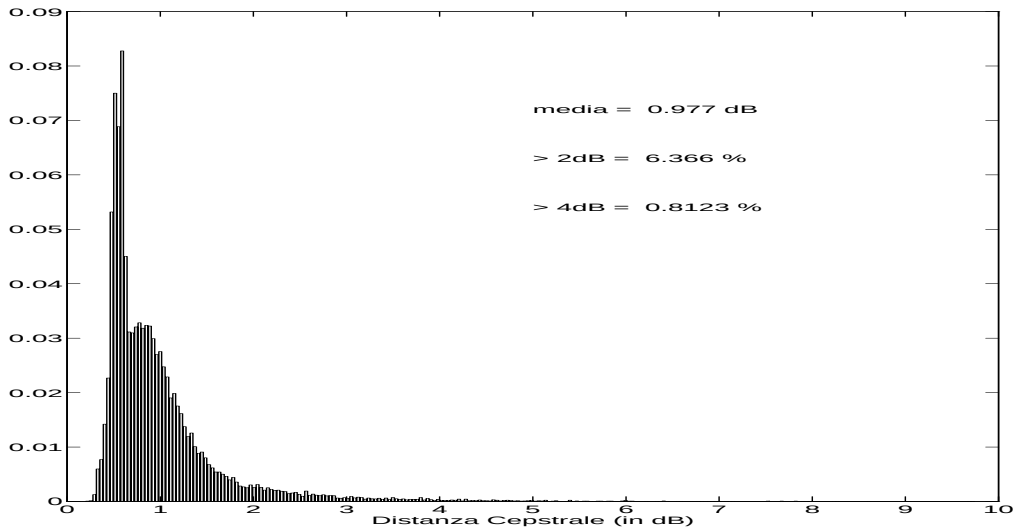

Figure 5.2: Prestazioni del quantizzatore sul Training Set.

- 4 bit per lo stato iniziale del trellis;
- 3 bit per le prime due LSF (percettivamente più significative);
- 2 bit per ciascuna delle rimanento otto LSF.

L'allocazione provata per il 24 bit è stata invece:

- 4 bit per lo stato iniziale del trellis;
- 2 bit per ciascuna delle dieci LSF.

I risultati sui files utilizzati per le prove di ascolto, sono riportati in Figura 5.4. Per una discussione sui files utilizzati si veda l'Appendice C.

La struttura del trellis utilizzato è stata per ora quella del trellis per la Trellis Coded Modulation, statistiche effettuate sull'utilizzo dei livelli sembrano suggerire che le prestazioni possano essere leggermente migliorate progettando una struttura *ad hoc* differente per ogni LSF.

5.2 L'eccitazione

5.2.1 Il caso periodico

Nell'eccitazione generata per le subframe di 80 campioni stimate periodiche sono presenti impulsi spaziati tra loro circa di un periodo di pitch. La quantizzazione delle posizioni di

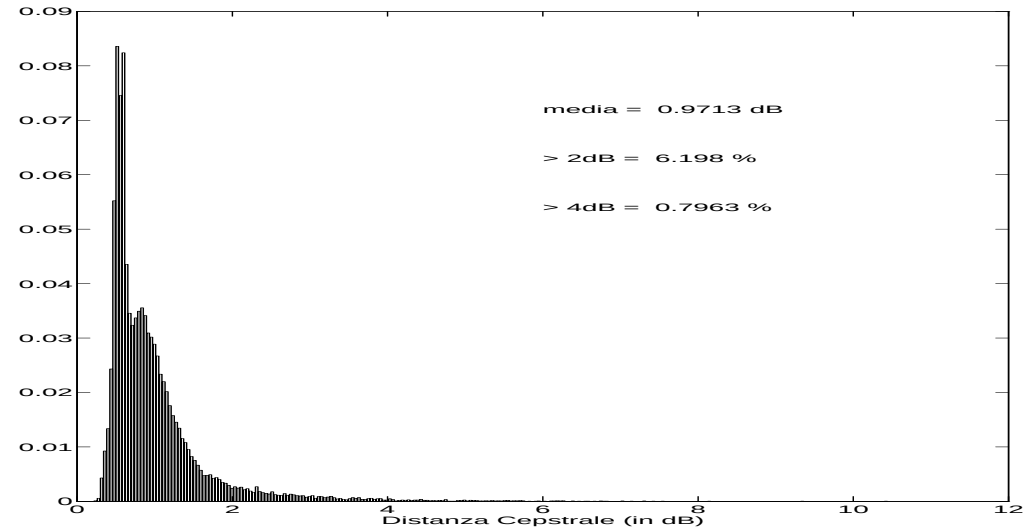

Figure 5.3: Prestazioni del quantizzatore sul Test Set.

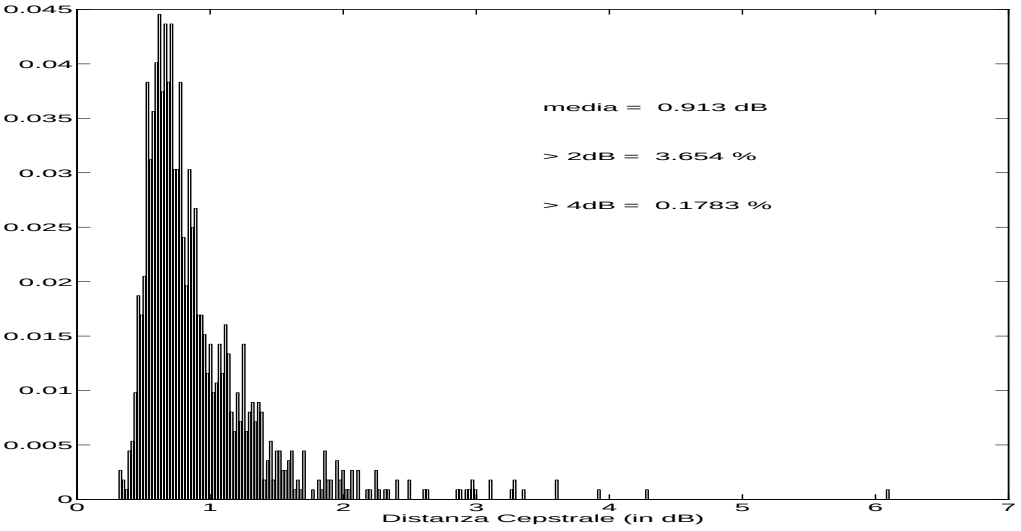

Figure 5.4: Prestazioni del quantizzatore sui files per le prove di ascolto.

questi impulsi può essere effettuata tenendo presente alcune semplici considerazioni:

- essendo il minimo periodo di pitch di 20 campioni, il numero di impulsi per subframe può variare tra zero e quattro;
- quando il numero di impulsi è due, il primo impulso può trovarsi in qualsiasi posizione compresa tra 0 e 69, ma è elevata la probabilità che si trovi in posizioni tra 0 e 39 (caso stazionario);
- un discorso analogo può farsi per la distanza del secondo impulso (quando ce ne siano due) dalla fine della subframe;
- quando nella subframe sono presenti tre impulsi, la posizione del primo (o la distanza dell'ultimo dalla fine della subframe) può variare tra 0 e 39, ma i valori più probabili sono compresi tra 0 e 27;
- nel caso siano presenti quattro impulsi la posizione del primo (o la distanza dell'ultimo dalla fine della subframe) può invece variare tra 0 e 20.

Partendo da queste semplici considerazioni basate solo sui valori che le posizioni degli impulsi possono assumere nella frame, possiamo pensare di quantizzare le posizioni con 10 bit nel seguente modo:

- due bit vengono utilizzati per specificare il numero di impulsi presenti nella subframe;
-
  - se il numero di impulsi è zero, allora non necessita codificare nient'altro;
  - se il numero di impulsi è uno, allora con gli 8 bit che rimangono si può esprimere in modo esatto la sua posizione nella frame;
  - se il numero di impulsi è due, si può codificare la posizione del primo con 4 bit e la distanza del secondo dalla fine della subframe con altri 4 bit facendo uso della tabella 5.1, costruita tenendo presenti le considerazioni fatte: andrà ricercato nella tabella il valore che meglio approssima la posizione da codificare;
  - se il numero di impulsi è invece uguale a tre, si può codificare la posizione del primo con 4 bit, la distanza dell'ultimo dalla fine della frame con altri 4 bit usando la tabella 5.2 ed in fase di sintesi, posizionare il secondo impulso nella posizione media tra i due.
  - la tabella 5.3 poò essere utilizzata per codificare la posizione del primo impulso con 4 bit, la distanza dell'ultimo dalla fine della frame con altri 4 bit nel caso di quattro impulsi per subframe; in fase di sintesi, la posizione degli impulsi intermedi è determinata interpolando.

Posizioni quantizzate nel caso di due impulsi per subframe															
0	3	6	9	12	15	18	21	24	27	30	34	38	42	51	60

Table 5.1: Livelli del quantizzatore per una coppia di impulsi per subframe.

Posizioni quantizzate nel caso di tre impulsi per subframe															
0	2	5	7	10	12	15	17	20	22	25	27	30	32	36	40

Table 5.2: Livelli del quantizzatore per tre impulsi per subframe.

I rimanenti $4-5$ bit per subframe, possono essere impiegati per quantizzare l'ampiezza dell'ultimo impulso nella subframe. Le ampiezze degli eventuali altri impulsi possono essere interpolate da questa e dall'ampiezza dell'impulso della frame precedentemente codificata.

5.2.2 Il caso non periodico

Nel caso di subframe non periodiche, si è visto come l'eccitazione ottima venga determinata attraverso una procedura di tipo CELP.

Un codebook contenente tratti di rumore gaussiano a spettro bianco viene ispezionato per la ricerca del vettore che minimizza l'errore tra il segnale sintetico e quello originale. È evidente come in questo caso l'indice stesso del vettore fornisca direttamente un valore quantizzato dell'eccitazione. Essendo il codebook utilizzato composto di 256 vettori, sono sufficienti 8 bit per la codifica dell'indice.

Il guadagno con il quale moltiplicare il vettore (si ricorda che i valori nel codebook sono normalizzati), può essere quantizzato senza perdita sostanziale di qualità con $4-5$ bit, progettando il quantizzatore con l'algoritmo LBG.

Si osserva che in questo caso non è stato necessario ricorrere ai $14-15$ bit previsti; potrebbe essere conveniente utilizzare una parte di quelli risparmiati per ampliare il codebook a 512 o addirittura a 1024 vettori.

Un'alternativa potrebbe essere quella di suddividere la subframe di 80 campioni in due parti di 40 campioni ciascuna e determinare separatamente eccitazione e guadagno; quest'ultimo potrebbe, ad esempio, essere codificato in maniera differenziale.

Posizioni quantizzate nel caso di quattro impulsi per subframe														
0	1	2	3	4	5	6	7	8	9	10	12	14	16	18	20

Table 5.3: Livelli del quantizzatore per quattro impulsi per subframe.

Chapter 6

I Risultati e gli Sviluppi Futuri

La struttura del codec proposto può essere considerata definitiva, le prove effettuate hanno confermato come la scelta ad anello chiuso tra due tipi differenti di eccitazione, sia senza dubbio una soluzione interessante. La codifica a bit-rate basso non può che trarre beneficio da una distinzione tra tratti di segnale periodico e non periodico, inoltre per segnali periodici è essenziale, per mantenere intellegibilità e naturalezza, preservare il pitch del parlatore.

Rimane da verificare la proposta di quantizzazione delle posizioni degli impulsi. La routine che ne implementa lo schema, pur essendo stata scritta e provata singolarmente, deve ancora essere inserita nel codificatore.

Non c'è motivo di credere che le prestazioni debbano degradarsi in maniera significativa, tuttavia le prove comparative non sono ancora state portate a termine e si stima richiederanno almeno una settimana.

Inutile sottolineare che numerosi dei problemi affrontati in questo lavoro sono da considerarsi completamente aperti; problemi come la stima del pitch, la decisione P/NP, la ricerca di misure percettivamente significative con le quali guidare la fase di codifica, sembrano attualmente lungi dall'avere una soluzione definitiva.

I fenomeni connessi alla generazione ed alla percezione della voce sono solo in minima parte noti e quando di rado questo avviene, non è cosa semplice formalizzarli ed esprimereli quantitativamente in modo da poterli efficacemente sfruttare.

Le soluzioni che abbiamo individuato attraverso questo studio devono solo, e non può essere altrimenti, essere considerate euristiche. Riguardo alla qualità della voce sintetizzata, bisogna sottolineare che a bit-rate così bassi non ha senso parlare di misure oggettive; tutte le metriche di qualità al momento note, forniscono risultati assolutamente non affidabili o privi di significato; anche in questo caso è doveroso notare come il problema sia connesso sopratutto ad una scarsa conoscenza dei fenomeni percettivi.

La qualità del segnale ricostruito è stata valutata, durante tutto il lavoro, attraverso l'attento ascolto di un insieme di frasi ritenute critiche per il comportamento di un codificatore ed utilizzate come riferimento praticamente da tutti i laboratori americani (Ap-

pendice C). Queste frasi sono state ascoltate in cuffia, in una serie di prove informali, allo scopo di identificare, per confronto con gli originali, difetti di codifica e rumori presenti.

Determinante è stato l'uso di codificatori standard già sviluppati, utilizzati come termini di paragone.

I risultati ottenuti confrontando il codec realizzato con lo standard LPC10E confermano senza dubbio la validità dell'approccio seguito. Nonostante la qualità del segnale ricostruito risulti essere abbastanza dipendente dal parlatore esso conserva il timbro e la naturalezza della voce originale.

Sono ancora presenti fastidiose discontinuità che andrebbero eliminate raffinando ulteriormente l'algoritmo; nei tratti vocalizzati inoltre, si percepisce un rumore periodico dovuto probabilmente alla povertà dell'eccitazione periodica. Numerose sono le proposte di miglioramento, esse toccano in modo più o meno uniforme tutte le parti del codec.

- Potrebbe essere conveniente quantizzare i parametri in gioco in modo meno preciso ed utilizzare i bit risparmiati per arricchire l'eccitazione "single-pulse" (una possibilità potrebbe essere quella di usare un "impulso glottale" o una coppia di impulsi).

  Sono state effettuate prove mirate ad eliminare la componente continua presente nel segnale sintetico a causa dell'eccitazione esclusivamente positiva; da un punto di vista percettivo non si è notata alcuna differenza, tuttavia questa procedura non è escluso possa rivelarsi utile nel controllare la precisione in una eventuale implementazione del codec in virgola fissa.

- Le prestazioni del quantizzatore a trellis potrebbero essere migliorate, o potrebbe essere ridotto il numero dei bit, progettando la struttura delle connessioni *ad hoc* per ogni LSF. Le statistiche di utilizzo delle connessioni sono state già effettuate, tuttavia si prevede che la definizione di un trellis di questo tipo richieda un grosso impegno di calcolo e la stesura di un algoritmo opportuno.

- La decisione P/NP viene ora presa utilizzando la funzione di autocorrelazione calcolata sul segnale; l'uso combinato con quello di altri indicatori (zero crossing, energia, ecc.) potrebbe fornire una stima più affidabile.

- Sarebbe opportuno interpolare i parametri LPC il più frequentemente possibile, al limite anche ad ogni periodo di pitch; questo potrebbe contribuire ad aumentare l'uniformità del ricostruito eliminando alcune delle discontinuità presenti.

Appendix A

Le LSF

Si consideri la funzione di trasferimento del filtro di analisi $A(z)$

$$A(z) = \sum_{k=0}^{P} a_k z^{-k} \tag{A.1}$$

A partire da questa si definiscono le seguenti funzioni della variabile z

$$F_1(z) = A(z) + z^{P+1} A(z^{-1}) \tag{A.2}$$

$$F_2(z) = A(z) - z^{P+1} A(z^{-1}) \tag{A.3}$$

Come si vede esse non sono altro che le funzioni di trasferimento di due filtri di ordine $P+1$ i cui primi P coefficienti di riflessione coincidono con quelli di A(z) mentre l' ultimo risulta pari a +1 e -1 rispettivamente. Dalla definizione, segue che $F_1(z)$ è il polinomio di grado $P+1$ nella variabile z^{-1}, simmetrico, del tipo seguente

$$F_1(z) = \begin{cases} 1 + f_{1,1}z^{-1} + \ldots + f_{1,q_1}z^{-q_1} + f_{1,q_1}z^{-(q_1+1)} + \ldots + \\ \qquad + f_{1,1}z^{-P} + z^{-(P+1)} & \text{P pari} \\ 1 + f_{1,1}z^{-1} + \ldots + f_{1,q_1-1}z^{-(q_1-1)} + f_{1,q_1}z^{-q_1} \\ \quad + f_{1,q_1-1}z^{-(q_1+1)} + \ldots + + f_{1,1}z^{-P} + z^{-(P+1)} & \text{P dispari} \end{cases} \tag{A.4}$$

dove si è posto

$$q_1 = \begin{cases} \frac{P}{2} & \text{P pari} \\ \frac{P+1}{2} & \text{P dispari} \end{cases} \tag{A.5}$$

e dove i coefficienti $f_{1,1}, \ldots, f_{1,q_1}$ hanno la seguente espressione

$$f_{1,i} = a_i + a_{P+1-i} \qquad 1 \leq i \leq q_1 \tag{A.6}$$

Analogamente dalla definizione (A.3) segue che $F_2(z)$ è un polinomio di grado P+1 nella variabile z^{-1}, antimmetrico, del tipo seguente

$$F_2(z) = \begin{cases} 1 + f_{2,1}z^{-1} + \ldots + f_{2,q_2}z^{-q_2} - f_{2,q_2}z^{-(q_2+1)} + \ldots + \\ \qquad - f_{2,1}z^{-P} - z^{(P+1)} & \text{P pari} \\ 1 + f_{2,1}z^{-1} + \ldots + f_{2,q_2-1}z^{-(q_2-1)} + f_{2,q_2}z^{-q_2} \\ \quad - f_{2,q_2-1}z^{-(q_2+1)} - \ldots - f_{2,1}z^{-P} - z^{(P+1)} & \text{P dispari} \end{cases} \tag{A.7}$$

con

$$q_2 = \begin{cases} \frac{P}{2} & \text{P pari} \\ \frac{P-1}{2} & \text{P dispari} \end{cases} \tag{A.8}$$

I coefficienti $f_{2,1}, \ldots, f_{2,q_2}$ risultano pertanto

$$f_{2,i} = a_i - a_{P+1-i} \qquad 1 \leq i \leq q_2 \tag{A.9}$$

Questa coppia di polinomi gode della seguenti proprietà:

- il polinomio $A(z)$ è a fase minima se e solo se gli zeri di $F_1(z)$ e $F_2(z)$ sono semplici, con modulo unitario e sono disposti alternati sul cerchio unitario.
- se P è pari, $F_1(z)$ ha uno zero in -1, mentre $F_2(z)$ ha uno zero in 1 ; se invece P è dispari $F_1(z)$ non ha zeri in +1 e -1, mentre $F_2(z)$ ha uno zero in -1 e uno in +1.

Poichè l' esistenza di tali zeri è indipendente del polinomio $A(z)$, essi non forniscono alcuna informazione utile. è allora conveniente definire i seguenti polinomi in z

$$G_1(z) = \begin{cases} \frac{F_1(z)}{1+z^{-1}} & \text{P pari} \\ F_1(z) & \text{P dispari} \end{cases} \tag{A.10}$$

$$G_2(z) = \begin{cases} \frac{F_2(z)}{1-z^{-1}} & \text{P pari} \\ \frac{F_2(z)}{1-z^{-2}} & \text{P dispari} \end{cases} \tag{A.11}$$

di grado rispettivamente $2q_1$ e $2q_2$ e che hanno gli stessi zeri di $F_1(z)$ e $F_2(z)$, esclusi chiaramente quelli in +1 e -1. Eseguendo la divisione nella A.10 si ottiene che $G_1(z)$ è un polinomio simmetrico del tipo seguente:

$$G_1(z) = 1 + g_{1,1}z^{-1} + \ldots + g_{1,q_1}z^{-q_1} + \ldots + g_{1,1}z^{-(2q_1-1)} + z^{-2*q_1} \tag{A.12}$$

dove se P è pari i coefficienti $g_{1,1}, \ldots, g_{1,q_1}$ sono dati dalla seguente formula ricorsiva :

$$g_{1,i} = -g_{1,i-1} + a_i + a_{P+1-i} \qquad 1 \leq i \leq q_1 \tag{A.13}$$

con la condizione iniziale

$$g_{1,0} = 1 \tag{A.14}$$

mentre se P è dispari coincidono con i coefficienti di $F_1(z)$ e quindi sono pari a:

$$g_{1,i} = a_i + a_{P+1-i} \qquad 1 \leq i \leq q_1 \tag{A.15}$$

Analogamente eseguendo la divisione in A.11 ed utilizzando la A.3 si ottiene che $G_2(z)$ è un polinomio simmetrico del tipo seguente:

$$G_2(z) = 1 + g_{2,1}z^{-1} + \ldots + g_{2,q_2}z^{-q_2} + \ldots + g_{2,1}z^{-(2q_2-1)} + z^{-2*q_1} \tag{A.16}$$

dove se P è pari i coefficienti $g_{2,1}, \ldots, g_{2,q_2}$ sono dati dalla seguente formula ricorsiva :

$$g_{2,i} = g_{2,i-1} + a_i - a_{P+1-i} \qquad 1 \leq i \leq q_2 \tag{A.17}$$

con la condizione iniziale

$$g_{2,0} = 1 \tag{A.18}$$

mentre se P è dispari i coefficienti sono dati dalla

$$g_{2,i} = g_{2,i-2} + a_i - a_{P+1-i} \qquad 1 \leq i \leq q_2 \tag{A.19}$$

con le condizioni iniziali

$$g_{2,0} = 1 \;\; g_{2,-1} = 0 \tag{A.20}$$

La prima proprietà dei polinomi $F_1(z)$ e $F_2(z)$ diventa la seguente per i polinomi $G_1(z)$ e $G_2(z)$: un polinomio $A(z)$ è a fase minima se e solo gli zeri dei polinomi $G_1(z)$ e $G_2(z)$ sono complessi coniugati, semplici e sono disposti alternati sul cerchio unitario.

Pertanto non si ha perdita di informazione considerando per ciascuna coppia di zeri coniugati solo l' argomento di quello a parte immaginaria positiva. Eseguendo il cambiamento di variabile

$$z = e^{j\theta} \tag{A.21}$$

si ottengono le seguenti funzioni :

$$\begin{aligned}\hat{G}_1(\theta) = 1 + g_{1,1}e^{-j\theta} + \ldots + g_{1,q_1}e^{-j\theta q_1} + \ldots + \\ + g_{1,1}z^{-(2q_1-1)} + e^{-j2q_1\theta}\end{aligned} \tag{A.22}$$

$$\begin{aligned}\hat{G}_2(\theta) = 1 + g_{2,1}e^{-j\theta} + \ldots + g_{2,q_2}e^{-j\theta q_2} + \ldots + \\ + g_{2,1}z^{-(2q_2-1)} + e^{-j2q_2\theta}\end{aligned} \tag{A.23}$$

Considerando solo i valori di θ appartenenti all' intervallo $[0, \pi]$ si ha che le funzioni $\hat{G}_1(\theta)$ e $\hat{G}_2(\theta)$ hanno rispettivamente q_1 e q_2 zeri. I loro zeri sono complessivamente P e, per il modo in cui tali funzioni sono state ottenute, sono ricavabili in modo univoco dai coefficienti di predizione $a_1, \ldots, a_P$ e viceversa.

Gli zeri $\theta_1, \ldots, \theta_P$ delle funzioni $\hat{G}_1(\theta)$ e $\hat{G}_2(\theta)$ sono detti LSF (Line Spectrum Frequencies). Dalle proprietà degli zeri di questi due polinomi si ricava che $A(z)$ è stabile se e solo se vale la seguente relazione d' ordine per le LSF:

$$0 < \theta_1 < \ldots < \theta_P < \pi \tag{A.24}$$

dove le pulsazioni di indice dispari sono riferite agli zeri di $\hat{G}_1(\theta)$ e quelle di ordine pari agli zeri di $\hat{G}_2(\theta)$.

Le pulsazioni relative delle LSF hanno solitamente un significato spettrale abbastanza evidente. Si ha un addensamento di una coppia di LSF in corrispondenza di ciascuna coppia di poli complessi coniugati della funzione di trasferimento del filtro di sintesi.

Dal momento che i poli complessi coniugati determinano i picchi di risonanza che approssimano le formanti del segnale vocale, si verifica un addensamento di due o più pulsazioni relative delle LSF in corrispondenza delle formanti. La posizione delle LSF rispecchia, dunque, in una certa misura, l'andamento delle formanti, la cui variazione nel tempo per il segnale di tipo vocale è lenta e regolare; ne consegue quindi una maggiore facilità di quantizzazione e di interpolazione.

Appendix B

L'algoritmo LBG

La progettazione dei livelli da usare nei vari stadi del trellis, è stata effettuata utilizzando un algoritmo di "clustering" proposto per la priva volta da Linde Buzo e Gray in [3].

Scegliere gli L livelli di un quantizzatore su un insieme di dati T equivale a partizionare l'insieme T in L celle $\{C_i, 1 \leq i \leq L\}$ ed associare ad ogni cella C_i un vettore y_i.

Il quantizzatore opera assegnando il vettore di codice y_i all'ingresso x se e solo se x appartiene a C_i.

Con questa strategia di quantizzazione, la distorsione totale sull'insieme di test T risulta pertanto:

$$D = \sum_{i=1}^{L} \sum_{x \in C_i} d(x, y_i) \tag{B.1}$$

dove $d(x, y)$ è la metrica con la quale si è deciso di calcolare le distanze.

Un quantizzatore si definisce ottimo (ovvero a minima distorsione) per l'insieme T se la distorsione calcolata come nella B.1 risulta minima tra tutti i possibili quantizzatori a L livelli.

Ci sono due condizioni necessarie perchè un quantizzatore sia ottimo. La prima è che il quantizzatore scelga la parola di codice y_i per l' ingresso x usando un criterio di minima distorsione

$$q(x) = y_i, \quad \text{se e solo se } d(x, y_i) \leq d(x, y_j) \;\; j \neq i \;\; 1 \leq j \leq L \tag{B.2}$$

La seconda condizione è che ogni parola di codice y_i sia scelta per minimizzare la distorsione media della cella C_i, ovvero che y_i sia il vettore y che minimizza

$$D_i = \sum_{x \in C_i} d(x, y) \tag{B.3}$$

Tale vettore viene detto centroide della cella C_i, e verrà indicato nel seguito come

$$y_i = cent(C_i) \tag{B.4}$$

Il valore del centroide dipenderà dalla misura scelta per la distanza.

Per la distanza euclidea è facile vedere che tale valore è dato da

$$y_i = \frac{1}{M_i} \sum_{x \in C_i} x \tag{B.5}$$

dove M_i è il numero di vettori contenuti nella cella C_i.

L' algoritmo LBG sfrutta queste due condizioni per la ricerca del codebook ottimo con un metodo iterativo.

Si suddivide l'insieme dei vettori del training set $\{x(n)\}$ in L raggruppamenti C_i in modo tale da soddisfare ad ogni passo le due condizioni necessarie. Nel seguito verrà indicato con m l'indice dell'iterazione, con $C_i(m)$ la i-esima partizione all'iterazione m e con $y_i(m)$ il suo centroide.

L'algoritmo procede nel modo seguente.

1. *Inizializzazione*: m=0 . Si sceglie secondo un qualche metodo (in questo lavoro in modo random) un insieme di vettori del codice $Y_i(0)$,$1 \leq i \leq L$.

2. *Classificazione*: i vettori del training set $x(n)$ sono raggrupati nelle celle C_i secondo la regola $x \in C_i$ se e solo se $d(x, y_i(m)) \leq d(x, y_j(m))\ \forall j \neq i$.

3. *Aggiornamento dei vettori del codebook*: m=m+1. L'aggiornamento avviene calcolando per ogni cella C_i il relativo centroide

$$y_i(m) = cent(C_i(m-1))$$

.

4. *Test per l' uscita dall' iterazione*: se il decremento della distorsione $D(m)$ rispetto a quella all' iterazione $m-1$ è sotto una certa soglia o si è superato il numero massimo di iterazioni previsto ci si ferma, diversamente si torna al passo 2.

L'algoritmo converge sempre rapidamente ad un minimo che però può non essere globale.

Appendix C

Il Training Set

Nella progettazione dei quantizzatori e nella messa a punto del codec, sono stati utilizzati una serie di files di riferimento.

La progettazione del quantizzatore a trellis per i coefficienti LSF è stata effettuata su 152 files consistenti in frasi nelle principali lingue europee pronunciate da parlatori differenti (maschili e femminili).

Il campionamento è stato effettuato ad una frequenza di 8 kHz con 16 bit per campione e metà di essi è stata filtrata con un simulatore di linea telefonica allo scopo di introdurre disturbi e rumore di fondo.

Metà di questi files (in parte filtrati con il simulatore di linea ed in parte no) è stata utilizzata per il calcolo dei livelli (Training Set), l'altra metà (composta allo stesso modo) per verificare le prestazioni del quantizzatore (Test Set). Le prestazioni sono sempre risultate estremamente simili su entrambi gli insiemi.

Le prove informali di ascolto si sono avvalse invece di una serie di files contenenti frasi pronunciate in inglese da parlatori madrelingua sia maschili che femminili.

Le frasi (della durata di circa 3 secondi) sono quelle comunemente utilizzate nei laboratori americani per le prove di ascolto:

- "Why were you away a year Roy?" (suoni vocalizzati);
- "Nanny may know my meaning" (suoni nasali);
- "Which tea-party did Baker go to?" (suoni esplosivi);
- "The little blanket lies around on the floor" (suoni esplosivi);
- "His vicious father has seizures" (suoni fricativi);
- "The problem with swimming is that you can drown" (suoni fricativi vocalizzati).

La queste frasi, per la varietà di fonemi che contengono e per la loro composizione, sono particolarmente critiche e risultano adatte alla verifica delle prestazioni di un codificatore.

Bibliography

[1] W. Granzow and B.S. Atal. High-quality digital speech at 4 kb/s. In *ICASSP'90*, pages 941–945, 1990.

[2] Chin-Chung Kuo, Fu-Rong Jean, and Hsiao-Chuan Wang. Low bit-rate quantization of lsp using two-dimensional differential coding. In *ICASSP-92*, volume 1, pages 97–100, California, 1992.

[3] Y. Linde, A. Buzo, and R.M. Gray. An algorithm for vector quantization design. *IEEE Trans. Commun.*, COM-28:84–95, January 1980.

[4] L. R. Rabiner and R. W. Schaffer. *Digital Processing of Speech Signals.* Prentice-Hall,Inc.

[5] M.R. Schroeder and B.S. Atal. Code-excited linear prediction (celp) : High-quality speech at very low bit rate. *Proc. ICASSP*, pages 937–940, March 1985.

[6] Stephanie Seneff. *Pitch and Spectral Analysis of Speech Based on an Auditory Sincrony Model.* Technical Report 504, MIT, 1985.

[7] W.Verhelst, B.Franco, and O.Steenhaut. An adaptive non-uniform sign clipping pre-processor (anusc) for real-time autocorrelative pitch detection. In *ICASSP'86*, pages 121–124, 1986.