

Przetwarzanie sygnałów w telekomunikacji
-kompresja sygnałów-

Przemysław Dymarski
Zakład Systemów Teletransmisyjnych
Instytut Telekomunikacji Politechniki Warszawskiej

2010

Spis treści

1	Wstęp	4
1.1	Kompresja bezstratna i stratna	4
1.2	Jakość sygnału poddanego kompresji	6
2	Metody kodowania różnicowego ADPCM	8
2.1	Kwantyzatory adaptacyjne	9
2.1.1	Wprowadzenie	9
2.1.2	Opis układu DPCM	9
2.1.3	Kwantyzator z adaptacją	12
2.1.4	Wyniki obliczeń i symulacji	15
2.1.5	Podsumowanie	19
2.2	Algorytmy predykcji	20
2.2.1	Wprowadzenie	20
2.2.2	Predyktor w układzie DPCM	20
2.2.3	Adaptacja predyktora metodami gradientu stochastycznego	22
2.2.4	Adaptacja predyktora metodą najmniejszych kwadratów	25
2.2.5	Wyniki badań symulacyjnych	30
2.2.6	Podsumowanie	34
3	Kwantyzatory wektorowe i kodery CELP	36
3.1	Kwantyzacja wektorowa	37
3.1.1	Zasada działania kwantyzatora wektorowego	37
3.1.2	Wartości graniczne dla kwantyzatora wektorowego	39
3.1.3	Problem złożoności obliczeniowej	41
3.2	Dekompozycja słownika kwantyzatora wektorowego	42
3.2.1	Słownik w postaci iloczynu kartezyjskiego	42
3.2.2	Kwantyzatory wektorowe typu kształt - wzmocnienie	43
3.2.3	Modelowanie wielostopniowe i predykcja długookresowa	44
3.2.4	Kodery CELP jako PCVQ	47
3.3	Projektowanie koderów typu kształt - wzmocnienie	50
3.3.1	Przeszukiwanie słowników i tworzenie modelu sygnału mowy	50
3.3.2	Wyznaczanie słowników	52
3.3.3	Porównanie wariantów SGVQ	56

3.4	Projektowanie koderów z modelowaniem wielostopniowym	60
3.4.1	Wprowadzenie	60
3.4.2	Przeszukiwanie słowników i tworzenie modelu sygnału mowy - sformu- łowanie zadania	61
3.4.3	Lokalnie optymalne algorytmy modelowania wielostopniowego	62
3.4.4	Modelowanie wielostopniowe z pojedynczym wzmocnieniem	70
3.4.5	Projektowanie słowników	74
3.4.6	Porównanie algorytmów modelowania	76
3.4.7	Przykład realizacji praktycznej	81
4	Kodowanie transformaty z wykorzystaniem kwantyzacji wektorowej	86
4.1	Wstęp	87
4.2	Wyznaczanie zysku transformaty	87
4.2.1	Analiza problemu w postaci ogólnej	88
4.2.2	Obliczanie zysku transformat stosowanych w praktyce	92
4.2.3	Transformaty "z nakładaniem"	94
4.2.4	Zestaw filtrów o zmiennej rozdzielczości	97
4.2.5	Transformata optymalna	100
4.3	Kodowanie w dziedzinie częstotliwości	104
4.3.1	Wykorzystanie zestawu filtrów o zmiennej rozdzielczości	104
4.3.2	Predykcyjne kodery transformaty	106
4.3.3	Koder transformaty o małym opóźnieniu	109
4.3.4	Koder o zmiennej przepływności binarnej	114
4.3.5	Podsumowanie	118
5	Parametryczne kodowanie mowy	119
5.1	Kodery parametryczne i kodery fali akustycznej	120
5.2	Metody dyskryminacji liniowej i ich zastosowanie do ekstrakcji tonu krtaniowego	121
5.2.1	Dyskryminatory	121
5.2.2	Rozszerzony ekstraktor tonu krtaniowego	123
5.3	Zmodyfikowany wokoder predykcyjny o przepływności 2.4 kbit/s	131
5.3.1	Ogólna charakterystyka wokodera	131
5.3.2	Nadajnik wokodera predykcyjnego 2.4 kbit/s	131
5.3.3	Odbiornik wokodera predykcyjnego	133
5.4	Wokoder 1.2 kbit/s	136
5.4.1	Wokoder radiowy	136
5.4.2	Interpolacja i kwantyzacja transmitowanych parametrów	137
5.4.3	Realizacja i testowanie wokodera	138
5.5	Zastosowanie ekstraktora tonu krtaniowego w koderze CELP	138
A	Oznaczenia	141
B	Ważniejsze skróty	147

C	Przegląd standardów kompresji	151
C.1	Standardy dla sygnałów telefonicznych	151
C.2	Standardy dla sygnałów akustycznych	154

Rozdział 1

Wstęp

1.1 Kompresja bezstratna i stratna

W drugiej połowie ub. stulecia prowadzono intensywne prace nad kompresją mowy. W początkowym okresie miały one za cel lepsze wykorzystanie ówczesnych telefonicznych sieci kablowych. Czy oznacza to, że obecny rozwój szerokopasmowych połączeń światłowodowych pozbawi znaczenia metody kompresji sygnałów? Nie jest to prawdą, gdyż pojawiły się nowe zastosowania, związane z komunikacją radiową i bezprzewodowym dostępem do sieci: telefonia komórkowa, bezpośrednia łączność satelitarna między obiektami ruchomymi, cyfrowa komunikacja radiowa, radiofonia cyfrowa, itp. Podobnie w przenośnych urządzeniach do odtwarzania dźwięku, oszczędność nośnika wymaga stosowania algorytmów kompresji sygnałów fonicznych (np. w DCC¹, Mini-Dyskach, itp.). Przegląd ważniejszych standardów kompresji mowy i sygnałów fonicznych zamieszczono w Dodatku C.

Wyróżnia się dwie grupy metod kompresji: metody kompresji **bezstratnej** (bez utraty informacji, np. kod Huffmana [47]) i **stratnej** (z utratą części informacji, co nie musi oznaczać zauważalnego pogorszenia się jakości dźwięku). Niniejsze opracowanie dotyczy wyłącznie metod kompresji stratnej (sensu stricto wszelkie algorytmy cyfrowego przetwarzania sygnałów analogowych mają charakter stratny, ze względu na nieodwracalność operacji kwantowania w przetworniku a/c).

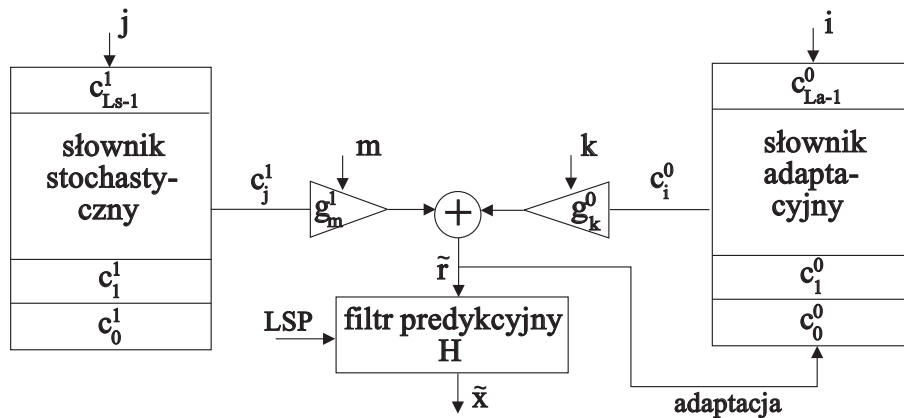
Podziału metod kompresji stratnej można dokonywać wg różnych kryteriów, np. można wyróżnić **kodery fali akustycznej** i **kodery parametryczne**. Te pierwsze dążą do odtworzenia przebiegu czasowego sygnału, te drugie generują sztuczny sygnał o brzmieniu podobnym do oryginału, jednak o innym przebiegu czasowym. Kodery parametryczne (wo-kodery) zapewniają uzyskanie mniejszej przepływności binarnej - od kilkuset do kilku tysięcy bitów na sekundę.

Ze względu na zasadę działania, można wyróżnić następujące grupy koderów mowy i sygnałów akustycznych:

Kwantyzatory skalarne

Kwantyzatory wektorowe

¹Słowniczek ważniejszych skrótów zamieszczono w Dodatku B



Rysunek 1.1: Uproszczony schemat odbiornika (syntezera) CELP

Kodery predykcyjne, w tym

- Kodery różnicowe
- Kodery CELP

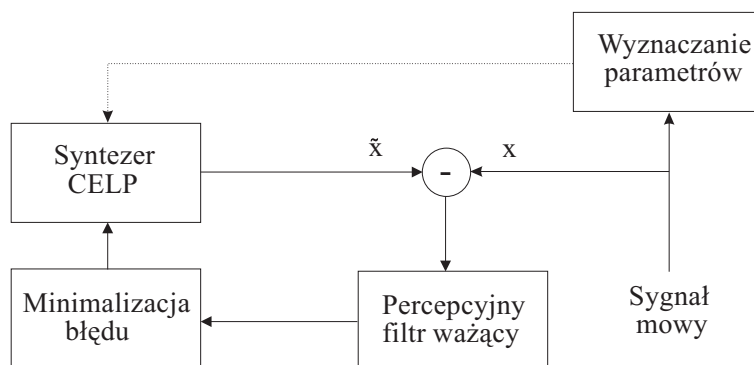
Koder predykcyjno - wektorowy (znany pod nazwą CELP - *Code Excited Linear Prediction*, a także VXC - *Vector Excitation Coding* lub SELP - *Stochastically Excited Linear Prediction*) łączy w sobie dwie niezwykle skuteczne techniki kodowania sygnałów: kwantyzację wektorową i predykcję liniową.

Schemat odbiornika CELP (Rys.1.1) zawiera predyktor długookresowy (pod nazwą słownika adaptacyjnego) oraz słownik kształtów sygnału (pod nazwą słownika stochastycznego). Wektory z obu słowników są poddawane wzmocnieniu (jak w SGVQ), w celu dopasowania ich poziomu do poziomu sygnału. Filtr predykcyjny $H(z)$ jest głównym elementem odróżniającym koder CELP od kwantyzatora wektorowego. Nadaje on sygnałowi odpowiedni skład widmowy, w związku z tym w słowniku kształtów nie trzeba magazynować wzorców sygnałów o różnym widmie. Są tam tylko fragmenty sygnału o płaskim widmie (np. szum). Jest wówczas możliwe zmniejszenie liczby wektorów w słowniku przy bardzo dobrym dopasowaniu koder do zmian poziomu i widma przetwarzanego sygnału. Nadajnik CELP działa na zasadzie analizy przez syntezę (Rys.1.2). Wybierane są parametry (numery wektorów, wzmocnienia), zapewniające najmniejszą odległość wektora syntezowanej mowy od wektora mowy rzeczywistej.

- Wokodery predykcyjne

Kodery operujące w dziedzinie transformaty

Niepredykcyjne kodery parametryczne, itp.



Rysunek 1.2: Uproszczony schemat nadajnika CELP

1.2 Jakość sygnału poddanego kompresji

Algorytm kompresji powinien przede wszystkim spełnić założenia co do **stopnia kompresji**, tzn. zapewnić pożądaną przepływność binarną. Ta ostatnia zależy od przewidywanego zastosowania - najmniejsze przepływności, rzędu 1-2 kbit/s są wymagane jedynie do komunikacji specjalnej, np. cyfrowe radiostacje krótkofalowe.

Podstawowym kryterium oceny algorytmu kompresji jest **jakość sygnału po kompresji**. Jakość sygnału bada się metodami subiektywnymi i obiektywnymi. Metody subiektywne polegają na przeprowadzeniu odsłuchów. ITU-T opublikowało zalecenia dotyczące przeprowadzania odsłuchów i opracowania ich wyników: w [95] opublikowano zalecenia dotyczące badania jakości sygnału telefonicznego, a w [98] - sygnału akustycznego o poszerzonym paśmie. Najczęściej przeprowadza się test słuchowy, stosując bezwzględną ocenę jakości w skali od 1 do 5 - tzw. metoda ACR. Niekiedy ocenia się pogorszenie jakości w stosunku do frazy oryginalnej, również w skali od 1 do 5 - metoda DCR. Metodą CCR porównuje się ze sobą dwie frazy, wyrażając ocenę w skali od (-3) do 3. Do metod porównawczych należy również MNRU [97]. W metodzie tej jakość sygnału wyraża się wartością SNR w układzie odniesienia, generującym sygnał o tej samej jakości, co sygnał badany. Jakość ocenia się metodą słuchową, przy czym warunki przeprowadzenia odsłuchów zostały opisane w normie [97].

W Polsce obowiązuje norma PN-90/T-05100, odnosząca się do oceny jakości mowy metodą pomiaru wyrazistości logatomowej [163]. W badaniu tym określa się stosunek procentowy poprawnie rozpoznanych logatomów² do całkowitej liczby logatomów zaprezentowanych słuchaczom.

Do obiektywnych metod oceny jakości sygnału należy SNR - stosunek mocy sygnału akustycznego do mocy zakłóceń (szumu kwantyzacji, itd.). Ze względu na niestacjonarność sygnału mowy, lepszym kryterium jest SNR w ujęciu segmentowym: SNR_{seg} . Jest to wartość średnia wyrażonych w decybelach wartości $SNR(i)$, określonych w obrębie segmentów o

²Logatomy to krótkie (1-3 sylabowe) wyrazy pozbawione znaczenia semantycznego. Rozpoznanie logatomu jest wyłącznie wynikiem usłyszenia wchodzących w jego skład fonemów

czasie trwania 10-30 ms (co odpowiada czasowi trwania krótkich głosek):

$$SNR_{seg} = \frac{1}{M} \sum_{i=1}^M SNR(i) \text{ [dB]} \quad (1.1)$$

Nietrudno zauważyć, że uszkodzenie "cichych" głosek (zastąpienie szumem lub sygnałem o wartości zerowej) spowoduje spadek SNR_{seg} , podczas gdy SNR może pozostać niemal bez zmian.

SNR , SNR_{seg} to kryteria "robocze", chętnie wykorzystywane w procesie projektowania urządzenia do kompresji sygnału (sprawdzanie poprawności algorytmu, porównywanie jego wariantów). Stosując te kryteria, należy pamiętać, że:

- Wartość SNR , SNR_{seg} zależy nie tylko od kodera, ale i od sygnału. Np. kodery predykcyjne bardzo dobrze przetwarzają sygnał harmoniczny, a znacznie gorzej- sygnał mowy. Nie należy w takich przypadkach prowadzić badań symulacyjnych z wykorzystaniem sygnału sinusoidalnego (chyba, że chodzi jedynie o stwierdzenie poprawności algorytmu)
- Dla koderów parametrycznych (wokoderów o małej przepływności binarnej) badanie SNR i SNR_{seg} nie ma sensu, ze względu na to, że w koderach tego typu nie dąży się do odtworzenia przebiegu czasowego sygnału (inny kształt fali akustycznej, przesunięcia fazowe, itp.)
- Wszelkiego typu pre- i postfiltry, mające na celu poprawę brzmienia sygnału, obniżają wartość SNR i SNR_{seg} , w trakcie badania powinny być wyłączone.
- Po wykonaniu modelu urządzenia kompresji, należy przeprowadzić badania jakości zgodnie z obowiązującą normą (w Polsce są to badania wyrazistości logatomowej).

Należy zaznaczyć, że ITU-T opracowała testy obiektywnej oceny jakości mowy [96], nadające się do zastosowania również w tych przypadkach, gdy SNR zawodzi - patrz też [85], [39], [204].

Rozdział 2

Metody kodowania różnicowego ADPCM

2.1 Kwantyzatory adaptacyjne

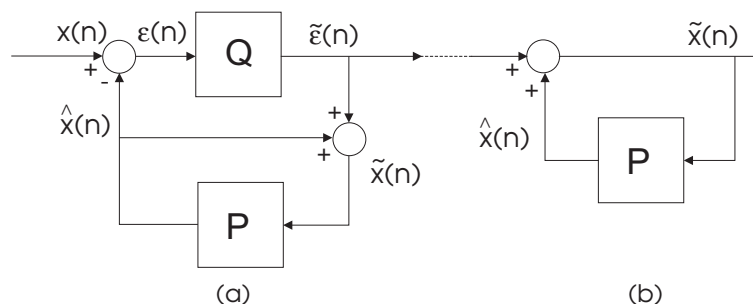
2.1.1 Wprowadzenie

W kodowaniu i transmisji cyfrowej sygnału mowy początkowo dominowały systemy oparte na próbkowaniu i kwantyzacji skalarnej (modulacja PCM). Od lat 80-tych ub. wieku coraz większe znaczenie zyskują systemy oparte na próbkowaniu, kwantyzacji i predykcji (modulacja DPCM), umożliwiające zmniejszenie przepływności binarnej przy zachowaniu dobrej jakości sygnału mowy. Przykładem jest system ADPCM dla transmisji sygnału telefonicznego kanałem o przepływności 32 kbit/s [18]. W systemach DPCM stosuje się kwantyzatory adaptacyjne, równomierne i nierównomierne. Algorytm projektowania kwantyzatorów równomiernych z adaptacją, przy założeniu że kwantowany sygnał ma charakter gaussowski, podał Jayant [101]. Przyjmując bliższy rzeczywistości rozkład prawdopodobieństwa próbek sygnału mowy (np. rozkład Laplace'a), otrzymuje się niegaussowski rozkład kwantowanego sygnału [1]. Wymaga to przeprowadzenia modyfikacji metody Jayanta projektowania kwantyzatorów, co zostało opisane w p.2.1.3. Proponowany algorytm nadaje się również do projektowania kwantyzatorów nierównomiernych przy dowolnej szybkości adaptacji. Wykorzystano go do zaprojektowania szeregu kwantyzatorów współpracujących z układami DPCM o szybkości transmisji 16, 24 i 32 kbit/s.

2.1.2 Opis układu DPCM

Schemat układu i jego właściwości

Ogólny schemat blokowy kodera i dekodera DPCM pokazano na rys.2.1.



Rysunek 2.1: Układ kodera (a) i dekodera (b) systemu DPCM: Q - kwantyzator, P - predyktor

Kwantowaniu podlegają próbki sygnału różnicowego $\epsilon(n)$, powstającego w wyniku odjęcia sygnału aproksymującego (sygnału predykcji) $\hat{x}(n)$ od sygnału wejściowego, czyli sygnału

mowy $x(n)$, gdzie n jest numerem próbki. Poza szczególnymi przypadkami (modulacja delta) próbkowanie sygnału telefonicznego przebiega z częstotliwością 8000 próbek/s. Sygnał aproksymujący $\hat{x}(n)$ powstaje w predyktorze P, w wyniku cyfrowego przetwarzania poprzednich próbek sygnału wyjściowego $\tilde{x}(n-1), \tilde{x}(n-2), \dots, \tilde{x}(n-p)$. Przy braku przekłamań w procesie transmisji, w koderze i dekoderze generowany jest ten sam sygnał predykcji $\hat{x}(n)$ - wówczas spełnione jest równanie:

$$\tilde{x}(n) - x(n) = [\tilde{\epsilon}(n) + \hat{x}(n)] - [\epsilon(n) + \hat{x}(n)] = \tilde{\epsilon}(n) - \epsilon(n) = e(n) \quad (2.1)$$

Oznacza to, że sygnał wyjściowy dekodera różni się od sygnału wejściowego kodera jedynie o błąd kwantyzacji $e(n) = \tilde{\epsilon}(n) - \epsilon(n)$.

Miarą jakości sygnału wyjściowego $\tilde{x}(n) = x(n) + e(n)$ jest stosunek mocy¹ wejściowego sygnału mowy $x(n)$ do mocy zniekształceń $e(n)$:

$$SNR = \frac{\sigma_x^2}{\sigma_e^2} = \frac{\sigma_x^2}{\sigma_\epsilon^2} \cdot \frac{\sigma_\epsilon^2}{\sigma_e^2} = G_p \cdot SNR_q \quad (2.2)$$

gdzie: $G_p = \sigma_x^2/\sigma_\epsilon^2$ - zysk predykcji równy stosunkowi mocy sygnału mowy do mocy sygnału różnicowego, czyli błędu predykcji, $SNR_q = \sigma_\epsilon^2/\sigma_e^2$ - stosunek mocy sygnału kwantowanego do mocy szumu kwantowania. Dla uzyskania możliwie najlepszej jakości sygnału wyjściowego, predyktor projektuje się na maksimum G_p , a kwantyzator - na maksimum SNR_q .

Predyktor

Zadaniem predyktora jest wytwarzanie sygnału aproksymującego $\hat{x}(n)$ na podstawie poprzednich próbek sygnału wyjściowego $\tilde{x}(n-1), \tilde{x}(n-2), \dots, \tilde{x}(n-p)$ w taki sposób, aby moc sygnału różnicowego $\epsilon(n)$ (błąd predykcji) była jak najmniejsza. Predyktory działają najczęściej na zasadzie przekształceń liniowych - najprostszy jest predyktor transwersalny opisany równaniem

$$\hat{x}(n) = \sum_{i=1}^p a_i \tilde{x}(n-i) = \mathbf{a}^t \tilde{\mathbf{x}}(n-1) \quad (2.3)$$

gdzie $\mathbf{a}^t = [a_1, a_2, \dots, a_p]$ - transponowany wektor współczynników predykcji, p -rzęd predyktora, $\tilde{\mathbf{x}}(n-1) = [\tilde{x}(n-1), \tilde{x}(n-2), \dots, \tilde{x}(n-p)]^t$ - wektor poprzednich próbek sygnału wyjściowego. Przy stosowaniu predyktora o stałych współczynnikach można uzyskać (dla telefonicznego sygnału mowy) zysk predykcji G_p rzędu 6 dB; większy zysk predykcji, rzędu kilkunastu dB, można uzyskać stosując adaptację predyktora. Metody adaptacji predyktora można podzielić na dwie grupy: metody adaptacji "w przód" (są to tzw. metody blokowe) i metody adaptacji "wstecz" (są to najczęściej tzw. metody sekwencyjne). Adaptacja "w przód" polega na zgromadzeniu pewnego bloku próbek sygnału mowy (obejmującego 10-20 ms), wyznaczeniu optymalnych współczynników predykcji (najczęściej metodą Levinsona-Durbina [102]), po czym następuje zakodowanie całego bloku w układzie DPCM. Wadą tych metod adaptacji jest konieczność transmitowania do odbiornika, co 10-20 ms, całego wektora współczynników predykcji. Wady tej nie mają metody sekwencyjne (adaptacja "wstecz"

¹Ze względu na zerową wartość średnią, moce sygnałów oznacza się symbolami właściwymi dla wariancji

" [168]), w których predyktor jest wyznaczany jedynie w oparciu o transmitowany do odbiornika, skwantowany sygnał błędu predykcji $\tilde{\epsilon}(n)$. W koderze i dekodekcie przechowywane są aktualne wartości wektora współczynników predykcji $\mathbf{a}(n)$, którą modyfikuje się z częstotliwością próbkowania:

$$\mathbf{a}(n+1) = \mathbf{a}(n) + \Delta\mathbf{a}(n) \quad (2.4)$$

gdzie $\Delta\mathbf{a}(n)$: poprawka, wprowadzona w chwili n .

Metody adaptacji predyktora będą omówione w dalszej części pracy.

Kwantyzator

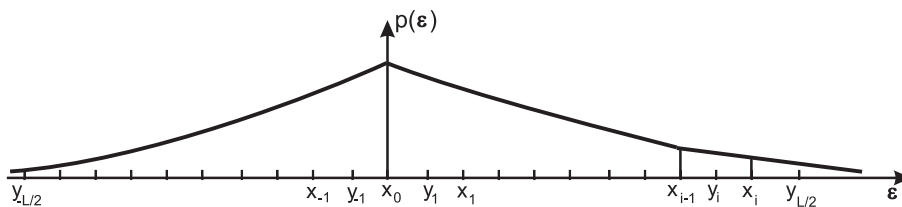
Zadaniem kwantyzatora jest zaokrąglenie wartości próbki sygnału wejściowego $\epsilon(n)$ do jednej z L wartości (poziomów kwantyzacji), co umożliwi zakodowanie próbki $\tilde{\epsilon}(n)$ w słowie binarnym o długości $b = \log_2 L$ bitów. Kwantyzator jest w pełni określony, gdy znane są poziomy kwantyzacji

$$y_{-L/2}, \dots, y_{-1}, y_1, \dots, y_{L/2}$$

oraz granice przedziałów kwantyzacji

$$x_{-L/2}, \dots, x_{-1}, x_0, x_1, \dots, x_{L/2}$$

gdzie $x_{-L/2} = -\infty$, $x_{L/2} = \infty$ - rys.2.2.



Rysunek 2.2: Zasada działania kwantyzatora: y_i - i -ty poziom kwantyzacji, od x_{i-1} do x_i rozciąga się i -ty przedział kwantyzacji (próbki leżące w tym przedziale będą zaokrąglone do wartości y_i), $p(\epsilon)$ - gęstość prawdopodobieństwa wartości próbki

Ze względu na właściwości sygnału (m.in. zerowa wartość średnia), zapewnia się symetrię przedziałów i poziomów kwantyzacji: $x_0 = 0$, $x_i = -x_{-i}$, $y_i = -y_{-i}$. Mając dany rozkład prawdopodobieństwa wartości próbki $p(\epsilon)$, można obliczyć moc szumu kwantyzacji ze wzoru [102]:

$$\sigma_e^2 = 2 \sum_{i=1}^{L/2} \int_{x_{i-1}}^{x_i} (\epsilon - y_i)^2 p(\epsilon) d\epsilon \quad (2.5)$$

Podstawiając moc kwantowanego sygnału

$$\sigma_e^2 = 2 \int_0^{\infty} \epsilon^2 p(\epsilon) d\epsilon$$

można określić stosunek mocy sygnału do szumu kwantyzacji $SNR_q = \sigma_\epsilon^2 / \sigma_e^2$. W przypadku kwantyzatora równomiernego, w którym kolejne punkty $x_0, y_1, x_1, y_2, x_2, y_3 \dots$ tworzą sieć o jednakowych odstępach Δ , wartość Δ i liczba poziomów kwantyzacji L stanowią pełny opis kwantyzatora. Dla danego sygnału ($p(\epsilon)$ znane) i liczby poziomów kwantyzacji L można zawsze obliczyć wartość Δ_{opt} , minimalizującą moc szumu kwantowania σ_e^2 , a co za tym idzie, maksymalizującą SNR_q .

W przypadku kwantyzatora nierównomiernego, minimalizację σ_e^2 (wzór 2.5) przeprowadza się ze względu na wszystkie wartości $y_1, \dots, y_{L/2}$ i $x_1, \dots, x_{L/2-1}$ [102]. Otrzymuje się w ten sposób kwantyzator optymalny, "dopasowany" do danego sygnału. Ażeby zapewnić prawidłową pracę kwantyzatora dla sygnałów o różnym poziomie (tzn. w warunkach, gdy σ_e^2 , a więc także $p(\epsilon)$ zależy od czasu), stosuje się adaptację kwantyzatora, uzależniając wielkości y_i, x_i od czasu. Adaptacja jest niezbędna w kodowaniu mowy, gdy liczba poziomów kwantyzacji jest mała ($L=4 - 16$).

Wyróżnia się metody adaptacji kwantyzatora "w przód" i "w tył". Adaptacja "w przód" polega na określeniu optymalnych parametrów kwantyzatora dla pewnego fragmentu sygnału wejściowego, a następnie skwantowaniu tego fragmentu sygnału. Tego typu adaptacją nie będziemy się zajmować, gdyż wymaga ona transmisji parametrów kwantyzatora do odbiornika, a ponadto nie może być stosowana w układach DPCM, w których sygnał wejściowy kwantyzatora nie może być z góry określony, ze względu na zamkniętą pętlę sprzężenia zwrotnego (rys 2.1).

Adaptacja "wstecz" polega na korekcy parametrów kwantyzatora jedynie w oparciu o wartości transmitowanych próbek $\tilde{\epsilon}(n)$. Większość stosowanych algorytmów adaptacji "wstecz" wywodzi się z koncepcji Jayanta [101] opracowanej dla kwantyzatora równomiernego. W algorytmie Jayanta parametr Δ , określający odległości poziomów kwantyzacji, zmienia się z częstotliwością próbkowania według wzoru:

$$\Delta(n+1) = \Delta(n) M[|I(n)|] \quad (2.6)$$

gdzie n - indeks próbki, $I(n)$ - numer poziomu kwantowania wykorzystywany w chwili n , $M(1) \dots M(L/2)$ - współczynniki adaptacji. Siatka poziomów kwantyzacji "rozciąga się" lub "kurczy" w zależności od wykorzystywanych poziomów kwantowania (niskie poziomy kwantowania implikują kurczenie się siatki, wysokie poziomy - jej rozciąganie). Według tej zasady działa szereg kwantyzatorów równomiernych i nierównomiernych [101],[32],[82],[102],[18], jednak w literaturze brak jest jasnych wskazówek co do wyboru parametrów, w szczególności wartości $M(|I(n)|)$. W przypadku systemu DPCM nie jest również jasne, jaką należy przyjąć gęstość prawdopodobieństwa próbek sygnału wejściowego kwantyzatora $p(\epsilon)$. Zagadnieniu optymalizacji parametrów kwantyzatorów adaptacyjnych równomiernych i nierównomiernych będzie poświęcona następną część pracy.

2.1.3 Kwantyzator z adaptacją

Kwantowanie sygnału stacjonarnego

Adaptację kwantyzatora stosuje się ze względu na niestacjonarność sygnału mowy, polegającą głównie na zmianach poziomu mocy tego sygnału (wariancji σ_x^2) w czasie. Nie ulega

jednak wątpliwości, że kwantyzator adaptacyjny winen prawidłowo przetwarzać również sygnały stacjonarne - gwarantuje to prawidłowe kwantowanie odcinków mowy o charakterze quasi-stacjonarnym (np. przeciągle wypowiedzianych samogłosek). Dla sygnału stacjonarnego lepsze wyniki daje jednak kwantyzator bez adaptacji o odpowiednio dobranych parametrach. Od kwantyzatora adaptacyjnego można jedynie wymagać, aby działał w tych warunkach w sposób zbliżony do kwantyzatora bez adaptacji. Aby prawidłowo zaprojektować kwantyzator bez adaptacji (w przypadku kwantyzatora równomiernego należy obliczyć Δ_{opt} , w przypadku kwantyzatora nierównomiernego wszystkie poziomy i przedziały kwantyzacji), wystarczy znać rozkład prawdopodobieństwa próbek sygnału wejściowego $p(\epsilon)$. Jayant [101] posługiwał się rozkładem gaussowskim. Rozkład ten niezbyt dokładnie odzwierciedla właściwości sygnału mowy - krótkie fragmenty mowy (głoski, pojedyncze wyrazy) lepiej opisuje rozkład dwustronnie wykładniczy (Laplace'a) [102]. Modelując sygnał mowy skorelowanym procesem stochastycznym o rozkładzie Laplace'a należy wziąć pod uwagę, że sygnał wejściowy kwantyzatora w układzie DPCM (tzn sygnał błędu predykcji) może mieć inny rozkład prawdopodobieństwa. Wyjaśnieniu tego zagadnienia jest częściowo poświęcona praca [1]. Po wykonaniu szeregu badań testem chi-kwadrat okazało się, że sygnał błędu predykcji może być opisany rozkładem prawdopodobieństwa, będącym kombinacją liniową dwóch znanych rozkładów: dwustronnie wykładniczego (Laplace'a) i gaussowskiego. Zawartość tych dwóch składników zależy od wartości współczynnika korelacji kolejnych próbek sygnału dwustronnie wykładniczego modelującego mowę, np. dla współczynnika korelacji równego 0.9, $p(\epsilon)$ jest w 70% rozkładem gaussowskim a w 30% Laplace'a, co można zapisać wzorem:

$$p(\epsilon) = 0.7 \frac{1}{\sqrt{2\pi\sigma_\epsilon^2}} \exp\left(-\frac{\epsilon^2}{2\sigma_\epsilon^2}\right) + 0.3 \frac{1}{\sqrt{2\sigma_\epsilon^2}} \exp\left(-\sqrt{2} \frac{|\epsilon|}{\sigma_\epsilon}\right) \quad (2.7)$$

W dalszych rozważaniach będziemy posługiwać się gęstością prawdopodobieństwa opisaną wzorem (2.7).

Dla sygnału o mocy jednostkowej ($\sigma_\epsilon^2 = 1$) można znaleźć optymalną siatkę poziomów i przedziałów kwantyzacji opisaną parametrami $\bar{y}_1 \dots \bar{y}_{L/2}, \bar{x}_1 \dots \bar{x}_{L/2-1}$. Dla kwantyzatora równomiernego wystarczy obliczyć wartość $\bar{\Delta}_{opt}$ minimalizującą σ_e^2 (wzór 2.5), podstawiając poziomy kwantowania $\bar{y}_i = (2i - 1)\bar{\Delta}_{opt}$ i granice przedziałów kwantowania $\bar{x}_i = 2i\bar{\Delta}_{opt}$. Algorytm wyznaczania optymalnych parametrów kwantyzatora nierównomiernego był wielokrotnie publikowany - patrz np. [102].

Dla sygnału o tym samym rozkładzie $p(\epsilon)$, ale innej mocy $\sigma_\epsilon^2 \neq 1$, optymalne parametry $y_1 \dots y_{L/2}, x_1 \dots x_{L/2-1}$ mogą być wyznaczone z zależności $y_i = \bar{y}_i\sigma_\epsilon, x_i = \bar{x}_i\sigma_\epsilon$, gdzie σ_ϵ jest wartością skuteczną sygnału wejściowego kwantyzatora (odchylenie standardowe rozkładu $p(\epsilon)$). Dla kwantyzatora równomiernego wystarczy wyznaczenie $\Delta_{opt} = \bar{\Delta}_{opt}\sigma_\epsilon$ i odtworzenie parametrów y_i i x_i ze wzorów $y_i = (2i - 1)\Delta_{opt}, x_i = 2i\Delta_{opt}$. W zależności od wartości σ_ϵ optymalna siatka poziomów i przedziałów kwantyzacji "rozciąga się" lub "kurczy się".

W przypadku, gdy moc σ_ϵ^2 nie jest znana, możemy posłużyć się jedynie estymacją wartości skutecznej sygnału σ_ϵ . Do estymacji możemy użyć tych wielkości, które są znane zarówno po stronie nadawczej jak i odbiorczej układu DPCM - są to numery poziomów kwantyzacji $I(n)$ w kolejnych chwilach n . W najprostszym przypadku otrzymuje się estymatę:

$$\hat{\sigma}_\epsilon(n+1) = \hat{\sigma}_\epsilon(n) M[|I(n)|] \quad (2.8)$$

gdzie $M(k)$ -współczynniki adaptacji (tzw. mnożniki), zależne od numeru poziomu kwantyzacji. Odpowiednie poziomy kwantyzacji w chwili $n + 1$ oblicza się z zależności $y_i(n + 1) = \bar{y}_i \hat{\sigma}_\epsilon(n + 1)$, a granice przedziałów kwantowania ze wzoru $x_i(n + 1) = \bar{x}_i \hat{\sigma}_\epsilon(n + 1)$.

W szczególności, dla kwantyzatora równomiernego obowiązują zależności: $\Delta(n + 1) = \bar{\Delta}_{opt} \hat{\sigma}_\epsilon(n + 1)$, $\Delta(n) = \bar{\Delta}_{opt} \hat{\sigma}_\epsilon(n)$, które po podstawieniu do (2.8) i pomnożeniu stronami przez $\bar{\Delta}_{opt}$ dają wzór (2.6). Oznacza to, że algorytm adaptacji opisany wzorem (2.8) jest uogólnieniem wzoru Jayanta [101] na przypadek kwantyzatora nierównomiernego.

Działanie kwantyzatora adaptacyjnego zależy w głównej mierze od wyboru parametrów adaptacji $M(1) \dots M(L/2)$. Swoboda wyboru tych współczynników jest ograniczona pewnym warunkiem, wynikającym z wymagania stabilnej pracy w procesie kwantowania sygnału stacjonarnego. Załóżmy, że mamy sygnał o jednostkowej mocy ($\sigma_\epsilon^2 = 1$) i rozpoczynamy kwantyzację z optymalnymi parametrami \bar{x}_i, \bar{y}_i (oznacza to, że $\hat{\sigma}_\epsilon(0) = 1$). Po dostatecznie długim czasie ($n \rightarrow \infty$) wartości estymaty (2.8) winny oscylować wokół wartości $\hat{\sigma}_\epsilon(n) \approx 1$, co oznacza, że

$$\prod_{k=1}^n M[|I(k)|] \approx 1 \quad (2.9)$$

Jeżeli przez $P(i)$ oznaczyć prawdopodobieństwo wystąpienia i -tego poziomu kwantowania (równe prawdopodobieństwu wystąpienia $(-i)$ -tego poziomu kwantowania), to można zauważyć, że iloczyn (2.9) składa się z $2nP(1)$ czynników o wartości $M(1)$, $2nP(2)$ czynników o wartości $M(2)$, itd. Po względnieniu tego faktu we wzorze (2.9) i podniesieniu obu stron do potęgi $\frac{1}{2n}$ otrzymuje się opublikowany przez Jayanta [101] warunek równowagi:

$$M(1)^{P(1)} M(2)^{P(2)} \dots M(L/2)^{P(L/2)} \approx 1 \quad (2.10)$$

W procesie kwantowania sygnału stacjonarnego o jednostkowej mocy kwantyzator z adaptacją winien zachowywać się podobnie, jak kwantyzator bez adaptacji o optymalnych parametrach \bar{x}_i, \bar{y}_i . Oznacza to, że średnie wartości $x_i(n), y_i(n)$ winny być zbliżone do \bar{x}_i, \bar{y}_i (dla kwantyzatora równomiernego wystarcza, że średnia wartość $\Delta(n)$ jest bliska $\bar{\Delta}_{opt}$ [81]). Ponadto prawdopodobieństwa wystąpienia poszczególnych poziomów kwantyzacji winny być zbliżone do odpowiednich prawdopodobieństw wyznaczonych dla kwantyzatora bez adaptacji, to jest:

$$P(i) = \int_{\bar{x}_{i-1}}^{\bar{x}_i} p(\epsilon) d\epsilon \quad (2.11)$$

Prawdopodobieństwa wyznaczone ze wzoru (2.11) należy podstawić do warunku równowagi (2.10).

Wyznaczenie współczynników adaptacji

Współczynniki adaptacji $M(1), \dots, M(L/2)$ muszą być wybrane w ten sposób, aby zapewnić "kurczenie się" siatki poziomów kwantyzacji przy zmniejszaniu się wartości skutecznej sygnału σ_ϵ , oraz "rozciąganie się" siatki w warunkach rosnącej wartości σ_ϵ . W związku z tym wartości $M(i)$ powinny być mniejsze od 1 dla niskich poziomów kwantyzacji i większe od 1 dla wysokich poziomów. Ponadto winien być spełniony warunek (2.10). Istnieje nieskończenie wiele wartości współczynników adaptacji spełniających ww. warunki. Wartości

współczynników adaptacji można wyznaczyć biorąc pod uwagę fakt, że wpływają one na estymatę wartości skutecznej sygnału (wzór 2.8). Weźmy pod uwagę następującą estymatę wariancji sygnału wejściowego kwantyzatora:

$$\hat{\sigma}_\epsilon^2(n+1) = \alpha \epsilon^2(n) + (1-\alpha) \hat{\sigma}_\epsilon^2(n) \quad (2.12)$$

gdzie $\alpha < 1$ jest miarą szybkości reakcji estymatora na zmiany wariancji sygnału. Wartość próbki sygnału różnicowego $\epsilon(n)$ nie jest znana po stronie odbiorczej układu DPCM, wobec czego należy wyrazić ją przez wartość skwantowaną $\tilde{\epsilon}(n)$:

$$\epsilon(n) = \tilde{\epsilon}(n) - e(n) = y_{I(n)} - e(n) = \bar{y}_{I(n)} \hat{\sigma}_\epsilon(n) - e(n) \quad (2.13)$$

Po uwzględnieniu tego wyrażenia w (2.12) otrzymuje się:

$$\hat{\sigma}_\epsilon^2(n+1) = \alpha \bar{y}_{I(n)}^2 \hat{\sigma}_\epsilon^2(n) - 2\alpha \bar{y}_{I(n)} \hat{\sigma}_\epsilon(n) e(n) + \alpha e^2(n) + (1-\alpha) \hat{\sigma}_\epsilon^2(n) \quad (2.14)$$

co można przepisać w postaci:

$$\hat{\sigma}_\epsilon^2(n+1) = \hat{\sigma}_\epsilon^2(n) \left[(1-\alpha) + \alpha \bar{y}_{I(n)}^2 - \frac{2\alpha \bar{y}_{I(n)} e(n)}{\hat{\sigma}_\epsilon(n)} + \frac{\alpha e^2(n)}{\hat{\sigma}_\epsilon^2(n)} \right] \quad (2.15)$$

Po porównaniu z (2.8) okazuje się, że wyrażenie w nawiasie kwadratowym jest kwadratem współczynnika adaptacji $M[I(n)]$, czyli

$$M(i) \approx \sqrt{(1-\alpha) + \alpha \bar{y}_i^2} \quad (2.16)$$

We wzorze (2.16) pominięto składniki zawierające małą wartość błędu kwantyzacji $e(n)$. Jest to uzasadnione w przypadku, gdy kwantowanie odbywa się bez przesterowania. W warunkach przesterowania, gdy $\epsilon(n) > y_{L/2}$ przyjmowany jest poziom kwantowania $I(n) = L/2$ (gdy $\epsilon(n) < y_{-L/2}$ jest to poziom $I(n) = -L/2$). Wówczas błąd $e(n) = \tilde{\epsilon}(n) - \epsilon(n)$ może osiągnąć dużą wartość bezwzględną, przy czym iloczyn $e(n)\tilde{\epsilon}(n) = e(n)\bar{y}_{I(n)}\hat{\sigma}_\epsilon(n)$ jest ujemny. Oznacza to, że oba pominięte składniki zawierające $e(n)$ są dodatnie i mogą mieć dużą wartość. Stąd wypływa wniosek, że wzór (2.16) szacuje wartość $M(L/2)$ z niedomiarem - należy więc dodać poprawkę, której wartość trzeba dobrać eksperymentalnie. Po podstawieniu dla kwantyzatora równomiernego $\bar{y}_i = (2i-1)\bar{\Delta}_{opt}$ oraz $\alpha = 0.5$ wzór (2.16) staje się identyczny z formułą Jayanta [101]. Oznacza to, że wzór (2.16) jest uogólnieniem formuły Jayanta na przypadek kwantyzatora nierównomiernego, oraz dowolnej szybkości adaptacji (α).

2.1.4 Wyniki obliczeń i symulacji

Projektowanie kwantyzatora z adaptacją

W pierwszym etapie obliczeń, posługując się rozkładem prawdopodobieństwa próbek błędu predykcji (2.7) i zakładając jednostkową wariancję sygnału, wyznaczono parametry kwantyzatora równomiernego ($\bar{\Delta}_{opt}$) oraz nierównomiernego (\bar{x}_i, \bar{y}_i) bez adaptacji, podobnie jak w [102]. Wyznaczono również prawdopodobieństwa $P(i)$ ze wzoru (2.11) oraz stosunek mocy

sygnału wejściowego kwantyzatora do szumu kwantyzacji $SNR_q = \sigma_\epsilon^2 / \sigma_e^2$, posługując się wzorem (2.5). Np. dla kwantyzatora równomiernego o $L=8$ poziomach kwantyzacji otrzymano $SNR_q = 13.07$ dB.

Zakładając wartość współczynnika szybkości adaptacji α można obliczyć współczynniki adaptacji $M(1), \dots, M(L/2)$ ze wzoru (2.16). Wyniki dla 8-poziomowego kwantyzatora równomiernego i nierównomiernego podano odpowiednio w tabl. 2.1 i 2.2 [1].

Tabela 2.1: Współczynniki adaptacji dla kwantyzatora równomiernego o $L=8$ poziomach kwantyzacji

mnożnik	α		
	0.1	0.2	0.5
$M(1)$	0.95	0.9	0.74
$M(2)$	0.99	0.99	0.97
$M(3)$	1.07	1.14	1.32
$M(4)$	1.18	1.33	1.71
$M(4)$ skorygowane	1.3	1.6	2.2

Tabela 2.2: Współczynniki adaptacji dla kwantyzatora nierównomiernego o $L=8$ poziomach kwantyzacji

mnożnik	α		
	0.1	0.2	0.5
$M(1)$	0.93	0.89	0.72
$M(2)$	0.98	0.95	0.87
$M(3)$	1.05	1.07	1.18
$M(4)$	1.18	1.35	1.77
$M(4)$ skorygowane	1.5	1.8	3.0

Wartość $M(L/2)$ wyznaczona jest z niedomiarem i należy ją skorygować. Próby wyznaczenia $M(L/2)$ z warunku (2.10) nie dały zadowalających rezultatów. Lepsze wyniki dała metoda symulacyjna. Eksperymenty polegały na generowaniu stacjonarnego sygnału $\epsilon(n)$ o rozkładzie prawdopodobieństwa (2.7) i jednostkowej wariancji $\sigma_\epsilon^2 = 1$. Sygnał ten był kwantowany przy użyciu zasymulowanego kwantyzatora adaptacyjnego (przyjmowano warunek początkowy $\hat{\sigma}_\epsilon(0) = 1$, co zapewniało start z optymalnymi wartościami \bar{x}_i, \bar{y}_i).

Po przeprowadzeniu serii obliczeń otrzymywano $SNR_q = \sum_n \epsilon^2(n) / \sum_n e^2(n)$ w funkcji $M(L/2)$ i wybierano wartość współczynnika $M(L/2)$ maksymalizującą SNR_q . Wyniki dla 8-poziomowego kwantyzatora równomiernego i nierównomiernego podano w ostatnich wierszach odpowiednio w tabl. 2.1 i 2.2. Jednocześnie kontrolowano wartości średnie poziomów kwantyzacji w każdej symulacji (wystarcza do tego obliczenie wartości średniej $\hat{\sigma}_\epsilon(n)$). Na

przykład, kwantując 5000 próbek sygnału przy użyciu 8-poziomowego kwantyzatora równomiernego o podanych wyżej współczynnikach adaptacji otrzymano wartość średnią estymaty wartości skutecznej $\hat{\sigma}_\epsilon(n)$ równą 0.93, co uznano za zadowalające.

Po ostatecznym wyznaczeniu współczynników adaptacji można sprawdzić warunek równowagi (2.10). We wszystkich analizowanych przypadkach otrzymywano wartości lewej strony wyrażenia (2.10) należące do przedziału (0.97,1). Spełnienie w zadowalającym stopniu warunku równowagi oraz utrzymywanie się wartości średnich poziomów kwantyzacji zbliżonych do \bar{y}_i świadczy o tym, że zaprojektowany kwantyzator adaptacyjny zachowuje się w procesie kwantowania sygnału stacjonarnego w podobny sposób, jak kwantyzator nieadaptacyjny o optymalnych parametrach.

Otwartą pozostaje sprawa wyboru szybkości adaptacji α . Im większa jest wartość α , tym szybsza jest reakcja kwantyzatora na zmiany mocy sygnału, lecz gorsze są wyniki w procesie kwantowania sygnału stacjonarnego. Ilustracją tego zjawiska są wyniki symulacji zamieszczone w tabl. 2.3 i 2.4. Eksperymenty polegały na kwantowaniu $M=10, 100, 1000$ oraz 5000 próbek stacjonarnego sygnału $\epsilon(n)$ o rozkładzie (2.7) i jednostkowej wariancji $\sigma_\epsilon^2 = 1$. Warunki początkowe kwantyzatora, tj. wartość $\hat{\sigma}_\epsilon(0)$ ustalającą początkową siatkę poziomów i przedziałów kwantyzacji $y_i(0) = \bar{y}_i \hat{\sigma}_\epsilon(0)$, $x_i(0) = \bar{x}_i \hat{\sigma}_\epsilon(0)$, wybierano w pięciu wariantach: na poziomie optymalnym $\hat{\sigma}_\epsilon(0) = 1$, w odległości ± 10 dB od poziomu optymalnego, oraz w odległości ± 20 dB od poziomu optymalnego. W tabelach zamieszczono wartości $SNR_q = \sum_{n=0}^{M-1} \epsilon^2(n) / \sum_{n=0}^{M-1} e^2(n)$, czyli stosunek mocy sygnału do szumu kwantyzacji. Porównując tabl. 2.3 i 2.4 można zauważyć, że większa szybkość adaptacji ($\alpha = 0.5$) gwarantuje lepsze wyniki dla krótkich fragmentów sygnału ($M=10-100$ próbek), natomiast mniejsza szybkość adaptacji ($\alpha = 0.2$) zapewnia o 1 dB lepszy SNR_q dla długich fragmentów sygnału ($M=1000-5000$ próbek).

Tabela 2.3: SNR_q [dB] w procesie kwantowania M próbek sygnału stacjonarnego przy użyciu kwantyzatora równomiernego o $L=8$ poziomach kwantowania; szybkość adaptacji $\alpha = 0.5$

$\hat{\sigma}_\epsilon(0)$ [dB]	M			
	10	100	1000	5000
-20	5.9	11.6	10.6	10.9
-10	9.0	12.4	10.5	10.9
0	13.6	12.9	10.5	10.9
10	8.1	12.2	10.5	10.8
20	-3.7	7.7	9.9	10.7

Porównując tabl. 2.4 i 2.5 można stwierdzić, że kwantyzator nierównomierny zachowuje się lepiej od kwantyzatora równomiernego o tych samych parametrach (liczba poziomów kwantowania, szybkość adaptacji) - zarówno w procesie kwantowania krótkich, jak i długich fragmentów sygnału.

Tabela 2.4: SNR_q [dB] w procesie kwantowania M próbek sygnału stacjonarnego przy użyciu kwantyzatora równomiernego o $L=8$ poziomach kwantowania; szybkość adaptacji $\alpha = 0.2$

$\hat{\sigma}_\epsilon(0)$ [dB]	M			
	10	100	1000	5000
-20	6.4	11.3	11.3	11.8
-10	9.3	13.2	11.3	11.8
0	14.2	13.4	11.7	11.9
10	4.5	11.9	11.4	11.9
20	-6.8	5.1	10.1	11.6

Tabela 2.5: SNR_q [dB] w procesie kwantowania M próbek sygnału stacjonarnego przy użyciu kwantyzatora nierównomiernego o $L=8$ poziomach kwantowania; szybkość adaptacji $\alpha = 0.2$

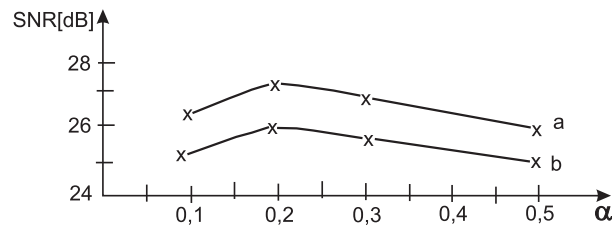
$\hat{\sigma}_\epsilon(0)$ [dB]	M			
	10	100	1000	5000
-20	10.0	11.5	11.7	12.6
-10	8.1	12.8	11.9	12.7
0	14.6	14.1	12.0	12.8
10	10.4	14.0	11.9	12.7
20	-0.8	9.1	11.3	12.5

Symulacja układu DPCM

Badania symulacyjne z wykorzystaniem sygnału stacjonarnego nie stanowią wystarczającej podstawy do określenia wszystkich parametrów kwantyzatora - przede wszystkim szybkości adaptacji α . Można to uczynić jedynie w oparciu o wyniki symulacji całego kodera DPCM wraz z kwantyzatorem i predyktorem (rys.2.1). W przeprowadzonych badaniach symulacyjnych stosowano predyktor transwersalny opisany wzorem (2.3) o $p = 8$ współczynnikach predykcji. Adaptacja współczynników predykcji przebiegała według zasady gradientu stochastycznego (opisanej w następnym podrozdziale). Przetwarzano frazę sygnału mowy o długości ok. 4 sekund.

Badano kwantyzatory równomierne i nierównomierne zaprojektowane według zasad podanych w poprzednim punkcie. Na rys.2.3 zamieszczono wykres wartyości $SNR = \sigma_x^2/\sigma_\epsilon^2$ w funkcji szybkości adaptacji α dla kwantyzatora równomiernego i nierównomiernego o $L=16$ poziomach. Optymalny współczynnik szybkości adaptacji α ma wartość 0.2, lecz nie jest to wartość krytyczna. Zmiany szybkości adaptacji w przedziale 0.1-0.5 nie obniżają SNR o więcej niż 1 dB. Kwantyzator nierównomierny zachowuje ponad 1 dB przewagę nad kwantyzatorem równomiernym.

Po ustaleniu optymalnej szybkości adaptacji $\alpha = 0.2$ wyznaczono SNR w funkcji liczby



Rysunek 2.3: SNR kodera DPCM w funkcji szybkości adaptacji α kwantyzatora ($L=16$ poziomów): a - kwantyzator nierównomierny, b - kwantyzator równomierny

poziomów kwantyzacji L (tabl. 2.6).

Tabela 2.6: Stosunek mocy sygnału do mocy zniekształceń SNR [dB] dla układu DPCM z kwantyzatorem o L poziomach kwantyzacji

kwantyzator	L		
	4	8	16
równomierny	14.4	20.4	25.5
nierównomierny	14.3	20.9	26.7

2.1.5 Podsumowanie

Dokonano przeglądu metod projektowania równomiernych i nierównomiernych kwantyzatorów adaptacyjnych, dla dowolnego sygnału wejściowego i dowolnej szybkości adaptacji. Przedstawiono zasadę działania kodera DPCM i omówiono wyniki symulacji koderów DPCM o szybkości transmisji 24 i 32 kbit/s - przy szybkości transmisji 16 kbit/s szum kwantyzacji jest nazbyt dokuczliwy (tabl.2.6).

2.2 Algorytmy predykcji

2.2.1 Wprowadzenie

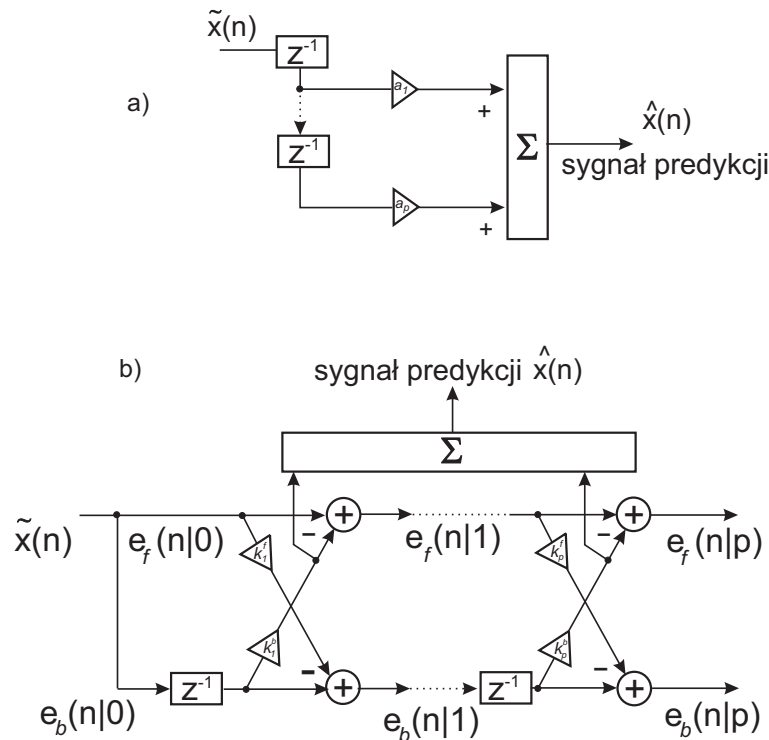
Niemal wszystkie stosowane obecnie metody kodowania telefonicznego sygnału mowy wywodzą się z zasady predykcji liniowej. Należy tu wymienić adaptacyjne kodery różnicowe (ADPCM) dla przepływności binarnych 16 - 32 kbit/s [102], kodery predykcyjne o pobudzeniu impulsowym (MP) dla przepływności binarnych 8 - 16 kbit/s [7], kodery predykcyjne o pobudzeniu stochastycznym (CELP) dla przepływności binarnych 4.8 - 12 kbit/s [5] i wokodery predykcyjne dla przepływności binarnych rzędu 2 kbit/s [165].

W niniejszym podrozdziale opisane są algorytmy predykcji dla adaptacyjnych koderów różnicowych (ADPCM) o przepływnościach binarnych 16 - 32 kbit/s. Algorytmy predykcji liniowej można podzielić na blokowe i rekursywne (sekwencyjne). Metody blokowe są rzadko stosowane w koderach ADPCM. Ich omówienie można znaleźć w [132]. Przegląd algorytmów blokowych, gradientowych i rekursywnych LS z uwzględnieniem licznych wariantów wynikających np. z różnych kryteriów optymalizacji filtra predykcyjnego można znaleźć w [192]. W kolejnych punktach niniejszej pracy dokonano przeglądu algorytmów rekursywnych predykcji liniowej, w tym algorytmów gradientu stochastycznego i najmniejszych kwadratów w wersjach autokorelacyjnej i kowariancyjnej. Uwzględniono dwie struktury predyktora: transversalną i kratową. Opisano programy symulacyjne dla wymienionych wyżej algorytmów predykcji liniowej i wyniki symulacji dla 3 fraz sygnału mowy (w sumie ok. 10 s materiału dźwiękowego). Przebadano wiele wariantów omawianych algorytmów (np. z normalizacją i bez normalizacji) i przeanalizowano ich właściwości w zależności od stałych określających szybkość zbieżności, rząd filtra predykcyjnego, stałą czasową, długość okna. Przeprowadzono także badania symulacyjne całego układu ADPCM, składającego się z adaptacyjnego predyktora i adaptacyjnego kwantyzatora, dla szybkości transmisji 16, 24 i 32 kbit/s. Projektowanie adaptacyjnych kwantyzatorów zostało przedstawione w poprzednim podrozdziale - patrz również [2].

2.2.2 Predyktor w układzie DPCM

Układ adaptacyjnego koderu różnicowego (ADPCM) pokazano na rys.2.1. Miarą jakości sygnału wyjściowego $\tilde{x}(n) = x(n) + e(n)$ jest stosunek mocy składowej użytecznej $x(n)$ do mocy zniekształceń $e(n)$: $SNR = G_p SNR_q$ - wzór (2.2). Zysk predykcji G_p , stosunek mocy sygnału wejściowego kwantyzatora do mocy szumu kwantyzacji SNR_q oraz wartość SNR sygnału wyjściowego określa się często w mierze segmentowej, jako średnią arytmetyczną wartości wyznaczonych w mierze logarytmicznej (dB) w segmentach o długości 10-30 ms (wzór 1.1). Duży zysk predykcji G_p (stosunek mocy sygnału wejściowego $x(n)$ do mocy błędu predykcji $\epsilon(n) = x(n) - \hat{x}(n)$) umożliwia uzyskanie dobrej jakości sygnału wyjściowego mimo stosowania kwantyzatora o małym SNR_q . Z tego względu prawidłowe działanie układu ADPCM zależy w dużej mierze od poprawnego działania predyktora.

W układach ADPCM najczęściej stosowane są predyktory AR [90] o strukturze transwersalnej i kratowej (rys.2.4). Innymi predyktorami (np. MA, ARMA) nie będziemy się tutaj zajmować.



Rysunek 2.4: Predyktor liniowy AR rzędu p w strukturze transwersalnej (a) i kratowej (b)

Predyktor transwersalny opisany jest wektorem współczynników predykcji $\mathbf{a} = [a_1, \dots, a_p]^t$ (t oznacza transpozycję), natomiast predyktor kratowy współczynnikami odbicia "w przód" k_1^f, \dots, k_p^f i "wstecz" k_1^b, \dots, k_p^b . W przypadku predyktorów o stałych współczynnikach można łatwo zapewnić równoważność obu struktur predyktora. Wystarczy spełnić równania wynikające z algorytmu Levinsona [132], podstawiając $k_i^f = k_i^b$, $i = 1, \dots, p$. W przypadku predyktorów adaptacyjnych, których współczynniki są funkcjami czasu, zapewnienie równoważności obu struktur jest sprawą bardziej skomplikowaną i jest cechą tylko nielicznych algorytmów adaptacji (np. niektórych wariantów metody najmniejszych kwadratów).

Algorytmy adaptacji predyktora można podzielić na blokowe i rekursywne (sekwencyjne). Metody blokowe utrzymują stałe wartości współczynników predykcji w obrębie ustalonego odcinka czasu - skokowe zmiany ich wartości następują co 10-20 ms [132]. Algorytmy te są rzadko stosowane w koderach ADPCM, gdyż wymagają transmitowania współczynników predykcji do odbiornika. Metody rekursywne adaptacji predyktora polegają na korekcji wartości współczynników predykcji w każdej chwili n . Algorytm adaptacji predyktora transwersalnego wykorzystuje zatem rekursję:

$$\mathbf{a}(n+1) = \mathcal{R} [\mathbf{a}(n)] \quad (2.17)$$

gdzie $\mathbf{a}(n) = [a_1(n), a_2(n), \dots, a_p(n)]^t$ - wektor współczynników predykcji w chwili n , p -rzęd predyktora, oraz \mathcal{R} - pewna funkcja. Funkcja \mathcal{R} uwzględnia wartości sygnału wejściowego predyktora $\tilde{x}(i)$ dla $i < n$ oraz kryterium adaptacji predyktora. Podobna rekursja wykorzystywana jest do adaptacji predyktora kratowego, gdzie wyznaczane są wartości współczynników odbicia $k_1^f(n+1), \dots, k_p^f(n+1)$ i $k_1^b(n+1), \dots, k_p^b(n+1)$. Metody sekwencyjne umożliwiają wyznaczenie identycznych współczynników predykcji w koderze i dekodez ADPCM (przy braku przekłamań w transmisji), bez konieczności transmitowania dodatkowych parametrów. Przesyła się jedynie próbki skwantowanego sygnału błędu predykcji $\tilde{\epsilon}(n)$.

Należy zwrócić uwagę na fakt, że predyktor w układzie ADPCM występuje w pętli sprzężenia zwrotnego, zarówno w koderze, jak i w dekodez. Powstaje zatem problem kontroli stabilności takiego układu ze sprzężeniem zwrotnym. Utrzymanie stabilności jest łatwiejsze w przypadku predyktora kratowego. Jak wiadomo [165] jest ona zapewniona, gdy spełnione są warunki:

$$|k_m^f| < 1, \quad |k_m^b| < 1, \quad m = 1, \dots, p \quad (2.18)$$

Odpowiednie warunki stabilności dla układu z predyktorem transwersalnym nie dadzą się sformułować w tak prosty sposób.

Innym problemem jest odporność układu ADPCM na błędy w transmisji. Dotyczy to algorytmów adaptacji kwantyzatora i predyktora. Wskutek przekłamań transmitowanej próbki $\epsilon(n)$ w odbiorniku generowane są fałszywe wartości poziomów kwantyzacji i wartości współczynników predykcji. Wpływa to ujemnie na dalszą pracę układu ADPCM. Dobre algorytmy adaptacji kwantyzatora i predyktora charakteryzują się tym, że powstałe w ten sposób zakłócenie ma stosunkowo krótki czas propagacji.

2.2.3 Adaptacja predyktora metodami gradientu stochastycznego

Predyktory o strukturze transwersalnej

Optymalny predyktor powinien maksymalizować zysk predykcji G_p , a co za tym idzie, minimalizować wartość średniokwadratową (moc) błędu predykcji $E[\epsilon^2(n)] = \sigma_\epsilon^2$. Błąd predykcji w chwili n można zapisać w postaci:

$$\epsilon(n) = x(n) - \hat{x}(n) = x(n) - \mathbf{a}^t(n) \tilde{\mathbf{x}}(n-1) \quad (2.19)$$

gdzie $\tilde{\mathbf{x}}(n-1) = [\tilde{x}(n-1), \tilde{x}(n-2), \dots, \tilde{x}(n-p)]^t$ - wektor poprzednich próbek sygnału wejściowego predyktora.

Wartości $x(n)$ i $\epsilon(n)$ nie mogą być wykorzystane do wyznaczania współczynników predykcji, gdyż nie są znane po stronie odbiorczej. W związku z tym zastępuje się je skwantowanymi odpowiednikami $\tilde{x}(n)$ i $\tilde{\epsilon}(n)$. Wielkości te związane są zależnością analogiczną do zależności (2.19):

$$\tilde{\epsilon}(n) = \tilde{x}(n) - \hat{\tilde{x}}(n) = \tilde{x}(n) - \mathbf{a}^t(n) \tilde{\mathbf{x}}(n-1) \quad (2.20)$$

Sygnał predykcji $\hat{\tilde{x}}(n)$ został wyznaczony przy użyciu współczynników predykcji $\mathbf{a}(n)$, przygotowanych już w poprzedniej chwili $n-1$. W chwili n przygotowuje się współczynniki dla następnej chwili, dodając poprawkę $\Delta\mathbf{a}(n)$:

$$\mathbf{a}(n+1) = \mathbf{a}(n) + \Delta\mathbf{a}(n) \quad (2.21)$$

Poprawka jest skierowana przeciwnie do kierunku wzrostu minimalizowanej funkcji, czyli mocy skwantowanego błędu predykcji $\tilde{\epsilon}^2(n)$. Kierunek wzrostu wskazywany jest przez gradient tej funkcji. Wynika stąd następująca wartość poprawki:

$$\begin{aligned}\Delta \mathbf{a}(n) &= -\frac{\beta}{2} \frac{\partial}{\partial \mathbf{a}} \tilde{\epsilon}^2(n) = -\frac{\beta}{2} 2 \tilde{\epsilon}(n) \frac{\partial}{\partial \mathbf{a}} \tilde{\epsilon}(n) \\ &= -\beta \tilde{\epsilon}(n) \frac{\partial}{\partial \mathbf{a}} [\tilde{x}(n) - \mathbf{a}^t(n) \tilde{\mathbf{x}}(n-1)] = \beta \tilde{\epsilon}(n) \tilde{\mathbf{x}}(n-1)\end{aligned}\quad (2.22)$$

Rekursja

$$\mathbf{a}(n+1) = \mathbf{a}(n) + \beta \tilde{\epsilon}(n) \tilde{\mathbf{x}}(n-1) \quad (2.23)$$

opisuje *metodę gradientu stochastycznego* wyznaczania współczynników predykcji [90]. Zbieżność wektora (2.23) jest zagwarantowana dla dostatecznie małego β . Parametr β reguluje jednocześnie szybkość zbieżności. Zbyt mała jego wartość oznacza małą poprawkę $\beta \tilde{\epsilon}(n) \tilde{\mathbf{x}}(n-1)$ i wolną zbieżność wektora współczynników. Zbyt duża wartość β prowadzi do rozbieżności wektora współczynników predykcji.

Szybkość zbieżności zależy także od poziomu przetwarzanego sygnału $x(n)$. Wartość poprawki zależy od kwadratu amplitudy sygnału, gdyż amplituda sygnału wpływa zarówno na wektor $\tilde{\mathbf{x}}(n-1)$ jak i na błąd predykcji $\tilde{\epsilon}(n)$. Jest to zjawisko niekorzystne, gdyż szybkość zbieżności nie powinna zależeć od poziomu sygnału. Można wyeliminować to zjawisko, uzależniając parametr β od mocy sygnału:

$$\beta(n) = \frac{L_\beta}{\hat{\sigma}_x^2(n) + M_\beta} \quad (2.24)$$

Stała L_β reguluje szybkość zbieżności, stała M_β zapobiega dzieleniu przez zero w przypadku braku sygnału na wejściu predyktora, natomiast $\hat{\sigma}_x^2(n)$ jest estymatą mocy sygnału w chwili n , którą oblicza się zgodnie z rekursją:

$$\hat{\sigma}_x^2(n) = \bar{\alpha} \hat{\sigma}_x^2(n-1) + (1 - \bar{\alpha}) \tilde{x}^2(n) \quad (2.25)$$

Wielkość $\bar{\alpha}$ określa szybkość reakcji estymatora na zmiany mocy chwilowej sygnału. Dla mowy spróbkowanej z częstotliwością 8 kHz przyjmuje się $\bar{\alpha} = 0.9 - 0.99$. Otrzymuje się w ten sposób *algorytm gradientu stochastycznego z normalizacją*.

W systemach ADPCM ważna jest odporność algorytmu adaptacji predyktora na błędy transmisji. Pojedynczy błąd transmisji (tzn. fałszywa wartość $\tilde{\epsilon}(n)$ po stronie odbiorczej) spowoduje rozbieżność ciągu wektorów $\mathbf{a}(n)$ obliczanych w nadajniku i w odbiorniku. Aby tego uniknąć, stosuje się *algorytm gradientu stochastycznego z tłumieniem*:

$$\mathbf{a}(n+1) = (1 - \mu) \mathbf{a}(n) + \beta \tilde{\epsilon}(n) \tilde{\mathbf{x}}(n-1) \quad (2.26)$$

Parametr $\mu \ll 1$ decyduje o szybkości "tłumienia" pojawiających się błędów transmisji. W przypadku braku sygnału na wejściu predyktora wyznaczone współczynniki predykcji dążą do zera - w przypadku algorytmu gradientu stochastycznego (2.23) bez tłumienia, współczynniki te pozostałyby niezmienniczone.

Predyktory o strukturze kratowej

Predyktor kratowy pokazano na rys. 2.4b. Równania m -tego ogniwa kraty mają następującą postać:

$$\begin{cases} e_f(n|m) = e_f(n|m-1) - k_m^b e_b(n-1|m-1) \\ e_b(n|m) = e_b(n-1|m-1) - k_m^f e_f(n|m-1) \end{cases} \quad (2.27)$$

Nietrudno zauważyć, że $e_f(n|p) = \tilde{x}(n) - \hat{x}(n)$ jest skwantowanym błędem predykcji otrzymanym przy wykorzystaniu predyktora rzędu p - dotychczas oznaczanym jako $\tilde{\epsilon}(n)$ - rys. 2.1. Zaletą struktury kratowej jest to, że optymalizację współczynników k_m^b i k_m^f można przeprowadzić niezależnie dla każdego ogniwa kraty. Współczynnik k_m^b wyznacza się minimalizując moc błędu predykcji $e_f^2(n|m)$. Do poprzedniej wartości $k_m^b(n)$ dodaje się poprawkę skierowaną przeciwnie do gradientu (w tym wypadku pochodnej) funkcji $e_f^2(n|m)$. W chwili n gradient ma postać:

$$\begin{aligned} \frac{\partial}{\partial k_m^b} e_f^2(n|m) &= 2 e_f(n|m) [-e_b(n-1|m-1)] \\ &= -2 e_f(n|m-1) e_b(n-1|m-1) + 2 k_m^b(n) e_b^2(n-1|m-1) \end{aligned} \quad (2.28)$$

Iteracje wykonuje się według wzoru:

$$\begin{aligned} k_m^b(n+1) &= k_m^b(n) - \frac{\beta}{2} \frac{\partial}{\partial k_m^b} e_f^2(n|m) \\ &= [1 - \beta e_b^2(n-1|m-1)] k_m^b(n) + \\ &+ \beta e_f(n|m-1) e_b(n-1|m-1) \end{aligned} \quad (2.29)$$

Podobne wyrażenie otrzymuje się dla współczynnika k_m^f , który wyznacza się minimalizując $e_b^2(n|m)$ [90]:

$$k_m^f(n+1) = [1 - \beta e_f^2(n|m-1)] k_m^f(n) + \beta e_f(n|m-1) e_b(n-1|m-1) \quad (2.30)$$

Podobnie jak w predyktorze transwersalnym, szybkość adaptacji i zbieżność algorytmu zależy od wyboru wartości parametru β . Aby uniezależnić szybkość adaptacji od poziomu sygnału, stosuje się algorytm gradientu stochastycznego z normalizacją. Parametr β we wzorach (2.29) i (2.30) zależy wówczas od mocy przetwarzanych w m -tym ogniwie kraty sygnałów wejściowych:

$$\beta_m(n) = \frac{L_\beta}{\hat{\sigma}_m^2(n) + M_\beta} \quad (2.31)$$

która jest estymowana zgodnie z rekursjami ($m = 1, \dots, p$):

$$\hat{\sigma}_m^2(n) = \bar{\alpha} \hat{\sigma}_m^2(n-1) + (1 - \bar{\alpha}) [e_f^2(n|m-1) + e_b^2(n-1|m-1)] \quad (2.32)$$

Stosowany jest także wariant algorytmu adaptacji, w którym zakłada się jednakowe wartości współczynników $k_m = k_m^b = k_m^f$ i oblicza się je minimalizując sumę $e_f^2(n|m) + e_b^2(n|m)$. Otrzymuje się wówczas następujący wzór iteracyjny:

$$k_m(n+1) = \left\{ 1 - \beta [e_f^2(n|m-1) + e_b^2(n-1|m-1)] \right\} k_m(n) + 2 \beta e_f(n|m-1) e_b(n-1|m-1) \quad (2.33)$$

Stosowane są również inne warianty algorytmu adaptacji, np. algorytm z tłumieniem ($\mu \ll 1$):

$$k_m^b(n+1) = [1 - \mu - \beta e_b^2(n-1|m-1)] k_m^b(n) + \beta e_f(n|m-1) e_b(n-1|m-1) \quad (2.34)$$

$$k_m^f(n+1) = [1 - \mu - \beta e_f^2(n|m-1)] k_m^f(n) + \beta e_f(n|m-1) e_b(n-1|m-1) \quad (2.35)$$

W układach ADPCM celowe jest kontrolowanie stabilności układu z predyktorem w pętli sprzężenia zwrotnego. W przypadku predyktora kratowego ogranicza się to do sprawdzania, czy obliczone wartości k_m^f i k_m^b leżą w przedziale $(-1, +1)$ dla $m = 1, 2, \dots, p$. Jeżeli ze wzorów (2.29), (2.30), (2.33), (2.34) lub (2.35) otrzyma się wartości spoza tego przedziału, to należy sprowadzić je do wartości $+1$ lub -1 . Inną możliwą reakcją na wykrycie niestabilności jest wyzerowanie współczynników k_m^b i k_m^f w "niestabilnym" ogniwie i we wszystkich dalszych ogniwach kraty. Należy podkreślić, że struktury transversalna i kratowa, z adaptacją współczynników metodą gradientu stochastycznego, nie są sobie równoważne. Dla obu tych struktur należy przeprowadzić osobne badania symulacyjne.

2.2.4 Adaptacja predyktora metodą najmniejszych kwadratów

Podstawowe warianty algorytmu najmniejszych kwadratów

Metody *najmniejszych kwadratów* (LS) obejmują algorytmy rekursywne, w których bieżące wartości współczynników predykcji $\mathbf{a}(n)$ wyznaczane są w taki sposób, aby minimalizowały energię błędu predykcji w obrębie określonego okna. Okno rozciąga się od chwili n_0 do chwili bieżącej n . Gdy $n_0 = \text{const}$ mamy do czynienia z oknem początkowym (rys.2.5a). Okno takie rozszerza się wraz z upływem czasu i jest najwłaściwsze dla analizy sygnałów stacjonarnych. W przypadku sygnału mowy należy adaptować predyktor do aktualnie wypowiedzanej głoski (poprzednie głoski winne być usunięte z okna). Można to osiągnąć wprowadzając okno początkowe z wagą wykładniczą (rys.2.5b) lub okno ruchome o stałej długości M (rys.2.5c).

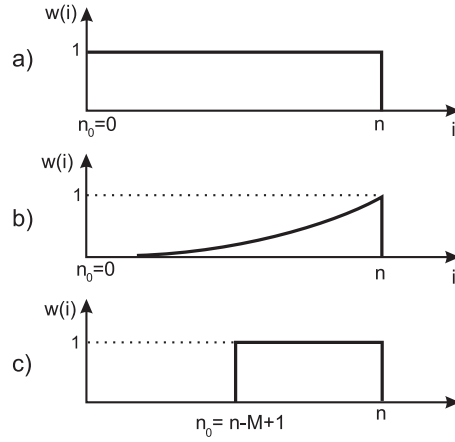
Algorytmy LS można podzielić na *autokorelacyjne* i *kowariancyjne* [90]. W algorytmach autokorelacyjnych dokonuje się minimalizacji energii błędu predykcji w obrębie całego okna $i = n_0, n_0 + 1, \dots, n$:

$$\mathcal{E}_f(n|p) = \sum_{i=n_0}^n \tilde{\epsilon}^2(i) = \sum_{i=n_0}^n [\tilde{x}(i) - \mathbf{a}^t(n) \tilde{\mathbf{x}}(i-1)]^2 \quad (2.36)$$

Ze względu na konieczność synchronicznego działania algorytmu w nadajniku i odbiorniku, używa się skwantowanych wartości $\tilde{\epsilon}(i)$ i $\tilde{x}(i)$. Stosując okno z wagą wykładniczą, otrzymuje się następujący wzór na energię $\mathcal{E}_f(n|p)$:

$$\mathcal{E}_f(n|p) = \sum_{i=n_0}^n \lambda^{n-i} \tilde{\epsilon}^2(i) \quad (2.37)$$

gdzie λ - współczynnik wagowy.



Rysunek 2.5: Okna $w(i)$, w obrębie których minimalizuje się sumę kwadratów błędów predykcji: a) okno początkowe, b) okno początkowe z wagą wykładniczą, c) okno ruchome o długości M

Przy $\tilde{\mathbf{x}}(i-1) = [\tilde{x}(i-1), \dots, \tilde{x}(i-p)]^t$, do obliczenia sygnału predykcji $\hat{x}(i) = \mathbf{a}^t(n)\tilde{\mathbf{x}}(i-1)$ dla $i = n_0, n_0+1, \dots, n_0+p-1$, a co za tym idzie, do wyznaczenia energii $\mathcal{E}_f(n|p)$, konieczne są próbki sygnału \tilde{x} leżące poza analizowanym oknem ($i < n_0$). Gdy próbki te są nieznanne, to (w metodzie autokorelacyjnej) przyjmuje się $\tilde{x}(i) = 0$, $i < n_0$.

W metodzie kowariancyjnej unika się tego założenia, minimalizując energię błędu predykcji w zawężonym zakresie $i = n_0 + p, n_0 + p + 1, \dots, n$:

$$\mathcal{E}_f(n|p) = \sum_{i=n_0+p}^n \tilde{\epsilon}^2(i) = \sum_{i=n_0+p}^n [\tilde{x}(i) - \mathbf{a}^t(n)\tilde{\mathbf{x}}(i-1)]^2 \quad (2.38)$$

Wyrażenie to nie zawiera wartości $\tilde{x}(i)$, $i < n_0$. Wielkość $\mathcal{E}_f(n|p)$ może być interpretowana jako kwadrat odległości euklidesowej dwóch wektorów: wektora $\mathbf{X}(n)$ próbek skwantowanego sygnału mowy i wektora $\hat{\mathbf{X}}(n)$ próbek sygnału predykcji. W metodzie autokorelacyjnej wektory mają postać:

$$\mathbf{X}(n) = [\tilde{x}(n), \tilde{x}(n-1), \dots, \tilde{x}(n_0)]^t \quad (2.39)$$

$$\begin{aligned} \hat{\mathbf{X}}(n) &= \begin{bmatrix} \hat{x}(n) \\ \hat{x}(n-1) \\ \vdots \\ \hat{x}(n_0+1) \\ \hat{x}(n_0) \end{bmatrix} = \begin{bmatrix} \tilde{\mathbf{x}}^t(n-1) \\ \tilde{\mathbf{x}}^t(n-2) \\ \vdots \\ \tilde{\mathbf{x}}^t(n_0) \\ \tilde{\mathbf{x}}^t(n_0-1) \end{bmatrix} \mathbf{a}(n) \\ &= \begin{bmatrix} \tilde{x}(n-1) & \tilde{x}(n-2) & \cdots & \tilde{x}(n-p) \\ \tilde{x}(n-2) & \tilde{x}(n-3) & \cdots & \tilde{x}(n-p-1) \\ \vdots & \vdots & & \vdots \\ \tilde{x}(n_0) & 0 & \cdots & 0 \\ 0 & 0 & \cdots & 0 \end{bmatrix} \mathbf{a}(n) = \mathbf{S}(n|p) \mathbf{a}(n) \quad (2.40) \end{aligned}$$

zaś w metodzie kowariancyjnej:

$$\mathbf{X}(n) = [\tilde{x}(n), \tilde{x}(n-1), \dots, \tilde{x}(n_0+p)]^t \quad (2.41)$$

$$\begin{aligned} \hat{\mathbf{X}}(n) &= \begin{bmatrix} \hat{x}(n) \\ \hat{x}(n-1) \\ \vdots \\ \hat{x}(n_0+p) \end{bmatrix} = \begin{bmatrix} \tilde{\mathbf{x}}^t(n-1) \\ \tilde{\mathbf{x}}^t(n-2) \\ \vdots \\ \tilde{\mathbf{x}}^t(n_0+p-1) \end{bmatrix} \mathbf{a}(n) \\ &= \begin{bmatrix} \tilde{x}(n-1) & \tilde{x}(n-2) & \cdots & \tilde{x}(n-p) \\ \tilde{x}(n-2) & \tilde{x}(n-3) & \cdots & \tilde{x}(n-p-1) \\ \vdots & \vdots & & \vdots \\ \tilde{x}(n_0+p-1) & \tilde{x}(n_0+p-2) & \cdots & \tilde{x}(n_0) \end{bmatrix} \mathbf{a}(n) \\ &= \mathbf{S}(n|p) \mathbf{a}(n) \end{aligned} \quad (2.42)$$

W obu przypadkach wektor próbek sygnału predykcji $\hat{\mathbf{X}}(n)$ jest kombinacją liniową kolumn macierzy $\mathbf{S}(n|p)$ występującej we wzorach (2.40), (2.42). Jak wiadomo, minimum odległości euklidesowej między $\mathbf{X}(n)$ i $\hat{\mathbf{X}}(n)$ zapewnia rzut ortogonalny wektora $\mathbf{X}(n)$ na przestrzeń rozpiętą na wektorach tworzących kolumny $\mathbf{S}(n|p)$. Wynika stąd, że optymalnym wektorem współczynników predykcji jest wektor [90]:

$$\mathbf{a}(n) = [\mathbf{S}^t(n|p) \mathbf{S}(n|p)]^{-1} \mathbf{S}^t(n|p) \mathbf{X}(n) \quad (2.43)$$

Analogiczne rozwiązanie otrzymuje się dla okna z wagą wykładniczą, należy tylko każdą składową wektora $\mathbf{X}(n)$ i każdy wiersz macierzy $\mathbf{S}(n|p)$ pomnożyć przez odpowiednią wagę λ^{n-i} , $i = 0, 1, \dots$

Obliczenie współczynników $\mathbf{a}(n)$ byłoby znacznie prostsze, gdyby kolumny macierzy

$$\mathbf{S}(n|p) = [\mathbf{X}(n|1), \dots, \mathbf{X}(n|p)]$$

były ortogonalne. Wówczas m -ty współczynnik predykcji $a_m(n)$ byłby współczynnikiem rzutu ortogonalnego wektora $\mathbf{X}(n)$ na wektor $\mathbf{X}(n|m)$:

$$a_m(n) = \frac{\mathbf{X}^t(n|m) \mathbf{X}(n)}{\mathbf{X}^t(n|m) \mathbf{X}(n|m)} \quad (2.44)$$

Z reguły macierz $\mathbf{S}(n|p)$ nie jest ortogonalna. Można ją zortogonalizować stosując procedurę Grama-Schmidta. Wektor $\mathbf{X}(n|1)$ pozostawia się bez zmiany, natomiast od kolejnych wektorów $\mathbf{X}(n|m)$ odejmuje się ich rzut na podprzestrzeń rozpiętą na wektorach $\mathbf{X}(n|1)$, $\mathbf{X}(n|2)$, \dots , $\mathbf{X}(n|m-1)$. Otrzymuje się w ten sposób bazę ortogonalną $\mathbf{X}_{ort}(n|1)$, \dots , $\mathbf{X}_{ort}(n|p)$. W niektórych przypadkach procedura ortogonalizacyjna może być przeprowadzona dla struktury kratowej jak na rys.2.4b [189]. Metoda autokorelacyjna z oknem początkowym i metoda kowariancyjna z oknem ruchomym o stałej długości należą do tego typu przypadków. Sygnały występujące na wyjściu układów opóźniających (z^{-i}) tworzą wówczas bazę ortogonalną

$\mathbf{X}_{ort}(n|1), \dots, \mathbf{X}_{ort}(n|p)$. Wartości $k_1^b(n), \dots, k_p^b(n)$ są współczynnikami rzutu ortogonalnego wektora $\mathbf{X}(n)$ na wektory tej bazy. Wektor próbek sygnału predykcji $\hat{\mathbf{X}}(n)$ jest sumą poszczególnych rzutów ortogonalnych. Ma on tę samą wartość jak w przypadku predyktora transwersalnego. Wynika stąd równoważność struktury transwersalnej i struktury kratowej (dla metody autokorelacyjnej z oknem początkowym i metody kowariancyjnej z oknem ruchomym). Ze względu na tę równoważność rozważać będziemy dalej jedynie strukturę kratową. Wybór tej właśnie struktury można uzasadnić łatwiejszą kontrolą stabilności, o czym była mowa w p.2.2.2.

Szybkie algorytmy rekursywne

Współczynniki opisujące strukturę kratową predyktora mogą być wyznaczone drogą działań na wektorze $\mathbf{X}(n)$ i kolumnach macierzy $\mathbf{S}(n|p)$. Wymaga to jednak wykonania wielu operacji arytmetycznych w każdej chwili n . Zauważmy jednak, że wektor $\mathbf{X}(n)$ i macierz $\mathbf{S}(n|p)$ różnią się w niewielkim stopniu od ich odpowiedników $\mathbf{X}(n-1)$ i $\mathbf{S}(n-1|p)$. Różnica polega na tym, że do wektora $\mathbf{X}(n-1)$ dopisana jest dodatkowa pierwsza składowa $\tilde{x}(n)$, a do macierzy $\mathbf{S}(n-1|p)$ dodatkowy pierwszy wiersz $[\tilde{x}(n-1), \tilde{x}(n-2), \dots, \tilde{x}(n-p)]$. Poza tym, w metodzie kowariancyjnej z oknem ruchomym, skasowana jest ostatnia składowa $\tilde{x}(n_0 + p - 1)$ wektora $\mathbf{X}(n-1)$ i ostatni wiersz $[\tilde{x}(n_0 + p - 2), \tilde{x}(n_0 + p - 3), \dots, \tilde{x}(n_0 - 1)]$ macierzy $\mathbf{S}(n-1|p)$. Można zatem oczekiwać, że między współczynnikami predykcji w chwilach $n-1$ i n istnieje pewien związek. Wykorzystując ten związek można wyprowadzić tzw. *szybki algorytm rekursywny najmniejszych kwadratów* (wyprowadzenie można znaleźć w [90]). Wymaga on wykonania w chwili n obliczenia następujących rekursji (dla $m = 0, \dots, p-1$):

$$K_{m+1}(n) = \lambda K_{m+1}(n-1) + e_f(n|m) e_b(n-1|m) \frac{1}{1 - \gamma(n-1|m)} \quad (2.45)$$

$$k_{m+1}^f(n) = \frac{K_{m+1}(n)}{\mathcal{E}_f(n|m)} \quad (2.46)$$

$$k_{m+1}^b(n) = \frac{K_{m+1}(n)}{\mathcal{E}_b(n-1|m)} \quad (2.47)$$

$$e_f(n|m+1) = e_f(n|m) - k_{m+1}^b(n) e_b(n-1|m) \quad (2.48)$$

$$e_b(n|m+1) = e_b(n-1|m) - k_{m+1}^f(n) e_f(n|m) \quad (2.49)$$

$$\mathcal{E}_f(n|m+1) = \mathcal{E}_f(n|m) - k_{m+1}^b(n) K_{m+1}(n) \quad (2.50)$$

$$\mathcal{E}_b(n|m+1) = \mathcal{E}_b(n-1|m) - k_{m+1}^f(n) K_{m+1}(n) \quad (2.51)$$

$$\gamma(n|m+1) = \gamma(n|m) + \frac{e_b^2(n|m)}{\mathcal{E}_b(n|m)} \quad (2.52)$$

$$e_f^*(n|m+1) = e_f^*(n|m) - k_{m+1}^b(n) e_b^*(n-1|m) \quad (2.53)$$

$$e_b^*(n|m+1) = e_b^*(n-1|m) - k_{m+1}^f(n) e_f^*(n|m) \quad (2.54)$$

$$\gamma^*(n|m+1) = \gamma^*(n|m) + \frac{e_b^{*2}(n|m)}{\mathcal{E}_b(n|m)} \quad (2.55)$$

$$K_{m+1}(n) \Leftarrow K_{m+1}(n) - e_f^*(n|m) e_b^*(n-1|m) \frac{1}{1 - \gamma^*(n-1|m)} \quad (2.56)$$

Pierwsza część wzorów (2.45 - 2.52) opisuje metodę najmniejszych kwadratów z oknem początkowym ($\lambda < 1$ jest wagą wykładniczą). Obie części opisują algorytm z oknem ruchomym (należy wówczas podstawić $\lambda = 1$). Znak \Leftarrow w (2.56) oznacza podstawienie nowej wartości $K_{m+1}(n)$ po zmodyfikowaniu dotychczasowej wartości występującej po prawej stronie równania.

Wyjaśnijmy znaczenie niektórych zmiennych występujących w szybkim algorytmie najmniejszych kwadratów. Kolejne próbki sygnału $e_b(n-1|m)$, $e_b(n-2|m)$, ... tworzą wektor $\mathbf{E}_b(n|m+1) = \mathbf{X}_{ort}(n|m+1)$ tzn. $m+1$ -szy wektor zortogonalizowanej bazy $\mathbf{X}_{ort}(n|1)$, ..., $\mathbf{X}_{ort}(n|p)$. Kolejne próbki sygnału $e_f(n|m+1)$, $e_f(n-1|m+1)$, ... tworzą współrzędne wektora błędu predykcji $\mathbf{E}_f(n|m+1)$ dla predyktora o $m+1$ współczynnikach. Wektor $\mathbf{E}_f(n|m+1)$ powstaje przez odjęcie od wektora $\mathbf{X}(n)$ jego rzutu na podprzestrzeń rozpiętą na wektorach będących $m+1$ pierwszymi kolumnami macierzy $\mathbf{S}(n|p)$. Rzut ten stanowi wektor próbek sygnału predykcji $\hat{\mathbf{X}}(n|m+1)$ uzyskany przy użyciu predyktora o $m+1$ współczynnikach. Wektor ten jest obliczany jako suma rzutów ortogonalnych wektora $\mathbf{X}(n)$ na kolejne wektory $\mathbf{X}_{ort}(n|1)$, ..., $\mathbf{X}_{ort}(n|m+1)$ czyli $\mathbf{E}_b(n|1)$, ..., $\mathbf{E}_b(n|m+1)$:

$$\hat{\mathbf{X}}(n|m+1) = \sum_{i=1}^{m+1} k_i^b(n) \mathbf{E}_b(n|i) \quad (2.57)$$

Pierwsza współrzędna tego wektora jest (dla $m = p-1$) poszukiwaną próbką sygnału predykcji $\hat{x}(n)$:

$$\hat{x}(n) = \sum_{i=1}^p k_i^b(n) e_b(n-1|i-1) \quad (2.58)$$

Współczynnik $k_{m+1}^b(n)$ jest współczynnikiem rzutu wektora $\mathbf{X}(n)$ na $\mathbf{X}_{ort}(n|m+1) = \mathbf{E}_b(n|m+1)$:

$$k_{m+1}^b(n) = \frac{\mathbf{E}_b^t(n|m+1) \mathbf{X}(n)}{\mathbf{E}_b^t(n|m+1) \mathbf{E}_b(n|m+1)} = \frac{\mathbf{E}_b^t(n|m+1) \mathbf{E}_f(n|m)}{\mathbf{E}_b^t(n|m+1) \mathbf{E}_b(n|m+1)} \quad (2.59)$$

Ostatnia równość wynika z faktu, że wektor $\mathbf{X}(n)$ można rozłożyć na składniki ortogonalne: $\mathbf{X}(n) = \hat{\mathbf{X}}(n|m) + \mathbf{E}_f(n|m)$, gdzie $\hat{\mathbf{X}}(n|m)$ leży w podprzestrzeni rozpiętej na wektorach $\mathbf{X}_{ort}(n|1)$, ..., $\mathbf{X}_{ort}(n|m)$, a $\mathbf{E}_f(n|m)$ jest do tej podprzestrzeni ortogonalny. W

związku z tym rzut wektora $\mathbf{X}(n)$ na $\mathbf{X}_{ort}(n|m+1)$ jest równy rzutowi wektora $\mathbf{E}_f(n|m)$ na $\mathbf{X}_{ort}(n|m+1)$.

Porównując wzór (2.59) ze wzorem definicyjnym (2.47) można zinterpretować $K_{m+1}(n)$ jako iloczyn skalarny $\mathbf{E}_b^t(n|m+1) \mathbf{E}_f(n|m)$, oraz $\mathcal{E}_b(n-1|m)$ jako kwadrat normy $\|\mathbf{E}_b(n|m+1)\|^2 = \mathbf{E}_b^t(n|m+1) \mathbf{E}_b(n|m+1)$. Podobnie $\mathcal{E}_f(n|m)$ jest interpretowane jako kwadrat normy wektora $\mathbf{E}_f(n|m)$.

Z nierówności Schwarzera

$$\left[\mathbf{E}_b^t(n|m+1) \mathbf{E}_f(n|m) \right]^2 \leq \|\mathbf{E}_f(n|m)\|^2 \|\mathbf{E}_b(n|m+1)\|^2 \quad (2.60)$$

wynika, że

$$K_{m+1}^2(n) \leq \mathcal{E}_f(n|m) \mathcal{E}_b(n-1|m) \quad (2.61)$$

Naruszenie tej nierówności (np. wskutek niedokładności obliczeń) powoduje, że wartości numeryczne norm wektorów występujących we wzorach (2.50), (2.51) stają się ujemne.

Jak już wspomniano, w chwili n dokonuje się modyfikacji macierzy $\mathbf{S}(n-1|p)$ dopisując do niej nowy pierwszy wiersz i dodatkowo kasując (dla metody kowariancyjnej z oknem ruchomym) ostatni wiersz w taki sposób, aby otrzymać macierz $\mathbf{S}(n|p)$. Współczynnik $\gamma(n-1|m)$ interpretuje się jako kwadrat sinusa kąta pomiędzy podprzestrzeniami rozpiętymi na $m+1$ kolumnach macierzy $\mathbf{S}(n-1|p)$ z dopisanym nowym wierszem i bez tego wiersza. Podobnie $\gamma^*(n-1|m)$ interpretuje się w odniesieniu do usuwanego ostatniego wiersza macierzy $\mathbf{S}(n-1|p)$. Wartości tych współczynników powinny być zawarte w przedziale $(0, 1)$.

W algorytmie kowariancyjnym z oknem ruchomym występują ponadto sygnały $e_f^*(n|m)$ i $e_b^*(n-1|m)$ reprezentujące "lewą" krawędź okna (rys.2.5c).

W chwili n obliczenia rozpoczyna się wyznaczeniem sygnału predykcji $\hat{x}(n)$ ze wzoru (2.58). Po otrzymaniu sygnału skwantowanego $\tilde{x}(n)$ (2.1) możliwa jest inicjalizacja wartości zmiennych niezbędnych do obliczenia rekursji (2.45 - 2.56). Zmiennym $e_f(n|0)$ i $e_b(n|0)$ nadaje się wartości początkowe równe $\tilde{x}(n)$. Podobnie podstawia się $e_f^*(n|0) = e_b(n|0) = \tilde{x}(n_0+p) = \tilde{x}(n-M+p+1)$, gdzie M jest szerokością okna (rys.2.5c). Wartości $\mathcal{E}_b(n|0) = \mathcal{E}_f(n|0)$ są kwadratem normy wektora $\mathbf{X}(n)$. Oblicza się je, dodając do kwadratu normy wektora $\mathbf{X}(n-1)$ wartość $\tilde{x}^2(n)$ i odejmując $\tilde{x}^2(n_0+p-1)$. Dla algorytmu z oknem początkowym i wagą wykładniczą mnoży się $\|\mathbf{X}(n-1)\|^2 = \mathcal{E}_f(n-1|0)$ przez λ i dodaje się $\tilde{x}^2(n)$. Współczynniki γ i γ^* dla $m=0$ są zerowe.

Inicjalizacja całego algorytmu polega na zastosowaniu w chwili n , $0 < n < p-1$ predyktora o $m=n$ współczynnikach. Dla $n \geq p$ stosuje się predyktor o p współczynnikach. Wszystkie sygnały dla $n < 0$ uważa się za zerowe.

2.2.5 Wyniki badań symulacyjnych

Warunki i zakres badań

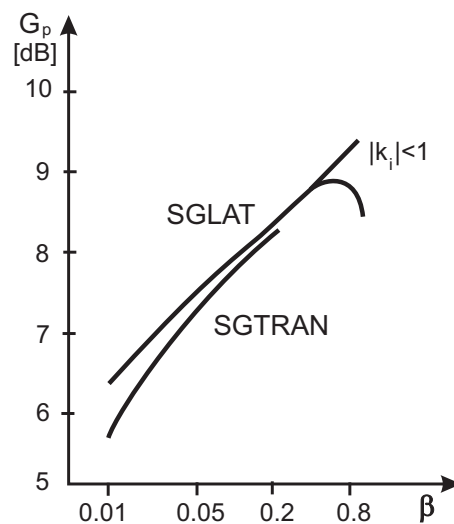
Badania symulacyjne wykonywane były na sygnale mowy. Wykorzystywano 3 frazy sygnału mowy, spróbkowanego z częstotliwością 8 kHz i przetworzonego na postać cyfrową w forma-

cie 12-bitowym (w sumie około 10 s materiału dźwiękowego). Badano następujące metody sekwencyjne predykcji liniowej:

- **SGTRAN**: algorytm gradientu stochastycznego, predyktor transwersalny,
- **SGTRAN-N**: algorytm gradientu stochastycznego z normalizacją, predyktor transwersalny,
- **SGLAT**: algorytm gradientu stochastycznego, predyktor kratowy,
- **SGLAT-N**: algorytm gradientu stochastycznego z normalizacją, predyktor kratowy,
- **LSLAT**: algorytm najmniejszych kwadratów z oknem wykładniczym, predyktor kratowy,
- **COVLAT**: algorytm kowariancyjny z oknem ruchomym, predyktor kratowy.

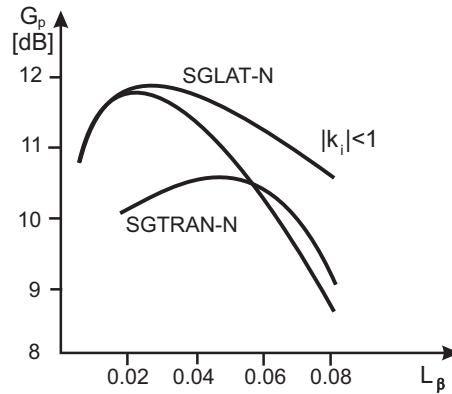
Badanie zysku predykcji

W pierwszym etapie badań rejestrowano zysk predykcji G_p w ujęciu segmentowym dla samego predyktora z adaptacją (z układu ADPCM wyeliminowano kwantyzator). Umożliwiło to dobór optymalnych parametrów predyktora i algorytmu jego adaptacji.



Rysunek 2.6: Zysk predykcji w funkcji parametru β , ($p = 8$)

W metodach gradientu stochastycznego SGTRAN i SGLAT najistotniejszą sprawą jest właściwy dobór stałej β określającej szybkość zbieżności. Dla zbyt małych wartości β szybkość zbieżności jest mała - objawia się to spadkiem zysku predykcji G_p (rys.2.6). Dla zbyt dużych wartości β następuje utrata stabilności algorytmu adaptacji - szczególnie w obrębie fragmentów sygnału mowy o wysokim poziomie.

Rysunek 2.7: Zysk predykcji w funkcji parametru L_β , ($p = 8$)

W metodzie SGLAT można ograniczyć skutki utraty stabilności algorytmu adaptacji wymuszając warunek $|k_i| < 1$, $i = 1, \dots, p$ (jest to jednocześnie warunek stabilności układu ADPCM z predyktorem w pętli sprzężenia zwrotnego). Ostatecznie ustalono następujące wartości stałej β : dla algorytmu SGTRAN $\beta = 0.2$, dla algorytmu SGLAT $\beta = 0.4$ (zakłada się, że próbki przetwarzanego sygnału mowy mieszczą się w przedziale od -1 do +1).

W metodach gradientu stochastycznego z normalizacją najistotniejszy jest wybór wartości L_β - stałej określającej szybkość zbieżności i stabilność algorytmu. Działa tu podobny mechanizm, jak w przypadku metod bez normalizacji (rys.2.7). Ostatecznie ustalono $L_\beta = 0.04$ dla algorytmu SGTRAN-N i $L_\beta = 0.02$ dla algorytmu SGLAT-N. Parametr M_β ma niewielki wpływ na zysk predykcji. W obliczeniach stosowano $M_\beta = 0.0001$. Parametr $\bar{\alpha}$, od którego zależy stała czasowa estymatora mocy sygnału (wzór 2.25), ma również niewielki wpływ na zysk predykcji - można go zmieniać w zakresie 0.8 - 0.98. Ostatecznie przyjęto $\bar{\alpha} = 0.92$ dla algorytmu SGTRAN-N i $\bar{\alpha} = 0.96$ dla algorytmu SGLAT-N.

W metodzie najmniejszych kwadratów z oknem początkowym i wagą wykładniczą (LSLAT) należy wybrać optymalną wartość λ , charakteryzującą efektywną długość wykładniczego okna. W wyniku symulacji otrzymano optymalną wartość $\lambda = 0.98$ (rys.2.8).

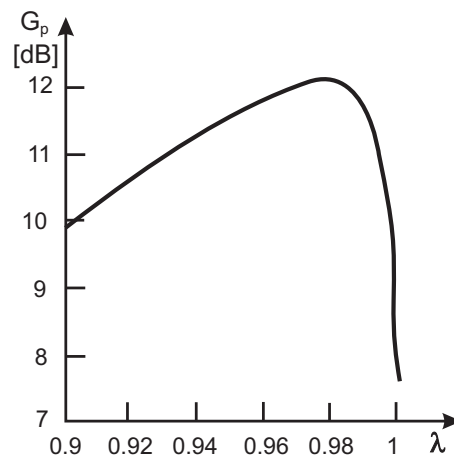
W metodzie kowariancyjnej z oknem ruchomym (COVLAT) dla niewielkich fragmentów mowy uzyskiwano duże wartości G_p około 16 dB. Symulacje z użyciem całych fraz mowy nie dały jednak zadowalających wyników ze względu na pojawiające się przypadki utraty stabilności algorytmu adaptacji predyktora. Szczegółowa analiza dowiodła, że występuje tu zjawisko kumulacji błędów zaokrągleń.

W algorytmie adaptacji predyktora (2.45 - 2.56) występują następujące zmienne, w których mogą kumulować się błędy zaokrągleń: $K_{m+1}(n)$, $m = 0, \dots, p-1$, oraz $\mathcal{E}_b(n|0)$ i $\mathcal{E}_f(n|0)$. W przypadku braku błędów zaokrągleń spełnione są równości

$$K_{m+1}(n) = \mathbf{E}_b^t(n|m+1) \mathbf{E}_f(n|m), \quad m = 0, \dots, p-1,$$

oraz

$$\mathcal{E}_b(n|0) = \mathcal{E}_f(n|0) = \|\mathbf{X}(n)\|^2.$$

Rysunek 2.8: Zysk predykcji w funkcji parametru λ , ($p = 8$)

Po przetworzeniu dłuższego fragmentu sygnału mowy błędy zaokrągleń osiągają tak wysoki poziom, że wyżej wymienione równości nie są już spełnione. Daje się to zauważyć szczególnie w sytuacji, gdy przetwarzany jest fragment ciszy międzywyrazowej i wartości $\mathbf{E}_b^t(n|m+1)$, $\mathbf{E}_f(n|m)$, oraz $\|\mathbf{X}(n)\|^2$ są małe. W tych warunkach często dochodzi do naruszenia nierówności (2.61), co powoduje otrzymanie ujemnych norm wektorów w kolejnych iteracjach i utratę stabilności numerycznej całego algorytmu adaptacji. Ograniczenie wartości $K_{m+1}(n)$, aby wymusić spełnienie nierówności (2.61), nie jest działaniem skutecznym, gdyż obliczane współczynniki predykcji tracą związek z przetwarzanym sygnałem mowy (zależą od skumulowanych błędów zaokrągleń).

Skutecznym działaniem okazał się natomiast restart całego algorytmu adaptacji po każdorazowym naruszeniu nierówności (2.61). Zmienne, zawierające skumulowane błędy zaokrągleń, są zerowane, a cały sygnał $\tilde{x}(n)$ poprzedzający moment restartu jest traktowany jako zerowy.

Po wprowadzeniu powyższych modyfikacji zbadano wpływ długości okna M na zysk predykcji. Najlepsze wyniki uzyskano dla okna liczącego $M = 100$ próbek (rys.2.9).

Po ustaleniu optymalnych wartości parametrów dla każdej z wymienionych metod adaptacji predyktora, zbadano zysk predykcji w funkcji liczby współczynników predykcji p (rys.2.10). Najlepszym okazał się algorytm COVLAT, jednak algorytmy LSLAT i SGLAT-N ustępują mu jedynie w niewielkim stopniu. Zależność zysku predykcji G_p od liczby współczynników predykcji p wykazuje szybkie nasycenie. Ostatecznie przyjęto $p = 8$ jako kompromis między jakością predyktora, a liczbą operacji arytmetycznych wykonywanych w procesie adaptacji.

Badanie układu ADPCM

Po ustaleniu optymalnych wartości parametrów dla układów adaptacji predyktora, zasymulowano układ ADPCM z kwantyzatorami adaptacyjnymi o 4, 8, i 16 poziomach kwantyzacji.

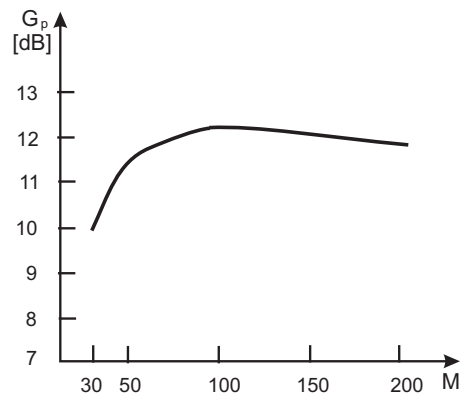
Rysunek 2.9: Zysk predykcji w funkcji długości okna M , ($p = 8$)

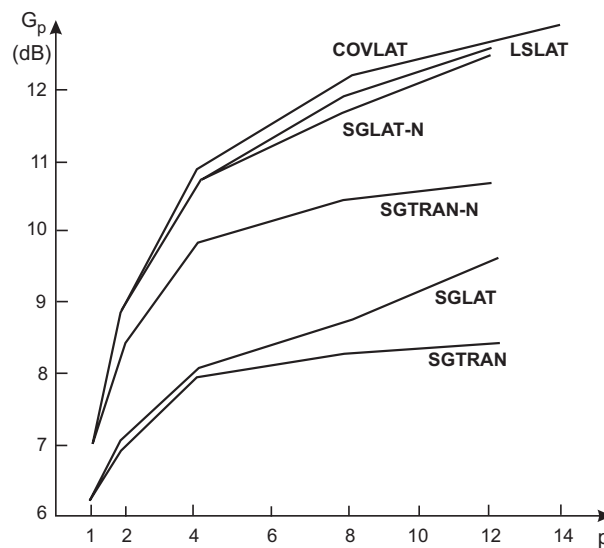
Tabela 2.7: Stosunek mocy sygnału użytecznego do mocy zniekształceń (SNR [dB]) w ujęciu segmentowym dla symulowanego układu ADPCM

Algorytm	Liczba poz. kwantyzacji L		
	4	8	16
bez predykcji	8.91	14.68	19.91
SGTRAN	14.59	20.98	24.28
SGLAT	14.69	21.14	24.21
SGTRAN-N	15.57	21.65	24.37
SGLAT-N	16.35	22.89	25.04
LSLAT	16.31	22.76	25.10
COVLAT	16.69	23.03	25.37

cji, co odpowiada szybkości transmisji 16, 24 i 32 kbit/s. Stosowano predyktory o $p = 8$ współczynnikach. Z analizy wyników (tabl.2.7) można wnioskować o opłacalności stosowania predyktorów (poprawa jakości mowy o 3 - 9 dB w porównaniu z układem bez predykcji). Najlepsze wyniki zapewnia metoda COVLAT, jednak metody LSLAT i SGLAT-N prawie jej dorównują. Ogólnie biorąc, predyktory o strukturze kratowej zapewniają lepsze działanie układu ADPCM niż predyktory transwersalne.

2.2.6 Podsumowanie

Celem podrozdziału było dokonanie analizy porównawczej algorytmów predykcji liniowej pod kątem ich przydatności do implementacji w koderach ADPCM, szczególnie z wykorzystaniem procesorów sygnałowych. Podobne badania przeprowadzono już w [91],[168]. W niniejszej pracy problem potraktowano szerzej, badając szerszą klasę algorytmów predykcji liniowej (m.in. algorytmy kowariancyjne z oknem ruchomym) oraz układy ADPCM o różnej szybkości

Rysunek 2.10: Zysk predykcji w funkcji liczby współczynników predykcji p

transmisji: 16, 24 i 32 kbit/s.

Najlepsze właściwości, z punktu widzenia zysku predykcji, mają algorytmy najmniejszych kwadratów. Algorytm z oknem ruchomym w swojej wersji oryginalnej [90] nie nadaje się do przetwarzania sygnału mowy ze względu na kumulację błędów zaokrągleń. Po modyfikacji, omówionej w p.2.2.5, algorytm ten okazał się najlepszy ze względu na zysk predykcji i wartość stosunku sygnał - szum w układzie ADPCM. Algorytm z oknem początkowym i wagą wykładniczą jest odporny na kumulację błędów zaokrągleń (w każdym momencie n , skumulowane błędy maleją w stosunku λ) i niewiele ustępuje algorytmowi z oknem ruchomym.

Metody gradientowe z normalizacją, zastosowane w predyktorach o strukturze kratowej, niewiele ustępują metodom najmniejszych kwadratów. Metody gradientowe są ciągle jeszcze udoskonalane. Proponowane są warianty bardziej złożone obliczeniowo, lecz szybciej zbieżne [26].

Ostateczne decyzje co do implementacji poszczególnych algorytmów predykcji liniowej w koderach ADPCM zależą od techniki realizacji tych układów. Metody gradientu stochastycznego w wersji transwersalnej są najmniej złożone obliczeniowo (liczba mnożeń i dzielení w każdej iteracji czasowej wynosi około $2p$ dla algorytmu SGTRAN i $3p$ dla algorytmu SGTRAN-N). Możliwa jest ich implementacja na procesorach sygnałowych starszej generacji (np. TMS 32010). Metody gradientu stochastycznego oraz metody najmniejszych kwadratów w wersji kratowej są bardziej złożone obliczeniowo (liczba mnożeń i dzielení na iterację czasową wynosi około $10p$ dla algorytmu SGLAT, $13p$ dla algorytmu SGLAT-N, $11p$ dla algorytmu LSLAT i $16p$ dla algorytmu COVLAT). Możliwa jest ich implementacja na procesorach sygnałowych typu TMS 320C25, TMS 320C30. Wydaje się, że predyktor kratowy z adaptacją metodą najmniejszych kwadratów, z oknem początkowym i wagą wykładniczą (LSLAT) jest szczególnie godny polecenia ze względu na duży zysk predykcji i jednorodną strukturę algorytmu (brak restartów).

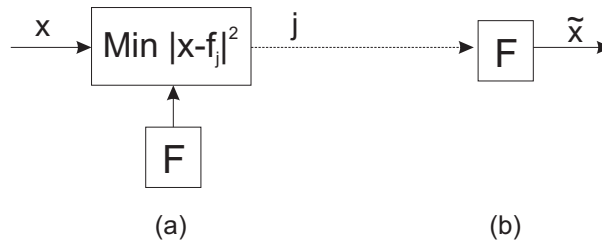
Rozdział 3

Kwantyzatory wektorowe i kodery CELP

3.1 Kwantyzacja wektorowa

3.1.1 Zasada działania kwantyzatora wektorowego

Kwantyzator wektorowy (VQ) jest uogólnieniem kwantyzatora skalarnego. Kwantowaniu podlega N -wymiarowy wektor $\mathbf{x}(i)$, złożony z N kolejnych próbek sygnału (indeks i oznacza numer kolejny wektora). Rolę poziomów kwantyzacji odgrywają wektory $\mathbf{f}_1, \mathbf{f}_2 \dots \mathbf{f}_L$, tworzące tzw. słownik \mathbf{F} (książkę kodową) kwantyzatora wektorowego. Przesyła się tylko numer wybranego wektora (j), i na tej podstawie odtwarza się N kolejnych próbek sygnału wyjściowego, tworzącego wektor $\tilde{\mathbf{x}}(i)$ - rys.3.1.



Rysunek 3.1: Kwantyzator wektorowy w postaci podstawowej: (a) - koder, (b) - dekoder

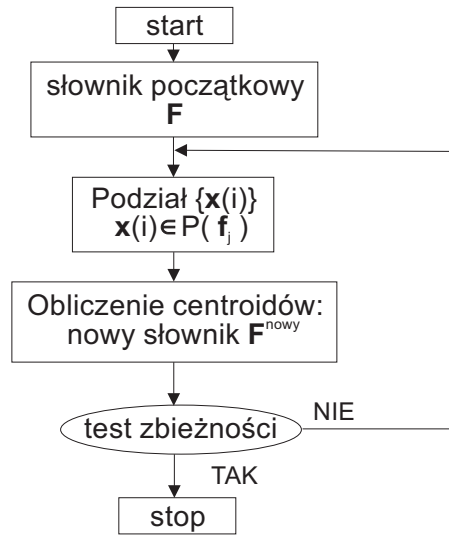
Kwantyzator wektorowy jest określony, jeżeli znany jest słownik \mathbf{F} i algorytm jego przeszukiwania (tzn. sposób wyboru wektora $\tilde{\mathbf{x}}(i)$ spośród wektorów słownika). Najczęściej przeszukuje się cały słownik (ang. *full search*) i wybiera się wektor \mathbf{f}_j zapewniający minimum odległości euklidesowej $\|\mathbf{x}(i) - \mathbf{f}_j\|$. Kryterium euklidesowe nie jest wybrane przypadkowo - $\|\mathbf{x}(i) - \tilde{\mathbf{x}}(i)\|^2$ jest energią błędu kwantyzacji N kolejnych próbek sygnału wejściowego. Algorytm przeszukiwania słownika generuje podział przestrzeni R^N na L tzw. komórek Voronoi'a $P(\mathbf{f}_1), \dots, P(\mathbf{f}_L)$, określonych w następujący sposób:

$$P(\mathbf{f}_j) = \{\mathbf{x} \in R^N : \tilde{\mathbf{x}} = \mathbf{f}_j\} \quad (3.1)$$

Każdy wektor słownika \mathbf{f}_j jest centroidem odpowiedniej komórki Voronoi'a $P(\mathbf{f}_j)$.

Przepływność binarną determinuje tzw. rozdzielczość b - liczba bitów przypadająca na próbkę sygnału. Rozdzielczość jest często liczbą ułamkową, nawet mniejszą od 1. Liczba bitów przeznaczona do zakodowania wektora sygnału bN jest najczęściej liczbą całkowitą, określającą wielkość słownika ($L = 2^{bN}$ wektorów).

Projektowanie optymalnych kwantyzatorów wektorowych (tzn. wyznaczenie słowników) odbywa się w podobny sposób, jak projektowanie optymalnych kwantyzatorów skalarnych. Najczęściej korzysta się z uogólnionego na przypadek wielowymiarowy, iteracyjnego algorytmu Lloyda [122], przedstawionego na rys.3.2. Mając dany zbiór wektorów sygnału akustycznego (tzw. sekwencję treningową) $\{\mathbf{x}(i)\}$, oraz słownik początkowy $\mathbf{F} = \{\mathbf{f}_j\}$ dokonuje się kwantyzacji sekwencji treningowej. Oznacza to podział sekwencji treningowej na L części, należących do poszczególnych komórek Voronoi'a (wektor $\mathbf{x}(i)$ należy do $P(\mathbf{f}_j)$, gdy



Rysunek 3.2: Uogólniony algorytm Lloyd'a projektowania słownika kwantyzatora wektorowego

$\| \mathbf{x}(i) - \mathbf{f}_j \| \leq \| \mathbf{x}(i) - \mathbf{f}_k \|$ dla $k = 1, 2, \dots, L$). W wyniku kwantowania sekwencji treningowej otrzymuje się energię błędu kwantyzacji:

$$D = \sum_i \| \mathbf{x}(i) - \tilde{\mathbf{x}}(i) \|^2 = \sum_{j=1}^L \sum_{i, \mathbf{x}(i) \in P(\mathbf{f}_j)} \| \mathbf{x}(i) - \mathbf{f}_j \|^2 = \sum_{j=1}^L D_j \quad (3.2)$$

gdzie

$$D_j = \sum_{i, \mathbf{x}(i) \in P(\mathbf{f}_j)} \| \mathbf{x}(i) - \mathbf{f}_j \|^2 = \sum_{i, \mathbf{x}(i) \in P(\mathbf{f}_j)} [\mathbf{x}^t(i) \mathbf{x}(i) - 2\mathbf{f}_j^t \mathbf{x}(i) + \mathbf{f}_j^t \mathbf{f}_j] \quad (3.3)$$

jest energią błędu kwantyzacji dla wektorów należących do j -tej komórki.

Nowy słownik $\mathbf{F}^{nowy} = \{\mathbf{f}_j^{nowy}\}$ oblicza się w taki sposób, aby zminimalizować D zachowując dotychczasowy podział przestrzeni R^N na komórki Voronoi'a. Z równania $\frac{\partial D}{\partial \mathbf{f}_j} = \frac{\partial D_j}{\partial \mathbf{f}_j} = \mathbf{0}$ otrzymuje się:

$$\mathbf{f}_j^{nowy} = \frac{1}{m_j} \sum_{i, \mathbf{x}(i) \in P(\mathbf{f}_j)} \mathbf{x}(i) \quad (3.4)$$

gdzie m_j jest liczbą wektorów sekwencji treningowej należącą do j -tej komórki. Wzór (3.4) określa centroid - nowy wektor słownika jest wartością średnią wektorów treningowych należących do odpowiedniej komórki.

Nowy słownik generuje nowy podział przestrzeni na komórki Voronoi'a i proces obliczania centroidów powtarza się aż do osiągnięcia punktu stałego - rys.3.2.

Sabin i Gray [175] badali warunki zbieżności tego algorytmu. W praktyce duże znaczenie ma wybór słownika początkowego. Problem wyznaczania warunków początkowych dla procedury Lloyd'a pomyślnie rozwiązali Linde, Buzo i Gray [121], tworząc powszechnie stosowany algorytm LBG (nie daje on jednak gwarancji zbieżności do minimum globalnego dla D).

3.1.2 Wartości graniczne dla kwantyzatora wektorowego

Założmy że znamy gęstość prawdopodobieństwa N - wymiarowych wektorów sygnału $p(\mathbf{x})$. O kwantyzatorze założmy, że ma on dużą rozdzielczość b , a więc słownik o dużej liczbie wektorów $L = 2^{bN}$. Oznacza to, że gęstość prawdopodobieństwa $p(\mathbf{x})$ może być zastąpiona wartością stałą w obrębie każdej komórki Voronoi'a. Można więc uprościć ogólny wzór opisujący moc szumu kwantowania:

$$\sigma_e^2(b, N) = \frac{1}{N} \int_{R^N} \|\mathbf{x} - \tilde{\mathbf{x}}\|^2 d\mathbf{x} = \frac{1}{N} \sum_{j=1}^L \int_{\mathbf{x} \in P(\mathbf{f}_j)} \|\mathbf{x} - \mathbf{f}_j\|^2 d\mathbf{x} \quad (3.5)$$

Dla optymalnego kwantyzatora wektorowego Zador [208] otrzymał następujące oszacowanie:

$$\sigma_e^2(b, N) = \bar{c}(N) \|p(\mathbf{x})\|_{\frac{N}{N+2}} 2^{-2b} \quad (3.6)$$

gdzie $\|p(\mathbf{x})\|_{\alpha} = [\int_{R^N} p^{\alpha}(\mathbf{x}) d\mathbf{x}]^{1/\alpha}$, natomiast $\bar{c}(N)$ jest stałą zależną od wymiaru wektora. Wartości $\bar{c}(N)$ obliczył Gersho [74].

Dla sygnału o rozkładzie gaussowskim i zerowej wartości średniej:

$$p(\mathbf{x}) = \frac{1}{(2\pi)^{N/2} \sqrt{\det \mathbf{R}^N(\mathbf{x})}} \exp \left[-\frac{1}{2} \mathbf{x}^t (\mathbf{R}^N(\mathbf{x}))^{-1} \mathbf{x} \right] \quad (3.7)$$

gdzie $\mathbf{R}^N(\mathbf{x}) = E[\mathbf{x}\mathbf{x}^t]$ jest macierzą kowariancji¹, wzór (3.6) daje się uprościć do następującej postaci [124]:

$$\sigma_e^2(b, N) = c(N) [\det \mathbf{R}^N(\mathbf{x})]^{1/N} 2^{-2b} \quad (3.8)$$

Stała $c(1) = \frac{\sqrt{3}\pi}{2}$, natomiast dla większych wartości N , $c(N)$ maleje dążąc do jedności.

Podstawiając $\mathbf{R}^N(\mathbf{x}) = \sigma_x^2 \rho^N(\mathbf{x})$, gdzie $\rho^N(\mathbf{x})$ jest znormalizowaną macierzą kowariancji (a zarazem korelacji), otrzymuje się następujący wzór na stosunek mocy sygnału do mocy szumu kwantyzacji:

$$SNR(b, N) = \frac{\sigma_x^2}{\sigma_e^2(b, N)} = c^{-1}(N) \frac{\sigma_x^2}{[\det \mathbf{R}^N(\mathbf{x})]^{1/N}} 2^{2b} = c^{-1}(N) \frac{1}{[\det \rho^N(\mathbf{x})]^{1/N}} 2^{2b} \quad (3.9)$$

Dla $N \rightarrow \infty$ wzór (3.9) staje się identyczny ze wzorem opisującym SNR_{max} - maksymalną wartość SNR dla sygnału gaussowskiego wynikającą z teorii Shannona o kodowaniu źródła. Shannon udowodnił [182, 183], że powyższy wniosek jest słuszny dla wszystkich sygnałów, nie tylko dla sygnału gaussowskiego. Oznacza to, że kwantyzator wektorowy jest asymptotycznie optymalnym koderem: dla danej przepływności binarnej, gdy wymiar wektora N (i oczywiście liczba wektorów w słowniku L) dąży do nieskończoności, kwantyzator wektorowy zapewnia najmniejsze zniekształcenia kodowania.

Ciekawe byłoby porównanie kwantyzatora wektorowego z koderem ADPCM. Dla nieskorelowanego sygnału gaussowskiego przewaga kwantyzatora wektorowego $\frac{SNR(b, N)}{SNR_{ADPCM}} = \frac{c(1)}{c(N)}$

¹Dla sygnału o zerowej wartości średniej macierz kowariancji jest równa macierzy korelacji, z tego względu użyto oznaczenia właściwego dla macierzy korelacji.

jest tym większa, im większy jest wymiar wektorów przetwarzanych w kwantyzatorze wektorowym. Dla skorelowanego sygnału gaussowskiego słuszny jest wzór:

$$\frac{SNR(b, N)}{SNR_{ADPCM}} = \frac{c(1)}{c(N)} \frac{\det \rho^{p+1}(\mathbf{x})}{\det \rho^p(\mathbf{x})} [\det \rho^N(\mathbf{x})]^{1/N} \quad (3.10)$$

Wynik porównania obu koderów zależy od sygnału, wymiaru przetwarzanych wektorów N i rzędu predyktora p . Moreau [139] wykonał serię obliczeń, porównując kwantyzator wektorowy z koderem ADPCM (dla $N = p$). Dla silnie skorelowanych sygnałów koder ADPCM przewyższał niekiedy kwantyzator wektorowy. Dzieje się tak dlatego, że predyktor wykorzystuje korelacje między próbkami należącymi do każdego wybranego ciągu, natomiast kwantyzator wektorowy nie wykorzystuje korelacji między próbkami należącymi do różnych wektorów. Jednak dla odpowiednio wysokiego N kwantyzator wektorowy zawsze przewyższa koder ADPCM.

Dla sygnałów niegaussowskich przewaga kwantyzatora wektorowego nad koderem ADPCM może być znacznie większa. Na podstawie wzoru (3.6) dla sygnału nieskorelowanego przewaga ta wynosi:

$$\frac{SNR(b, N)}{SNR_{ADPCM}} = \frac{\sigma_e^2(b, 1)}{\sigma_e^2(b, N)} = \frac{\bar{c}(1)}{\bar{c}(N)} \frac{\|p(x)\|_{\frac{1}{3}}}{\|p(\mathbf{x})\|_{\frac{N}{N+2}}} \quad (3.11)$$

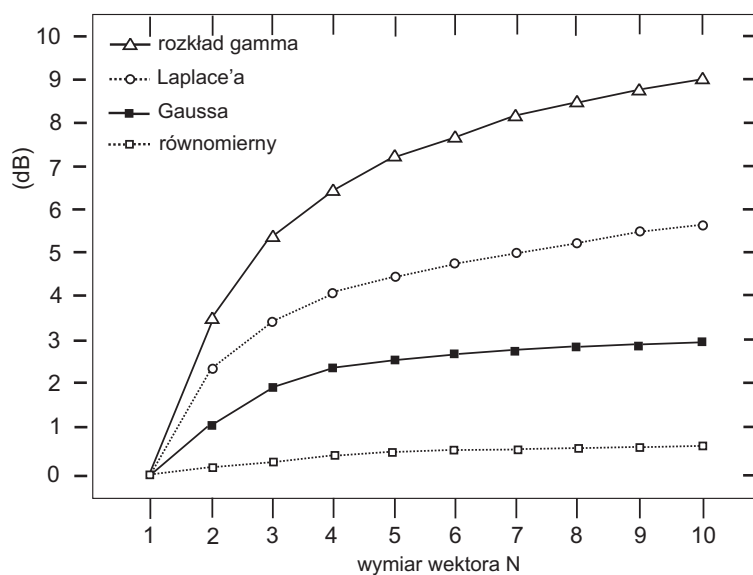
Uwzględniając brak korelacji między składowymi wektora \mathbf{x} , można napisać $p(\mathbf{x}) = \prod_{i=1}^N p(x_i)$, z czego wynika następująca równość:

$$\begin{aligned} \|p(\mathbf{x})\|_{\frac{N}{N+2}} &= \left[\int_{R^N} \prod_{i=1}^N (p(x_i))^{\frac{N}{N+2}} dx_1 \dots dx_N \right]^{\frac{N+2}{N}} \\ &= \prod_{i=1}^N \left[\int_{R^1} (p(x_i))^{\frac{N}{N+2}} dx_i \right]^{\frac{N+2}{N}} \\ &= \left[\int_{R^1} (p(x))^{\frac{N}{N+2}} dx \right]^{N+2} \end{aligned} \quad (3.12)$$

Po podstawieniu do równania (3.11) otrzymuje się:

$$\frac{SNR(b, N)}{SNR_{ADPCM}} = \frac{\bar{c}(1)}{\bar{c}(N)} \frac{\left[\int_{R^1} (p(x))^{\frac{1}{3}} dx \right]^3}{\left[\int_{R^1} (p(x))^{\frac{N}{N+2}} dx \right]^{N+2}} \quad (3.13)$$

Wyniki (opracowane na podstawie [124]) podano na rys.3.3. Im bardziej $p(x)$ odbiega od rozkładu równomiernego, tym większy jest zysk kwantyzatora wektorowego nad koderem ADPCM.



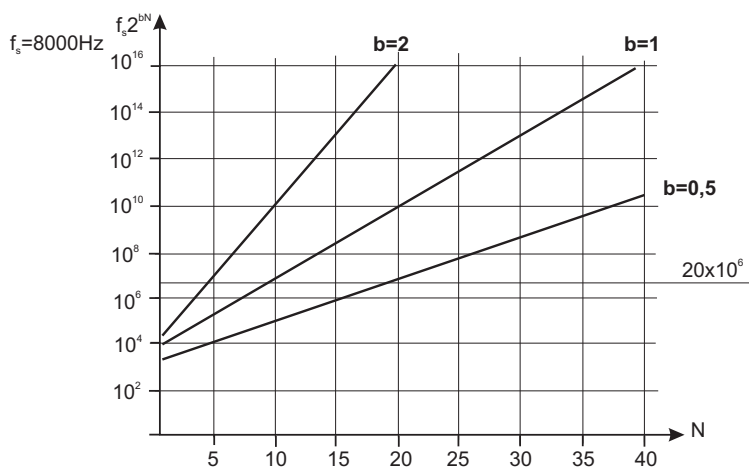
Rysunek 3.3: Przewaga kwantyzatora wektorowego nad koderem ADPCM ($SNR(b, N)/SNR_{ADPCM}$) dla sygnałów nieskorelowanych

3.1.3 Problem złożoności obliczeniowej

W punkcie 3.1.2 zwrócono uwagę, że kwantyzator wektorowy jest asymptotycznie optymalnym koderem źródła sygnału, tzn. w miarę wzrostu wymiaru przetwarzanych wektorów N błąd kwantowania zbliża się do wartości minimalnej określonej przez Shannona [182, 183]. Minimalna moc zniekształceń zależy od właściwości sygnału. W pracy [8] podano algorytmy obliczania minimalnej mocy zniekształceń i przedstawiono wyniki obliczeń dla konkretnych sygnałów.

Nadmierne zwiększanie wymiaru wektorów N nie jest jednak możliwe z uwagi na złożoność obliczeniową algorytmu przeszukiwania słownika i z uwagi na pojemność pamięci przeznaczoną na jego przechowywanie. Przy rozdzielczości b bitów na próbkę słownik kwantyzatora wektorowego liczy $L = 2^{bN}$ wektorów i jego przeszukiwanie (obliczenie L odległości euklidesowych N -wymiarowych wektorów) wymaga wykonania NL mnożeń. Na jedną próbkę przypada więc L mnożeń, a na sekundę $f_s L = f_s 2^{bN}$ mnożeń, gdzie f_s jest częstotliwością próbkowania.

Na wykresie 3.4 pokazano wykładniczą zależność liczby mnożeń na sekundę od wymiaru wektora N przy danej rozdzielczości b . Moc obliczeniowa typowych procesorów sygnałowych, rzędu 20 milionów operacji długich na sekundę, poważnie ogranicza wymiar wektora i nie umożliwia zmniejszenia poziomu szumu kwantyzacji.



Rysunek 3.4: Złożoność obliczeniowa algorytmu przeszukiwania słownika (tzn. liczba operacji długich na sekundę) w funkcji wymiaru wektora (b jest rozdzielczością w bitach na próbkę)

3.2 Dekompozycja słownika kwantyzatora wektorowego

3.2.1 Słownik w postaci iloczynu kartezjańskiego

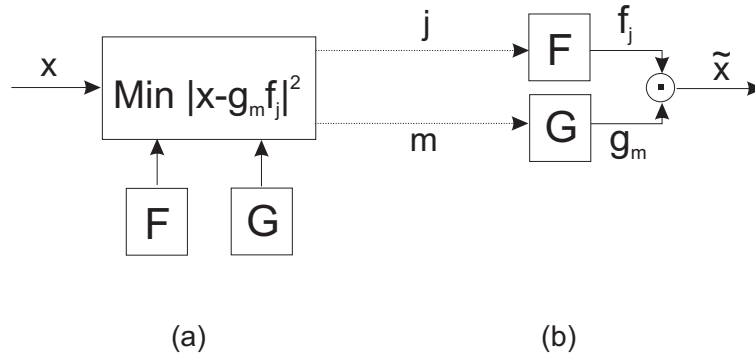
Jak pokazano w poprzednim punkcie, wymiar przetwarzanych wektorów jest ograniczony, ze względu na wzrost (wg zależności wykładniczej) liczby wektorów w słowniku ($L = 2^{bN}$). Przeszukiwanie wielkich słowników w czasie rzeczywistym nie jest możliwe, można jednak nadać słownikowi pewną strukturę, umożliwiającą przeszukiwanie tylko jego części. Przykładem może być słownik o strukturze drzewa binarnego [75], który wymaga jedynie $\log_2 L = bN$ porównań kwantowanego wektora z wektorami słownika. Oczywiście nie ma pewności, że działając w ten sposób odnajdzie się ten z L wektorów, który jest najbliższy wektorowi kwantowanemu. Innym rozwiązaniem jest przedstawienie słownika \mathbf{F} w postaci iloczynu kartezjańskiego K mniejszych słowników:

$$\mathbf{F} = \mathbf{F}^1 \times \mathbf{F}^2 \times \dots \times \mathbf{F}^K \quad (3.14)$$

i przeszukiwanie tych słowników po kolei. Każdy słownik elementarny \mathbf{F}^k zawiera L_k wektorów (niekoniecznie o wymiarze N - niektóre słowniki mogą składać się nawet ze skalarów):

$$\mathbf{F}^k = \{\mathbf{f}_1^k, \dots, \mathbf{f}_{L_k}^k\} \quad (3.15)$$

Z wektorów należących do słowników elementarnych można zestawić $L = \prod_{k=1}^K L_k$ wektorów słownika \mathbf{F} . Dzięki szeregowemu przeszukiwaniu słowników elementarnych liczba porównań wynosi $\sum_{k=1}^K L_k$. Tak więc dzięki dekompozycji słownika \mathbf{F} na K słowników elementarnych możliwe jest radykalne zmniejszenie złożoności obliczeniowej algorytmu kwantowania wektorowego (oczywiście kosztem zwiększenia błędu kwantowania). Takie podejście znane jest w literaturze pod nazwą PCVQ - *Product Code Vector Quantization* [75].



Rysunek 3.5: Kwantyzator wektorowy typu kształt wzmocnienie: (a) - koder, (b) - dekoder

Kwantyzator wektorowy z dekompozycją słownika (*Product Code Vector Quantizer*) jest określony, gdy znane są słowniki elementarne $\mathbf{F}^1, \mathbf{F}^2, \dots, \mathbf{F}^K$ oraz algorytm modelowania wektora \mathbf{x} , tj. wyznaczania jego skwantowanej kopii $\tilde{\mathbf{x}}$:

$$\tilde{\mathbf{x}} = \tilde{\mathbf{x}}(\mathbf{f}_{j_1}^1, \mathbf{f}_{j_2}^2, \dots, \mathbf{f}_{j_K}^K) \quad (3.16)$$

Należy zwrócić uwagę na fakt, że algorytm projektowania słowników PCVQ zależy od wybranego algorytmu modelowania.

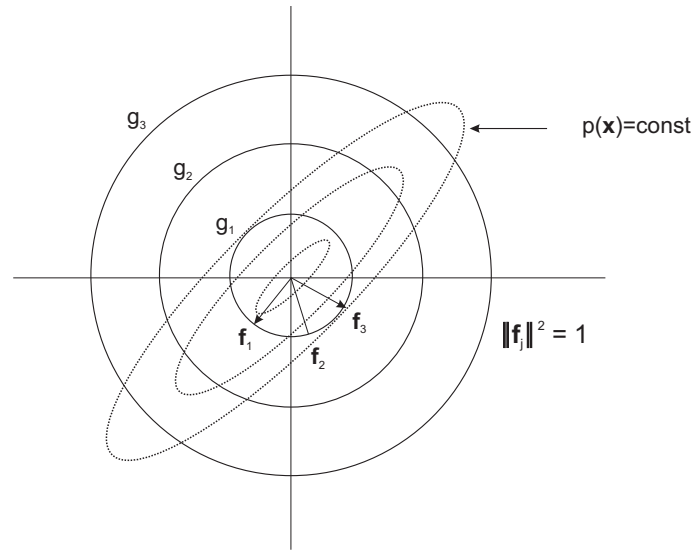
3.2.2 Kwantyzatory wektorowe typu kształt - wzmocnienie

Najczęściej wykorzystywaną metodą dekompozycji słownika kwantyzatora wektorowego jest iloczyn kartezyjański dwóch słowników: słownika kształtów (\mathbf{F}) i słownika wzmocnień (\mathbf{G}). Słownik kształtów zawiera L_s wektorów N -wymiarowych, natomiast słownik wzmocnień - L_g skalarów, które są poziomami kwantyzacji wzmocnienia. Wektor sygnału skwantowanego jest iloczynem kształtu i wzmocnienia:

$$\tilde{\mathbf{x}} = g_m \mathbf{f}_j \quad (3.17)$$

Działanie takiego kwantyzatora wektorowego, zwanego kwantyzatorem wektorowym typu kształt-wzmocnienie (SGVQ - *Shape Gain Vector Quantizer* [174]) pokazano na rys.3.5. Tego typu dekompozycja wynika z właściwości przetwarzanego sygnału (najczęściej telefonicznego): poziom sygnału może się zmieniać w szerokim zakresie, natomiast jego kształt w niewielkim stopniu zależy od poziomu (podobne kształty opisują wektory pobrane z głośno i cicho wypowiedzanej frazy).

Należy odróżniać SGVQ od kwantyzatora wektorowego z adaptacyjnym wzmocnieniem [75, 52, 167], w którym wzmocnienie dobiera się raz na kilkanaście - kilkadziesiąt wektorów, proporcjonalnie do wyestymowanej wartości skutecznej sygnału ($\hat{\sigma}_x$). Jest to w istocie klasyczny VQ, przetwarzający znormalizowany (z wykorzystaniem $\hat{\sigma}_x$) sygnał wejściowy.



Rysunek 3.6: Słownik kształt - wzmocnienie i gęstość prawdopodobieństwa skorelowanego sygnału gaussowskiego

Kwantyzator wektorowy typu kształt wzmocnienie występuje w kilku odmianach: ze znormalizowanym słownikiem kształtów ($\|\mathbf{f}_j\| = 1$) i bez tej normalizacji, a także jako tzw. kwantyzator wektorowy typu kontur - wzmocnienie (CGVQ - *Contour Gain Vector Quantizer* [128]). Znormalizowany słownik kształtów nadaje się do kwantyzacji sygnałów nieskorelowanych. Dostępne modele wektorów sygnału wejściowego, otrzymywane przez łączenie kształtów i wzmocnień, leżą na powierzchni L_g sfer o promieniach g_1, \dots, g_{L_g} - rys.3.6. Z kolei N - wymiarowa gęstość prawdopodobieństwa skorelowanego sygnału gaussowskiego jest stała na powierzchni elipsoid. Wynika stąd, że niektóre zestawienia $g_m \mathbf{f}_j$ będą wykorzystywane bardzo rzadko. Celowe jest więc zróżnicowanie norm wektorów kształtów (ich końce powinny leżeć na powierzchni jednej ze wspomnianych elipsoid). Inną możliwość przestudiowano w pracy Dymarskiego, Chmielewskiego i Ratre [52]: dla każdego kształtu przewidziano inny kwantyzator wzmocnień.

3.2.3 Modelowanie wielostopniowe i predykcja długookresowa

W kwantyzatorze wektorowym z modelowaniem wielostopniowym (ang. *multistage VQ* [108]) wektor sygnału wejściowego \mathbf{x} jest przybliżany sumą K wektorów:

$$\tilde{\mathbf{x}} = \mathbf{f}_{j_1}^1 + \mathbf{f}_{j_2}^2 + \dots + \mathbf{f}_{j_K}^K \quad (3.18)$$

W ogólnym przypadku każdy z tych wektorów może być pobrany z osobnego słownika: $\mathbf{f}_{j_k}^k \in \mathbf{F}^k$, $k = 1, \dots, K$.

Często, zwłaszcza w koderach CELP, o których będzie mowa w następnym punkcie, na każdym etapie jest wykorzystywany kwantyzator wektorowy typu kształt- wzmocnienie:

$$\tilde{\mathbf{x}} = g_{m_1}^1 \mathbf{f}_{j_1}^1 + g_{m_2}^2 \mathbf{f}_{j_2}^2 + \dots + g_{m_K}^K \mathbf{f}_{j_K}^K \quad (3.19)$$

Równaniem (3.19) można opisać wiele znanych kwantyzatorów wektorowych i koderów CELP (tych ostatnich po dołączeniu filtru predykcyjnego - patrz następny punkt). Zależy to od zawartości słowników, ich liczby, oraz ograniczeń dotyczących kolejności i sposobu ich przeszukiwania. Np. stosując na każdym z K etapów taki sam słownik opisany macierzą jednostkową \mathbf{I} stopnia $N \times N$ otrzymuje się pobudzenie impulsowe dla kodera CELP [7]. Indeksy j_k i wzmocnienia g_k oznaczają odpowiednio położenia i amplitudy K wybranych impulsów. Po uwzględnieniu dodatkowych ograniczeń dotyczących położenia impulsów, otrzymamy koder predykcyjny (CELP) o pobudzeniu regularnym [44]. Położenia K impulsów są wówczas określone wzorem $j_{k+1} = j_k + N/K$, co umożliwi zakodowanie położenia tylko pierwszego impulsu.

W niektórych kwantyzatorach wektorowych i w większości koderów CELP stosowana jest predykcja długookresowa (LTP - ang. *long-term prediction*) [169, 170]. Odwołuje się ona do sygnału wyjściowego dekodera $\tilde{\mathbf{x}}$ wygenerowanego w bezpośredniej przeszłości. Umożliwia ona efektywne zamodelowanie mowy dźwięcznej, jeżeli wykorzystany zostanie sygnał opóźniony o okres tonu krtaniowego lub jego wielokrotność. Opóźniony sygnał można umieścić w tzw. słowniku adaptacyjnym [111, 112], który dla kwantyzatora wektorowego, przetwarzającego wektor sygnału wejściowego $\mathbf{x} = [x(0), x(1), \dots, x(N-1)]^t$, przedstawia się następująco:

$$\mathbf{F}^1 = \begin{bmatrix} \tilde{x}(-N) & \tilde{x}(-N-1) & \dots & \tilde{x}(-N-L_1+1) \\ \tilde{x}(-N+1) & \tilde{x}(-N) & \dots & \tilde{x}(-N-L_1+2) \\ \vdots & \vdots & & \vdots \\ \tilde{x}(-1) & \tilde{x}(-2) & \dots & \tilde{x}(-L_1) \end{bmatrix} \quad (3.20)$$

Po zamodelowaniu bieżącego wektora \mathbf{x} , do słownika wpisuje się N nowych wektorów, usuwając N wektorów zawierających sygnał najbardziej opóźniony. W większości przypadków ze słownika adaptacyjnego pobiera się tylko jeden wektor $\mathbf{f}_{j_1}^1$. Jego indeks j_1 ma związek z opóźnieniem wektora sygnału w predykcji długookresowej (opóźnienie jest równe $j_1 + N$). Odpowiednie wzmocnienie $g_{m_1}^1$ jest współczynnikiem predykcji długookresowej. Niekiedy ze słownika adaptacyjnego pobiera się kilka kolejnych wektorów, tworząc predyktor długookresowy o wielu współczynnikach [166]. W słowniku można również umieszczać dodatkowe wektory otrzymane metodą interpolacji kolejnych wektorów zapisanych w kolumnach macierzy (3.20). Otrzymuje się w ten sposób tzw. predyktor o opóźnieniach ułamkowych [133, 114].

W kwantyzatorach wektorowych i koderach CELP liczba etapów modelowania K jest najczęściej ograniczona do dwóch: w pierwszym etapie wykorzystuje się słownik adaptacyjny, a w drugim - stały. W kwantyzatorze wektorowym konieczne jest obliczanie słownika stałego z uwzględnieniem właściwości przetwarzanego sygnału. W koderze CELP nie jest to regułą - niekiedy słownik się oblicza [38, 37], niekiedy wypełnia się go sekwencją o rozkładzie gaussowskim, lub specjalnymi sygnałami łatwymi w obróbce. Należy tu wymienić sekwencje "rzadkie" [36] (tzn. zawierające dużo próbek zerowych), binarne [87] (próbki o wartościach

± 1), ternarne [93] (próbki o wartościach $0, \pm 1$). Sekwencje binarne i ternarne zwane są algebraicznymi [3]. Sekwencje ternarne są z natury swej "rzadkie", niekiedy również regularne [180].

Wprowadzając odpowiednie ograniczenia, można do opisu koderów ze słownikami algebraicznymi wykorzystać wzór (3.19). Np. sekwencja binarna może być otrzymana drogą kolejnego wybierania $K = N$ wektorów ze słownika opisanego macierzą jednostkową \mathbf{I} stopnia $N \times N$ z następującym ograniczeniem co do współczynników wzmocnienia:

$$g^k = g s^k \quad k = 1 \dots K \quad (3.21)$$

gdzie g oznacza amplitudę a s^k znak k -tej próbki sygnału. W tym przypadku współczynniki wzmocnienia różnią się jedynie znakiem. W ten sam sposób można opisać sygnał ternarny, jednak w tym wypadku wybiera się $K < N$ wektorów. Dla ustalonego K można otrzymać $N!/(K!(N-K)!) 2^K$ różnych sygnałów. Nie określając liczby składników K , można otrzymać 3^N sygnałów. Innym przykładem jest składanie N -wymiarowego wektora poprzez konkatencję ("sklejanie") K krótkich wektorów wybranych z tego samego słownika \mathbf{F}' z tym samym współczynnikiem wzmocnienia g (lub z ograniczeniem opisanym wzorem (3.21)). W tym ostatnim przypadku można przepisać równanie (3.19) w następującej postaci:

$$\tilde{\mathbf{x}} = g \sum_{k=1}^K s^k \mathbf{f}_{j_k}^k \quad (3.22)$$

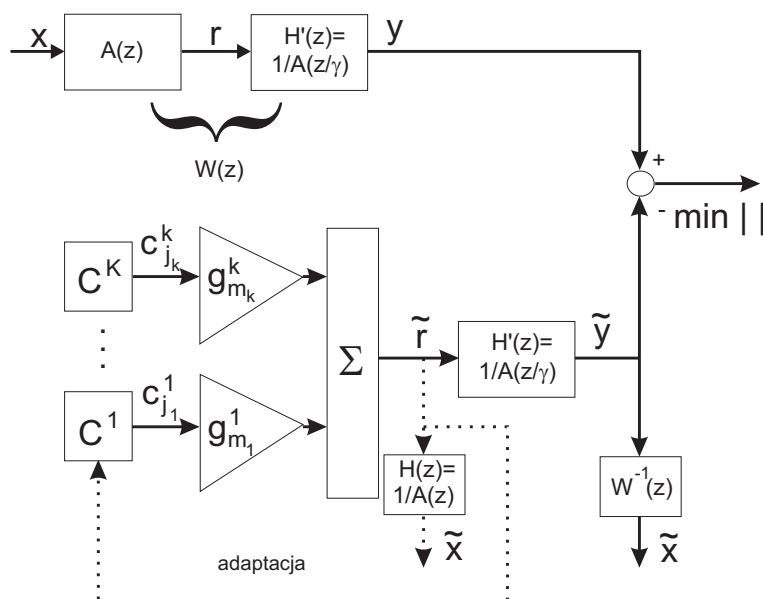
Słowniki $\mathbf{F}^1, \mathbf{F}^2, \dots, \mathbf{F}^K$, z których wybiera się kolejne wektory występujące we wzorze (3.22), opisane są wówczas wzorami:

$$\mathbf{F}^1 = \begin{bmatrix} \mathbf{F}' \\ \mathbf{0} \\ \vdots \\ \mathbf{0} \end{bmatrix}, \quad \mathbf{F}^2 = \begin{bmatrix} \mathbf{0} \\ \mathbf{F}' \\ \vdots \\ \mathbf{0} \end{bmatrix}, \quad \dots \quad \mathbf{F}^K = \begin{bmatrix} \mathbf{0} \\ \mathbf{0} \\ \vdots \\ \mathbf{F}' \end{bmatrix} \quad (3.23)$$

Słowniki \mathbf{F}^k , $k = 1, \dots, K$ można połączyć w jeden słownik "mieszany"

$$\mathbf{F}^1 \quad \mathbf{F}^2 \quad \dots \quad \mathbf{F}^K \quad (3.24)$$

Np. w kwantyzatorze wektorowym z predykcją długookresową słownik "mieszany" zawiera część adaptacyjną i stałą. Pierwszy wybrany wektor powinien pochodzić z części adaptacyjnej, a drugi - ze stałej. Można jednak znieść to ograniczenie i przeszukiwać dwukrotnie cały słownik "mieszany" - umożliwi to np. wyeliminowanie w naturalny sposób predykcji długookresowej dla mowy bezdźwięcznej, gdy jest ona mało skuteczna. Pomysł utworzenia słownika "mieszanego" i usunięcia ograniczeń w trakcie jego przeszukiwania został opublikowany przez N.Moreau i autora niniejszej rozprawy [140, 141].



Rysunek 3.7: Koder/dekoder CELP

3.2.4 Koder CELP jako PCVQ

Podstawowa różnica między koderem CELP (*Code Excited Linear Prediction*) a kwantyzatorem wektorowym polega na tym, że w dekodery CELP (podobnie jak w dekodery ADPCM) sygnał jest przepuszczany przez predykcyjny filtr syntezy o transmitancji $H(z) = 1/A(z)$. Operację tę można potraktować jako dodatkowy etap dekompozycji słownika kwantyzatora wektorowego: jest on iloczynem kartezjańskim zbioru wektorów nieodfiltrowanego słownika i zbioru dopuszczalnych transmitancji filtru predykcyjnego. Zbiór transmitancji zawiera skończoną liczbę elementów, gdyż współczynniki predykcji określające każdą transmitancję są skwantowane (skalarnie lub wektorowo).

Typowy koder CELP zawiera adaptacyjne filtry preemfazy i deemfazy (tzw. filtry percepcyjne $W(z)$ i $W^{-1}(z)$) kształtujące widmo szumu kwantyzacji. Modelowanie sygnału jest wielostopniowe - najczęściej dwustopniowe, z wykorzystaniem w pierwszym etapie predyktora długookresowego w układzie ze słownikiem adaptacyjnym tak, jak przedstawiono to w punkcie 3.2.3. W każdym etapie modelowania sygnału wykorzystuje się kwantyzator wektorowy typu kształt- wzmocnienie (SGVQ), patrz p.3.2.2.

Ogólny schemat kodera i dekodera CELP pokazano na rys.3.7. Minimalizacja odległości (euklidesowej) sygnału zrekonstruowanego w odbiorniku i sygnału oryginalnego jest przeprowadzona na poziomie sygnałów "percepcyjnych" \tilde{y} i y . Na tym poziomie dąży się do uzyskania możliwie "płaskiej" gęstości mocy szumu kwantyzacji. Umożliwia to uzyskanie pożądanej gęstości mocy szumu kwantyzacji na poziomie sygnału wyjściowego odbiornika \tilde{x} . Odpowiednie filtry "percepcyjne" mają transmitancje określone przez współczynniki predykcji: transmitancja filtru nadawczego wynosi najczęściej $W(z) = A(z)/A(z/\gamma)$, a filtru odbiorczego $W^{-1}(z) = A(z/\gamma)/A(z)$. Współczynnik tłumienia $\gamma < 1$ określa głębokość

preemfazy/deemfazy: dla wartości zbliżonych do jedności preemfaza zanika. $A(z)$ jest transmitancją filtru inwersyjnego: podając na jego wejście sygnał akustyczny \mathbf{x} na jego wyjściu otrzymuje się sygnał resztkowy (ang. *residual*) \mathbf{r} . Sygnał resztkowy jest w swojej istocie błędem predykcji sygnału \mathbf{x} - w teorii koderów ADPCM zwany jest również sygnałem różnicowym. Filtry "percepcyjne" adaptują się do sygnału, gdyż co 10-30 ms oblicza się i przesyła do odbiornika nowe współczynniki predykcji. Kodery CELP, w których współczynniki predykcji są obliczane z sygnału zrekonstruowanego (w związku z tym nie są transmitowane, podobnie jak w koderach ADPCM), należą do rzadkości (patrz np. [28]). W większości przypadków do wyznaczania współczynników predykcji stosuje się klasyczne metody blokowe predykcji liniowej (metoda autokorelacyjna, kowariancyjna). Są one opisane w licznych monografiach, np. [132], [189], [102], w związku z tym szczegółowy ich opis zostanie tu pominięty. Współczynniki predykcji są najczęściej przesyłane jako tzw. współczynniki widma prążkowego (LSF - *line spectrum frequencies* lub LSP - *line spectrum pairs*) [94, 190]. Współczynniki te kwantowano początkowo skalarnie, obecnie coraz częściej wektorowo [109, 187, 161, 203].

Wielostopniowy kwantyzator wektorowy typu kształt- wzmocnienie operuje bezpośrednio na poziomie sygnału resztkowego \mathbf{r} , chociaż błąd kwantyzacji mierzony jest na poziomie sygnału percepcyjnego \mathbf{y} . Zestawiany jest pewien model sygnału resztkowego $\tilde{\mathbf{r}}$ z wykorzystaniem słowników sygnałów resztkowych $\mathbf{C}^1, \mathbf{C}^2, \dots, \mathbf{C}^K$. Sygnał percepcyjny jest otrzymywany przez filtrację sygnału resztkowego za pomocą filtru o transmitancji $H'(z) = 1/A(z/\gamma)$. Sygnał wyjściowy odbiornika $\tilde{\mathbf{x}}$ może być otrzymany przez filtrację sygnału "percepcyjnego" $\tilde{\mathbf{y}}$ z wykorzystaniem filtru $W^{-1}(z)$ - częściej stosowaną metodą jest bezpośrednia filtracja sygnału resztkowego $\tilde{\mathbf{r}}$ z wykorzystaniem filtru predykcyjnego $H(z)$, o transmitancji opisanej następującym wzorem:

$$H(z) = H'(z) W^{-1}(z) = \frac{1}{A(z/\gamma)} \frac{A(z/\gamma)}{A(z)} = \frac{1}{A(z)} \quad (3.25)$$

Sygnał $\tilde{\mathbf{x}}$ jest wytwarzany jedynie w odbiorniku i tylko tam jest wykorzystywany filtr $H(z)$. W nadajniku wystarcza odtworzenie sygnału percepcyjnego $\tilde{\mathbf{y}}$.

Wektor sygnału percepcyjnego $\tilde{\mathbf{y}}$ jest otrzymywany metodą filtracji odpowiedniego wektora sygnału resztkowego $\tilde{\mathbf{r}}$, co można opisać wzorem:

$$\tilde{\mathbf{y}} = \mathbf{H}' \tilde{\mathbf{r}} + \tilde{\mathbf{y}}_0, \quad \mathbf{H}' = \begin{bmatrix} h'_0 & 0 & \cdots & 0 \\ h'_1 & h'_0 & \cdots & 0 \\ h'_2 & h'_1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ h'_{N-1} & h'_{N-2} & \cdots & h'_0 \end{bmatrix} \quad (3.26)$$

gdzie $\tilde{\mathbf{y}}_0$ jest wektorem próbek odpowiedzi filtru $H'(z)$ na zerowe pobudzenie (ang. *ringing*)², natomiast macierz \mathbf{H}' tego filtru, zawierająca próbki jego odpowiedzi impulsowej, reprezentuje filtrację z zerowymi warunkami początkowymi.

²W dalszej części tekstu wektor $\tilde{\mathbf{y}}_0$ będzie nazywany w skrócie "ringing" $\tilde{\mathbf{y}}_0$.

W koderach CELP z modelowaniem wielostopniowym, do modelowania sygnału resztkowego jest wykorzystywany schemat kształt - wzmocnienie:

$$\tilde{\mathbf{r}} = \sum_{k=1}^K g_{m_k}^k \mathbf{c}_{j_k}^k \quad (3.27)$$

gdzie $\mathbf{c}_{j_k}^k \in \mathbf{C}^k$.

Po podstawieniu do wzoru (3.26) otrzymuje się:

$$\tilde{\mathbf{y}} = \sum_{k=1}^K g_{m_k}^k \mathbf{f}_{j_k}^k + \tilde{\mathbf{y}}_0 \quad (3.28)$$

gdzie $\mathbf{f}_{j_k}^k = \mathbf{H}' \mathbf{c}_{j_k}^k$ jest odfiltrowanym wektorem $\mathbf{c}_{j_k}^k$. Można powiedzieć, że w wyniku filtracji (z zerowymi warunkami początkowymi) słownika sygnałów resztkowych \mathbf{C}^k otrzymuje się słownik sygnałów percepcyjnych \mathbf{F}^k :

$$\mathbf{F}^k = \mathbf{H}' \mathbf{C}^k \quad (3.29)$$

Słowniki \mathbf{C}^k , $k = 1, \dots, K$ można połączyć w jeden słownik "mieszany" (podobnie jak to opisano we wzorze (3.24)), można wówczas mówić o nieodfiltrowanym słowniku "mieszanym" \mathbf{C} i o odfiltrowanym słowniku "mieszanym" $\mathbf{F} = \mathbf{H}' \mathbf{C}$.

Wprowadzając odpowiednie ograniczenia, o których była mowa w poprzednim punkcie, można przy użyciu wzoru (3.28) opisać sekwencje binarne, ternarne itp. W szczególności można wymagać, aby współczynniki wzmocnienia różniły się jedynie znakiem (wzór (3.21)) - wówczas wzór (3.28) można przepisać w postaci:

$$\tilde{\mathbf{y}} = g \sum_{k=1}^K s^k \mathbf{f}_{j_k}^k + \tilde{\mathbf{y}}_0 \quad (3.30)$$

Nietrudno zauważyć, że wzór (3.28) przechodzi we wzór (3.19) (podobnie wzór (3.30) przechodzi we wzór (3.22)), gdy słownik nie podlega filtracji (wówczas $\mathbf{H}' = \mathbf{I}$, $\tilde{\mathbf{y}}_0 = \mathbf{0}$). Oznacza to, że kwantyzator wektorowy (z adaptacyjną preemfazą i deemfazą) jest szczególnym przypadkiem koder CELP. Preemfazę i deemfazę można też usunąć, podstawiając $W(z) = 1$ lub $\gamma = 1$.

Takie ujednoczone podejście do analizy kwantyzatorów wektorowych i koderów CELP umożliwi uzyskanie wspólnych, dla obu klas koderów, algorytmów przeszukiwania i konstruowania słowników.

3.3 Projektowanie koderów typu kształt - wzmocnienie

3.3.1 Przeszukiwanie słowników i tworzenie modelu sygnału mowy

Kwantyzator wektorowy typu kształt - wzmocnienie (SGVQ) został przedstawiony w punkcie 3.2.2. Obecnie omówione będą algorytmy kwantowania, tzn. tworzenia wektora sygnału zrekonstruowanego $\tilde{\mathbf{x}}$, będącego modelem (aproksymacją) i -go wektora sygnału wejściowego $\mathbf{x} = \mathbf{x}(i)$ (dla uproszczenia notacji indeks "i" będzie pomijany, o ile nie będzie to prowadziło do niejednoznaczności). Jeśli zastosowano preemfazę i deemfazę kształtującą widmo szumu kwantyzacji, to modelowanie odbywa się na poziomie tzw. sygnału percepcyjnego \mathbf{y} . Podejście takie jest regułą w koderach typu CELP. Zadanie można sformułować w następujący sposób:

Mając dany słownik kształtów $\{\mathbf{f}_j\}, j = 1, \dots, L_s$, słownik wzmocnień $\{g_m\}, m = 1, \dots, L_g$, oraz N -wymiarowy wektor próbek sygnału percepcyjnego \mathbf{y} poszukuje się jego modelu $\tilde{\mathbf{y}}$ według wzoru

$$\tilde{\mathbf{y}} = g_m \mathbf{f}_j \quad (3.31)$$

dążąc do jak najmniejszego kwadratu odległości euklidesowej

$$D(i) = \|\mathbf{y} - \tilde{\mathbf{y}}\|^2 = \|\mathbf{y} - g_m \mathbf{f}_j\|^2 \quad (3.32)$$

Powyższe zadanie może odnosić się do kodera CELP, z tym zastrzeżeniem, że model sygnału percepcyjnego powstaje tu drogą filtracji odtworzonego sygnału resztkowego (według wzoru (3.26): $\tilde{\mathbf{y}} = \mathbf{H}' \tilde{\mathbf{r}} + \tilde{\mathbf{y}}_0$). W najprostszym przypadku model sygnału resztkowego $\tilde{\mathbf{r}}$ powstaje w kwantyzatorze wektorowym typu kształt - wzmocnienie: $\tilde{\mathbf{r}} = g_m \mathbf{c}_j$, gdzie $\{\mathbf{c}_j\}, j = 1, \dots, L_s$, jest słownikiem kształtów sygnału resztkowego. Typowy CELP składa się z kilku, najczęściej dwóch etapów modelowania opartych na SGVQ - będzie o tym mowa w dalszej części pracy. W pierwszym etapie otrzymuje się następujący model sygnału percepcyjnego:

$$\tilde{\mathbf{y}} = g_m \mathbf{H}' \mathbf{c}_j + \tilde{\mathbf{y}}_0 = g_m \mathbf{f}_j + \tilde{\mathbf{y}}_0 \quad (3.33)$$

gdzie $\mathbf{f}_j = \mathbf{H}' \mathbf{c}_j$ jest wektorem ze słownika "odfiltrowanego" (wzór (3.29)). Wzór (3.32) określający energię błędu kwantyzacji, można dla kodera CELP przepisać w następującej postaci:

$$D(i) = \|\mathbf{y} - \tilde{\mathbf{y}}_0 - g_m \mathbf{H}' \mathbf{c}_j\|^2 = \|\mathbf{y}_0 - g_m \mathbf{f}_j\|^2 \quad (3.34)$$

gdzie $\mathbf{y}_0 = \mathbf{y} - \tilde{\mathbf{y}}_0$ jest sygnałem docelowym (ang. *target signal*) w pierwszym etapie modelowania. Sygnał ten otrzymuje się, odejmując od sygnału percepcyjnego odpowiedź filtru $H'(z)$ na zerowe pobudzenie (ang. *ringing*). Tak więc zadania kwantyzacji w SGVQ (opisane wzorami (3.31) i (3.32)) i w CELP (opisane wzorami (3.33) i (3.34)) są w istocie swojej identyczne (w CELP należy jedynie posłużyć się odfiltrowanym słownikiem i zmodyfikowanym sygnałem percepcyjnym).

Jak już wspomniano, testowanie wszystkich możliwych kombinacji \mathbf{f}_j, g_m nie jest możliwe ze względów praktycznych - najczęściej słowniki przeszukuje się w kolejności $\{f_j\}, \{g_m\}$. Za każdym razem chodzi o minimalizację kwadratu odległości (dla SGVQ należy podstawić wektor docelowy $\mathbf{y}_0 = \mathbf{y}$):

$$\begin{aligned} D(i) &= \|\mathbf{y}_0 - g_m \mathbf{f}_j\|^2 \\ &= \mathbf{y}_0^t \mathbf{y}_0 - 2g_m \mathbf{y}_0^t \mathbf{f}_j + g_m^2 \mathbf{f}_j^t \mathbf{f}_j \\ &= \mathbf{y}_0^t \mathbf{y}_0 - \frac{[\mathbf{y}_0^t \mathbf{f}_j]^2}{\mathbf{f}_j^t \mathbf{f}_j} + \mathbf{f}_j^t \mathbf{f}_j [g_m - g]^2 \end{aligned} \quad (3.35)$$

gdzie $g = \frac{\mathbf{y}_0^t \mathbf{f}_j}{\mathbf{f}_j^t \mathbf{f}_j}$

Algorytm przeszukiwania słowników wynika ze wzoru (3.35). Mając dany kształt \mathbf{f}_j można bez trudu znaleźć odpowiednie wzmocnienie g_m - ze słownika wzmocnień wybiera się wartość najbliższą g . Wartość g może być uważana za optymalne (nieskwantowane) wzmocnienie. Nietrudno zresztą zauważyć że $g\mathbf{f}_j$ jest rzutem ortogonalnym \mathbf{y}_0 na \mathbf{f}_j .

W pierwszej kolejności wybierany jest odpowiedni kształt \mathbf{f}_j . Jeśli słownik kształtów jest znormalizowany, czyli

$$\mathbf{f}_j^t \mathbf{f}_j = 1, \quad j = 1, \dots, L_s \quad (3.36)$$

i jeśli słownik wzmocnień zawiera jedynie wartości dodatnie, czyli

$$g_m > 0, \quad m = 1, \dots, L_g \quad (3.37)$$

to wówczas, na podstawie wzoru (3.35), najlepszy wektor kształtu maksymalizuje iloczyn skalarny $\mathbf{y}_0^t \mathbf{f}_j$, i to niezależnie od wybranego wzmocnienia. Innymi słowy, oznacza to wybór wektora \mathbf{f}_j , tworzącego najmniejszy kąt $\phi[\mathbf{y}_0, \mathbf{f}_j]$ z wektorem docelowym \mathbf{y}_0 . Maksymalizowany iloczyn skalarny ma związek z kosinusem tego kąta:

$$\cos\phi[\mathbf{y}_0, \mathbf{f}_j] = \frac{\mathbf{y}_0^t \mathbf{f}_j}{\|\mathbf{y}_0\| \|\mathbf{f}_j\|} \quad (3.38)$$

Ten powszechnie stosowany algorytm modelowania (wybór kształtu minimalizującego kąt z wektorem docelowym, a następnie wybór wzmocnienia najbliższego długości rzutu wektora docelowego na wektor kształtu) jest optymalny tylko przy spełnieniu warunków (3.36), (3.37). Z warunku (3.37) można łatwo zrezygnować: dla ujemnych wzmocnień szuka się kształtu, minimalizującego (ujemny!) iloczyn skalarny $\mathbf{y}_0^t \mathbf{f}_j$, co oznacza, że kąt $\phi[\mathbf{y}_0, \mathbf{f}_j]$ zbliża się do π . Mając zatem w słowniku wzmocnień pary wartości przeciwnych ($\pm g_m$), poszukuje się kształtu maksymalizującego $\cos^2\phi[\mathbf{y}_0, \mathbf{f}_j]$, czyli kwadrat iloczynu skalarnego $\mathbf{y}_0^t \mathbf{f}_j$.

Większe problemy występują w sytuacji, gdy słownik kształtów nie jest znormalizowany. Jest to regułą w koderze CELP: wektory \mathbf{f}_j otrzymuje się drogą filtracji wektorów kształtu sygnału resztkowego \mathbf{c}_j , a filtr zmienia się co 5-30 ms. W tej sytuacji maksymalizacja $\cos^2\phi[\mathbf{y}_0, \mathbf{f}_j]$ zapewnia jedynie suboptymalny wybór wektora kształtu. Niekiedy wyznacza się kilka wektorów zapewniających duże wartości $\cos^2\phi$ i dla każdego wektora wyznacza się odpowiednie wzmocnienie (wybiera się g_m najbliższe g). Ostatecznym kryterium jest błąd $D(i)$ [128].

3.3.2 Wyznaczanie słowników

Jest dany zbiór wektorów sygnału akustycznego, ewentualnie poddanego adaptacyjnej preemfazie i tworzącego bazę danych do projektowania, tzw. sekwencję treningową $\{\mathbf{y}(i)\}$. Poszukuje się słowników kształtów $\{\mathbf{f}_j\}$ (dla kodera CELP $\{\mathbf{c}_j\}$) i wzmocnień $\{g_m\}$. Do projektowania tych słowników, wchodzących w skład kwantyzatora wektorowego typu kształt-wzmocnienie, wykorzystuje się zmodyfikowany algorytm Lloyda [174] przedstawiony na rys.3.8. Mając dane słowniki początkowe $\{\mathbf{f}_j\}$ i $\{g_m\}$ można, uruchamiając SGVQ dla bazy danych $\{\mathbf{y}(i)\}$, dokonać podziału tej bazy danych na dwa sposoby. Po pierwsze, każdemu wektorowi \mathbf{f}_j ze słownika kształtów przypisane są wszystkie wektory bazy danych $\{\mathbf{y}(i)\}$, reprezentowane tym kształtem (tzn. $\mathbf{y}(i) \in P(\mathbf{f}_j)$). Po drugie, każdemu elementowi g_m ze słownika wzmocnień przypisane są wszystkie wektory bazy danych $\{\mathbf{y}(i)\}$, modelowane przy użyciu tego współczynnika wzmocnienia (tzn. $\mathbf{y}(i) \in P(g_m)$). Innymi słowy, jeśli wektor $\mathbf{y}(i)$ z bazy danych jest zamodelowany jako $\tilde{\mathbf{y}}(i) = g_m \mathbf{f}_j$, wówczas $\mathbf{y}(i) \in P(\mathbf{f}_j)$ i jednocześnie $\mathbf{y}(i) \in P(g_m)$. Po dokonaniu podziału możliwa jest modyfikacja słownika kształtów i wzmocnień w taki sposób, aby zmniejszyć błąd kodowania całej sekwencji treningowej $D = \sum_i D(i)$. Ten błąd może być wyrażony w dwojaki sposób: jako

$$D = \sum_{m=1}^{L_g} D_m$$

gdzie

$$D_m = \sum_{i, \mathbf{y}(i) \in P(g_m)} D(i) = \sum_{i, \mathbf{y}(i) \in P(g_m)} \mathbf{y}^t(i) \mathbf{y}(i) - 2g_m \sum_{i, \mathbf{y}(i) \in P(g_m)} \mathbf{y}^t(i) \mathbf{f}_{j(i)} + g_m^2 \sum_{i, \mathbf{y}(i) \in P(g_m)} \mathbf{f}_{j(i)}^t \mathbf{f}_{j(i)} \quad (3.39)$$

oraz jako

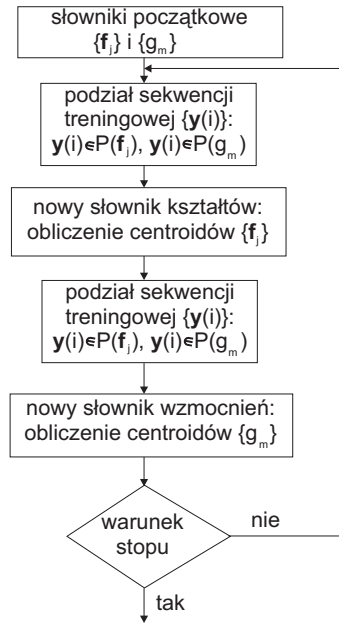
$$D = \sum_{j=1}^{L_s} D_j$$

gdzie

$$D_j = \sum_{i, \mathbf{y}(i) \in P(\mathbf{f}_j)} D(i) = \sum_{i, \mathbf{y}(i) \in P(\mathbf{f}_j)} \mathbf{y}^t(i) \mathbf{y}(i) - 2 \left[\sum_{i, \mathbf{y}(i) \in P(\mathbf{f}_j)} g_{m(i)} \mathbf{y}^t(i) \right] \mathbf{f}_j + \mathbf{f}_j^t \mathbf{f}_j \sum_{i, \mathbf{y}(i) \in P(\mathbf{f}_j)} g_{m(i)}^2 \quad (3.40)$$

W obrębie każdej części bazy danych, wynikającej z ostatnio dokonanego podziału, oblicza się nowy element słownika, minimalizując odpowiednią część sumy $D = \sum_i D(i)$. I tak, dla wszystkich wektorów $\mathbf{y}(i) \in P(\mathbf{f}_j)$ oblicza się nowy wektor \mathbf{f}_j^{nowy} (centroid), przyrównując do zera gradient wyrażenia (3.40): $\frac{\partial D}{\partial \mathbf{f}_j} = \frac{\partial D_j}{\partial \mathbf{f}_j} = \mathbf{0}$. Otrzymuje się:

$$\mathbf{f}_j^{nowy} = \frac{\sum_{i, \mathbf{y}(i) \in P(\mathbf{f}_j)} [g_{m(i)} \mathbf{y}(i)]}{\sum_{i, \mathbf{y}(i) \in P(\mathbf{f}_j)} [g_{m(i)}^2]} \quad (3.41)$$



Rysunek 3.8: Uogólniony algorytm Lloyd'a projektowania słowników SGVQ

gdzie $g_{m(i)}$ - odpowiedni element słownika wzmacnień $\{g_m\}$ przypisany wektorowi $\mathbf{y}(i)$. Podobnie, dla wszystkich wektorów $\mathbf{y}(i) \in P(g_m)$ oblicza się nowy element g_m^{nowy} przyrównując do zera pochodną wyrażenia (3.39): $\frac{\partial D}{\partial g_m} = \frac{\partial D_m}{\partial g_m} = 0$. Otrzymuje się

$$g_m^{nowy} = \frac{\sum_{i, \mathbf{y}(i) \in P(g_m)} [\mathbf{y}^t(i) \mathbf{f}_j(i)]}{\sum_{i, \mathbf{y}(i) \in P(g_m)} [\mathbf{f}_j^t(i) \mathbf{f}_j(i)]} \quad (3.42)$$

Kolejne podziały i modyfikacje słowników wykonuje się aż do uzyskania zbieżności, tj. niemal identycznych słowników $\{\mathbf{f}_j\}$ i $\{g_m\}$ w dwóch kolejnych iteracjach algorytmu z rys.3.8.

W opisanym algorytmie projektowania słowników kształtów i wzmacnień istnieje pewien stopień swobody: wektory kształtu można pomnożyć przez pewien skalar, jeśli elementy słownika wzmacnień podzieli się przez ten sam skalar.

Niekiedy wymusza się silniejsze ograniczenie, normalizując wektory kształtu (wzór (3.36)). Upraszcza się wtedy algorytm przeszukiwania takiego słownika (była o tym mowa w poprzednim punkcie), jednak ograniczenie (3.36) wpływa na algorytm obliczania centroidów (wzór (3.41) już nie obowiązuje). Minimalizacja D z warunkiem $\mathbf{f}_j^t \mathbf{f}_j = 1$ prowadzi do następującego wzoru na centroid:

$$\mathbf{f}_j^{nowy} = \frac{\sum_{i, \mathbf{y}(i) \in P(\mathbf{f}_j)} [g_{m(i)} \mathbf{Y}(i)]}{\left\| \sum_{i, \mathbf{y}(i) \in P(\mathbf{f}_j)} [g_{m(i)} \mathbf{Y}(i)] \right\|} \quad (3.43)$$

Jak to pokazano na rys.3.6, tego typu normalizacja słownika nie jest korzystna dla silnie skorelowanych sygnałów akustycznych. Malone i Fischer [128] w swoim kwantyzatorze wektorowym typu kontur- wzmacnienie (CGVQ) zaproponowali innego typu normalizację:

końce wektorów słownika powinny leżeć nie na powierzchni kuli, lecz na powierzchni o stałej gęstości prawdopodobieństwa wektorów przetwarzanego sygnału. Dla sygnału gaussowskiego o zerowej wartości średniej taką powierzchnią jest elipsoida opisana równaniem

$$\mathbf{y}^t [\mathbf{R}^N(\mathbf{y})]^{-1} \mathbf{y} = \text{const} \quad (3.44)$$

gdzie $\mathbf{R}^N(\mathbf{y}) = E[\mathbf{y}\mathbf{y}^t]$ jest macierzą kowariancji (równą macierzy korelacji) przetwarzanego sygnału.

Minimalizacja D z warunkiem $\mathbf{f}_j^t [\mathbf{R}^N(\mathbf{y})]^{-1} \mathbf{f}_j = 1$ prowadzi do złożonego układu równań, który rozwiązuje się metodami iteracyjnymi lub przybliżonymi [128].

Niekiedy zachodzi potrzeba zaprojektowania słownika kształtów niezależnie od słownika wzmocnień. Przykładem jest zaproponowany przez Sabina [173] SGVQ z adaptacyjnym kwantowaniem wzmocnienia oraz zaproponowany z udziałem autora niniejszej rozprawy SGVQ z wieloma kwantyzatorami wzmocnienia, doboranymi ze względu na wybrany kształt [52, 64]. Projektując słownik kształtów zakłada się, że błąd kwantowania wzmocnienia jest zerowy, tzn. we wzorze (3.35) podstawia się $g_m = g$:

$$D(i) = \mathbf{y}^t(i) \mathbf{y}(i) - \frac{[\mathbf{y}^t(i) \mathbf{f}_{j(i)}]^2}{\mathbf{f}_{j(i)}^t \mathbf{f}_{j(i)}} \quad (3.45)$$

Jeśli projektuje się słownik znormalizowany, to mianownik ułamka we wzorze (3.45) przyjmuje wartość jednostkową. Do projektowania wykorzystuje się algorytm Lloyd'a z rys.3.2. Należy zmienić jedynie zasadę podziału sekwencji treningowej na komórki Voronoi'a. Wybierając $\mathbf{f}_{j(i)}$ należy maksymalizować ostatni człon wyrażenia (3.45), czyli minimalizować kąt ostry $\phi[\mathbf{y}(i), \mathbf{f}_{j(i)}]$. Taka powinna być też zasada podziału sekwencji treningowej: do komórki $P(\mathbf{f}_j)$ należą wektory $\mathbf{y}(i)$ tworzące najmniejszy kąt z \mathbf{f}_j : $\mathbf{y}(i) \in P(\mathbf{f}_j)$, gdy $\phi[\mathbf{y}(i), \mathbf{f}_j] < \phi[\mathbf{y}(i), \mathbf{f}_k]$, $k \neq j$. W procesie projektowania minimalizuje się błąd kwantowania sekwencji treningowej:

$$D = \sum_{j=1}^{L_s} D_j$$

gdzie

$$D_j = \sum_{i, \mathbf{y}(i) \in P(\mathbf{f}_j)} D(i) = \sum_{i, \mathbf{y}(i) \in P(\mathbf{f}_j)} \mathbf{y}^t(i) \mathbf{y}(i) - \sum_{i, \mathbf{y}(i) \in P(\mathbf{f}_j)} [\mathbf{y}^t(i) \mathbf{f}_j]^2 \quad (3.46)$$

Oznacza to, że centroid \mathbf{f}_j powinien maksymalizować odjemnik w wyrażeniu (3.46):

$$Q_j = \sum_{i, \mathbf{y}(i) \in P(\mathbf{f}_j)} [\mathbf{y}^t(i) \mathbf{f}_j]^2 = \sum_{i, \mathbf{y}(i) \in P(\mathbf{f}_j)} \mathbf{f}_j^t \mathbf{y}(i) \mathbf{y}^t(i) \mathbf{f}_j = \mathbf{f}_j^t \left[\sum_{i, \mathbf{y}(i) \in P(\mathbf{f}_j)} \mathbf{y}(i) \mathbf{y}^t(i) \right] \mathbf{f}_j = \mathbf{f}_j^t \mathbf{R}_j^N(\mathbf{y}) \mathbf{f}_j \quad (3.47)$$

gdzie $\mathbf{R}_j^N(\mathbf{y})$ jest macierzą korelacji wektorów $\mathbf{y}(i) \in P(\mathbf{f}_j)$ czyli tworzących najmniejszy kąt z wektorem \mathbf{f}_j obliczonym w poprzedniej iteracji algorytmu Lloyd'a. Maksymalizacja Q_j z ograniczeniem $\|\mathbf{f}_j\| = 1$ prowadzi do następującego równania:

$$\mathbf{R}_j^N(\mathbf{y}) \mathbf{f}_j = Q_j \mathbf{f}_j \quad (3.48)$$

Rozwiązaniem jest wektor własny macierzy $\mathbf{R}_j^N(\mathbf{y})$ odpowiadający największej wartości własnej Q_j , czyli oś główna zbioru wektorów $\mathbf{y}(i) \in P(\mathbf{f}_j)$. Oś główną można znaleźć wykonując następujące iteracje:

$$\mathbf{f}_j^{(k+1)} = \frac{\mathbf{R}_j^N(\mathbf{y}) \mathbf{f}_j^{(k)}}{\|\mathbf{R}_j^N(\mathbf{y}) \mathbf{f}_j^{(k)}\|} \quad k = 1, 2, \dots \quad (3.49)$$

Za $\mathbf{f}_j^{(1)}$ należy podstawić dowolny wektor nie będący wektorem własnym macierzy $\mathbf{R}_j^N(\mathbf{y})$.

Zagadnienie projektowania słowników dla koderów CELP ma mniejsze znaczenie, ze względu na adaptacyjną filtrację słownika kształtów sygnału resztkowego $\{\mathbf{c}_j\}$, $j = 1, \dots, L_s$, z wykorzystaniem filtru $H'(z)$. Filtracja zapewnia "dopasowanie się" słownika do właściwości widmowych sygnału, w związku z tym nie jest konieczne dalsze "dopasowywanie" słownika do sygnału w procesie projektowania. W typowym koderze CELP nie projektuje się słownika kształtów sygnału resztkowego, nadając mu z góry narzuconą strukturę (rzadką, algebraiczną, nakładkową, itp.), umożliwiającą szybką jego filtrację i przeszukiwanie. W niektórych koderach CELP przeprowadza się jednak proces projektowania słownika; pierwszą publikacją na ten temat była praca [38].

Przystępując do projektowania słowników kształtów i wzmocnień dla kodera CELP należy wykorzystać wzór (3.34) na energię błędu kwantyzacji dla wektora sygnału percepcyjnego $\mathbf{y}(i)$:

$$D(i) = \|\mathbf{y}_0(i) - g_{m(i)} \mathbf{H}'(i) \mathbf{c}_{j(i)}\|^2 \quad (3.50)$$

Należy zwrócić uwagę na fakt, że macierz filtracji $\mathbf{H}'(i)$ zmienia się co 10-30 ms, wobec tego staje się funkcją indeksu przetwarzanego wektora sygnału. Wektor sygnału docelowego $\mathbf{y}_0(i) = \mathbf{y}(i) - \tilde{\mathbf{y}}_0(i)$ zależy, poprzez "ringing" $\tilde{\mathbf{y}}_0(i)$, od słowników kształtu i wzmocnienia sygnału resztkowego. Należy mieć to na uwadze w procesie projektowania tych słowników, przeprowadzając projektowanie w tzw. układzie zamkniętym (po każdej iteracji, w której otrzymuje się nowe słowniki, należy wyznaczyć nowe wektory docelowe $\mathbf{y}_0(i)$).

Słowniki kształtów $\{\mathbf{c}_j\}$, $j = 1, \dots, L_s$ i wzmocnień $\{g_m\}$, $j = 1, \dots, L_g$ projektuje się w oparciu o frazę treningową, z której otrzymuje się bazę wektorów docelowych $\{\mathbf{y}_0(i)\}$ i macierzy filtracji $\{\mathbf{H}'(i)\}$. Podobnie jak w przypadku SGVQ, wykorzystuje się zmodyfikowany algorytm Lloyda [174] przedstawiony na rys.3.8. Dąży się do minimalizacji energii błędu kwantowania dla całej frazy treningowej,

$$\begin{aligned} D &= \sum_i D(i) = \sum_i [\mathbf{y}_0(i) - g_{m(i)} \mathbf{H}'(i) \mathbf{c}_{j(i)}]^t [\mathbf{y}_0(i) - g_{m(i)} \mathbf{H}'(i) \mathbf{c}_{j(i)}] \\ &= \sum_i [\mathbf{y}_0^t(i) \mathbf{y}_0(i) - 2 g_{m(i)} \mathbf{y}_0^t(i) \mathbf{H}'(i) \mathbf{c}_{j(i)} + g_{m(i)}^2 \mathbf{c}_{j(i)}^t \mathbf{H}'^t(i) \mathbf{H}'(i) \mathbf{c}_{j(i)}] \end{aligned} \quad (3.51)$$

przedstawiając ją w postaci analogicznej do (3.39) i (3.40). I tak, z zapisu

$$D = \sum_{m=1}^{L_g} D_m$$

gdzie

$$D_m = \sum_{i, \mathbf{y}_0(i) \in P(g_m)} D(i)$$

$$\begin{aligned}
&= \sum_{i, \mathbf{y}_0(i) \in P(g_m)} \mathbf{y}_0^t(i) \mathbf{y}_0(i) - 2 g_m \sum_{i, \mathbf{y}_0(i) \in P(g_m)} \mathbf{y}_0^t(i) \mathbf{H}'(i) \mathbf{c}_{j(i)} \\
&+ g_m^2 \sum_{i, \mathbf{y}_0(i) \in P(g_m)} \mathbf{c}_{j(i)}^t \mathbf{H}^{t'}(i) \mathbf{H}'(i) \mathbf{c}_{j(i)}
\end{aligned} \tag{3.52}$$

otrzymuje się, różniczkując względem g_m , wzór na centroid, stanowiący element nowego słownika wzmocnień:

$$g_m^{nowy} = \frac{\sum_{i, \mathbf{y}_0(i) \in P(g_m)} [\mathbf{y}_0^t(i) \mathbf{H}'(i) \mathbf{c}_{j(i)}]}{\sum_{i, \mathbf{y}_0(i) \in P(g_m)} [\mathbf{c}_{j(i)}^t \mathbf{H}^{t'}(i) \mathbf{H}'(i) \mathbf{c}_{j(i)}]} \tag{3.53}$$

Analogicznie, z zapisu

$$D = \sum_{j=1}^{L_s} D_j$$

gdzie

$$\begin{aligned}
D_j &= \sum_{i, \mathbf{y}_0(i) \in P(\mathbf{c}_j)} D(i) \\
&= \sum_{i, \mathbf{y}_0(i) \in P(\mathbf{c}_j)} \mathbf{y}_0^t(i) \mathbf{y}_0(i) - 2 \left[\sum_{i, \mathbf{y}_0(i) \in P(\mathbf{c}_j)} g_{m(i)} \mathbf{y}_0^t(i) \mathbf{H}'(i) \right] \mathbf{c}_j \\
&+ \mathbf{c}_j^t \left[\sum_{i, \mathbf{y}_0(i) \in P(\mathbf{c}_j)} g_{m(i)}^2 \mathbf{H}^{t'}(i) \mathbf{H}'(i) \right] \mathbf{c}_j
\end{aligned} \tag{3.54}$$

otrzymuje się, przyrównując do zera gradient względem \mathbf{c}_j :

$$\frac{\partial D_j}{\partial \mathbf{c}_j} = -2 \sum_{i, \mathbf{y}_0(i) \in P(\mathbf{c}_j)} g_{m(i)} \mathbf{H}^{t'}(i) \mathbf{y}_0(i) + 2 \left[\sum_{i, \mathbf{y}_0(i) \in P(\mathbf{c}_j)} g_{m(i)}^2 \mathbf{H}^{t'}(i) \mathbf{H}'(i) \right] \mathbf{c}_j = \mathbf{0}$$

układ równań na centroid, stanowiący element nowego słownika kształtów:

$$\left[\sum_{i, \mathbf{y}_0(i) \in P(\mathbf{c}_j)} g_{m(i)}^2 \mathbf{H}^{t'}(i) \mathbf{H}'(i) \right] \mathbf{c}_j^{nowy} = \sum_{i, \mathbf{y}_0(i) \in P(\mathbf{c}_j)} g_{m(i)} \mathbf{H}^{t'}(i) \mathbf{y}_0(i) \tag{3.55}$$

Jak wynika z badań Davidsona, Yong i Gersho, algorytm Lloyda jest w tym wypadku wolno zbieżny, zwłaszcza w sytuacji, gdy w pierwszym etapie modelowania stosuje się słownik adaptacyjny (predykcję długookresową). Predykcja długookresowa zależy, podobnie jak "ringing", od słowników, co utrudnia proces ich projektowania, wymuszając działanie "w pętli zamkniętej".

3.3.3 Porównanie wariantów SGVQ

Autor wraz z A.Chmielewskim i S.Ratree przeprowadził symulacyjne badania porównawcze różnych wariantów SGVQ oraz zaproponował SGVQ z wieloma kwantyzatorami wzmocnienia dobieranymi ze względu na wybrany kształt [52, 64]. Badania przeprowadzono z wykorzystaniem trzech fraz sygnału mowy o jakości telefonicznej (częstotliwość próbkowania 8 kHz): frazy treningowej (użytej do projektowania słowników), innej frazy i frazy treningowej stłumionej o 20 dB. Założono wymiar przetwarzanych wektorów $N = 8$, co odpowiada 1 ms czasu trwania sygnału. Badano SGVQ o przepływności binarnej 13 kbit/s, co oznacza, że na zakodowanie jednego wektora przypadało 13 bitów.

Jako wskaźnik jakości skwantowanej mowy wykorzystano SNR_{seg} - stosunek mocy sygnału do mocy szumu kwantyzacji w ujęciu segmentowym (wzór 1.1). Jest to wartość średnia SNR obliczanego w segmentach (przyjęto długość segmentu równą 128 próbek) i wyrażonego w dB. Segmenty zawierające ciszę międzywyrazową nie były brane pod uwagę (próg ciszy wyznaczono 43 dB poniżej poziomu sygnału sinusoidalnego przesterowującego przetwornik A/C w karcie dźwiękowej komputera).

W pierwszym etapie rozdzielono 13 bitów między kształt i wzmocnienie - tab.3.1.

Tabela 3.1: Rozdział bitów między kształt i wzmocnienie - SNR_{seg} dla standardowego SGVQ

L_s	L_g	SNR_{seg}
128	64	13.58
256	32	14.16
512	16	13.28

Najlepsze rezultaty otrzymano biorąc 8 bitów na kodowanie kształtu ($L_s = 256$) i 5 bitów na kodowanie wzmocnienia ($L_g = 32$). Taki właśnie rozdział bitów był zachowany w trakcie symulacji zdokumentowanych w tab.3.2. Dla porównania podano wyniki symulacji SGVQ bez kwantowania wzmocnienia (zamiast skwantowanej wartości g_m podstawiono wartość dokładną $g = \frac{\mathbf{y}^t \mathbf{f}_j}{\mathbf{f}_j^t \mathbf{f}_j}$).

Dla standardowego SGVQ przetestowano dwa algorytmy projektowania słowników: zmodyfikowany algorytm Lloyd'a, w którym centroidy są obliczane ze wzorów (3.42), (3.43), oraz, zaproponowany z udziałem autora, algorytm oparty na wyznaczaniu osi głównej zbioru wektorów (wzór (3.48)). Oba algorytmy dały zbliżone rezultaty, przy czym ten drugi jest znacznie mniej złożony obliczeniowo (słownik kształtów projektuje się tu niezależnie od słownika wzmocnienia). Zmodyfikowany algorytm Lloyd'a daje zauważalnie lepsze rezultaty, gdy liczba poziomów kwantyzacji wzmocnienia jest mała (rzędu 2-8).

Jak pokazano na rys.3.6, standardowy SGVQ ze znormalizowanym słownikiem kształtów nie jest właściwym koderem dla silnie skorelowanego sygnału mowy. W kwantyzatorze typu kontur - wzmocnienie (CGVQ) długość wektorów dobiera się w ten sposób, że ich końce leżą na powierzchni o stałej gęstości prawdopodobieństwa wektorów przetwarzanego sygnału (założono, że jest to elipsoida). Otrzymano wyraźnie lepsze wyniki niż dla standardowego SGVQ - tab.3.2.

Tabela 3.2: Porównanie wariantów SGVQ - SNR_{seg} dla trzech fraz

kwantyzator	kbit/s	fraz		
		treningowa	inna	treningowa stłumiona o 20dB
standardowy SGVQ bez kwantowania wzmocnienia	-	15.48	12.56	15.44
standardowy SGVQ SGVQ - równomierne	13	14.16	10.74	6.24
kwantyzatory wzmocnienia SGVQ - nierównomierne	13	14.97	10.91	8.49
kwantyzatory wzmocnienia CGVQ	13	15.25	11.16	13.05
kwantyzatory wzmocnienia SGVQ - adaptacyjne	13	14.79	12.02	11.85
kwantowanie wzmocnienia	13.6	14.20	10.43	14.17

Alternatywnym podejściem, zaproponowanym z udziałem autora, jest zastosowanie, dla każdego znormalizowanego kształtu \mathbf{f}_j , innego słownika wzmocnienia $\{g_m^{(j)}\}$ - [52, 64]. Proces modelowania sygnału \mathbf{y} przebiega następująco: Dla każdego kształtu \mathbf{f}_j wybierane jest ze słownika $\{g_m^{(j)}\}$ wzmocnienie $g_m^{(j)}$ najbliższe wartości niekwantowanej $g = \frac{\mathbf{y}^t \mathbf{f}_j}{\mathbf{f}_j^t \mathbf{f}_j} = \mathbf{y}^t \mathbf{f}_j$ - wzór (3.35). Para $\mathbf{f}_j, g_m^{(j)}$ wybierana jest ze względu na minimum energii błędu kwantowania $D = \|\mathbf{y} - g_m^{(j)} \mathbf{f}_j\|$. Słownik kształtów projektuje się według algorytmu opartego na wyznaczaniu osi głównej zbioru wektorów (wzór (3.48)). W pierwszym etapie stosowano równomierne kwantyzatory wzmocnień, dobierając indywidualnie zakres pracy dla każdego z nich. Zakres pracy był równy maksymalnej, niekwantowanej wartości wzmocnienia dla danego kształtu. Lepsze wyniki dało projektowanie (z wykorzystaniem "jednowymiarowej" wersji algorytmu Lloyda) nierównomiernych kwantyzatorów wzmocnień - tab.3.2. Za poprawę SNR_{seg} "płaci się" jednak koniecznością pamiętania poziomów kwantyzacji L_s kwantyzatorów wzmocnienia.

Omówione warianty SGVQ nie są odporne na spadek poziomu kwantowanego sygnału. Dla frazy stłumionej o 20 dB następuje wyraźny spadek SNR_{seg} . Aby temu zapobiec, zastosowano adaptacyjne kwantowanie wzmocnienia. Początkowo testowano adaptację "wstecz", proponowaną w [174]. Adaptacja taka jest wykorzystywana w telefonicznych systemach ADPCM, gdzie zakres pracy kwantyzatora jest korygowany co próbkę. W kwantyzatorach wzmocnienia w SGVQ, w których kwantyzacja odbywa się raz na N próbek, zmiany wzmocnienia są zbyt szybkie i kwantyzator za nimi nie nadąża (często ulega przesterowaniu). W tej sytuacji zdecydowano się na adaptację kwantyzatora metodą "w przód" [52, 64]. Każdy wektor \mathbf{y} jest modelowany w następujący sposób:

$$\tilde{\mathbf{y}} = g_m^{(j)} \hat{\sigma}_y \mathbf{f}_j \quad (3.56)$$

gdzie $\hat{\sigma}_y$ jest parametrem adaptacji (oszacowanie wartości skutecznej, lub średniej wartości

normy wektora \mathbf{y}). Parametr ten jest obliczany w segmentach (np. co 10 ms), kwantowany i przesyłany do odbiornika. W procesie modelowania sygnału, najpierw wybiera się kształt minimalizujący kąt $\phi(\mathbf{y}, \mathbf{f}_j)$, następnie wybierane jest wzmocnienie ze słownika $\{g_m^{(j)}\}$, najmniej odległe od niekwantowanej wartości $g/\hat{\sigma}_y = \mathbf{y}^t \mathbf{f}_j/\hat{\sigma}_y$. Adaptacja wpływa na projektowanie słownika wzmocnień: jako dane treningowe należy wykorzystać znormalizowane wzmocnienia $g/\hat{\sigma}_y$. Zaprojektowany w ten sposób SGVQ jest bardzo odporny na zmiany poziomu sygnału (tab.3.2), jednak płaci się za to zwiększonym opóźnieniem oraz niewielkim powiększeniem przepływności binarnej (parametr adaptacji $\hat{\sigma}_y$ musi być transmitowany).

3.4 Projektowanie koderów z modelowaniem wielostopniowym

3.4.1 Wprowadzenie

Kwantyzator wektorowy z modelowaniem wielostopniowym został przedstawiony w punkcie 3.2.3, a wielostopniowy koder CELP w punkcie 3.2.4. Typowy koder CELP zawiera dwa etapy (stopnie) modelowania: w pierwszym etapie, tzw. predykcji długookresowej, wykorzystywany jest słownik adaptacyjny C^1 , a w drugim etapie słownik stochastyczny C^2 . W przypadku kodera CELP o pobudzeniu impulsowym, można wyróżnić nawet około $K = 10$ etapów modelowania - w każdym etapie do sygnału pobudzającego dodawany jest jeden impuls.

W koderze wielostopniowym można stosować różne algorytmy modelowania sygnału, tj. wyboru wektorów w kolejnych etapach kodowania (każdorazowo wybierany jest jeden wektor). Zostaną one przedstawione w punktach 3.4.3 i 3.4.4. Algorytm "standardowy" stosowany powszechnie we wczesnych wersjach kodera CELP charakteryzuje się prostotą, jednak jest daleki od optymalności. N.Moreau z autorem niniejszej rozprawy [141] zaproponowali algorytm modelowania mający cechy lokalnej optymalności, polegający na ortogonalizacji słownika względem wektorów wybranych w poprzednich etapach. Później tego typu algorytmy były stosowane m.in. przez Johnsona, Taniguchi'ego, Ohtę [104, 105, 193], Gersona i Jasiuka [78], Miki'ego, Mano, Ohmuro i Moriy'ę [138], oraz Pati'ego, Rezaifara i Krishna-prasada [162]. Algorytm lokalnie optymalny może mieć wiele równoważnych pod względem jakości, ale różniących się pod względem złożoności obliczeniowej implementacji. W kolejnych pracach Dymarski, Moreau i Vigier [60], Dymarski, Moreau i Vos [61], Dymarski i Moreau [142, 143, 57, 145] porównali złożoność obliczeniową kilku wariantów algorytmu lokalnie optymalnego (bezpośrednia ortogonalizacja słownika, zmodyfikowany algorytm Grama - Schmidta, dekompozycja Choleskiego, dekompozycja QR, tzw. filtracja odwrotna) i zaproponowali wersję najbardziej efektywną: rekursywny algorytm Grama - Schmidta. Złożoność obliczeniową badali dla słowników o strukturze "gęstej" i "rzadkiej". Jednocześnie badali jakość sygnału mowy uzyskiwaną przy użyciu lokalnie optymalnego algorytmu modelowania i algorytmów nie mających właściwości lokalnej optymalności. W artykule [145] N.Moreau i autor niniejszej rozprawy przedstawili dwa modele sygnału stosowane w wielostopniowym koderze CELP: najczęściej stosowany model oparty na kombinacji liniowej wektorów i model oparty na sumie wektorów pomnożonej przez pojedynczy współczynnik wzmocnienia (odpowiadający np. pobudzeniu "algebraicznemu"). Dla obu modeli zaproponowali lokalnie optymalne algorytmy modelowania. W punkcie 3.4.6 przedstawione będą wyniki badań porównawczych (jakość mowy i złożoność obliczeniowa) tych wariantów kodera CELP w warunkach tej samej szybkości transmisji (rzędu 9kbit/s).

W pracy [59] Dymarski, Moreau i Chmielewski zastosowali w kwantyzatorze wektorowym algorytmy, które Moreau i Dymarski zaproponowali uprzednio dla kodera CELP [140, 141]:

utworzenie ze słowników, adaptacyjnego i stałego tzw. słownika mieszanego i wybieranie z niego wektorów z wykorzystaniem sukcesywnej ortogonalizacji słownika. W wielostopniowych kwantyzatorach wektorowych, podobnie jak w koderach CELP, można również realizować predykcję długookresową, mnożąc wybrany wektor ze słownika adaptacyjnego przez odpowiedni współczynnik wzmocnienia. Tego typu koder był badany m.in. przez Dymarskiego i Chmielewskiego [51]. Jest on szczególnym przypadkiem wielostopniowej (ang. *multistage*) kwantyzacji wektorowej, badanej przez Juanga i Gray'a [108], patrz też [77].

W wielostopniowym kwantyzatorze wektorowym słowniki stałe powinny być projektowane, dla danej sekwencji treningowej sygnału mowy - patrz np. praca Dymarskiego i Ratre [64]. Dla koderów CELP rzadko projektuje się słowniki (podstawy ich projektowania opublikowali Davidson, Yong i Gersho w [38]). Dla tego typu koderów Moreau i Dymarski [140, 141] zaproponowali stworzenie tzw. pobudzenia mieszanego: w jednym słowniku umieścili wektory adaptowane (reprezentujące pobudzenie wygenerowane w przeszłości), pojedyncze impulsy (jak w MP-LPC) i sekwencje szumu (jak w klasycznym CELP). O wyborze wektorów ze słownika pobudzenia mieszanego decyduje wyłącznie energia błędu kodowania. Połączenie słownika impulsowego i stochastycznego rozważali później Granzow, Atal, Paliwal i Schroeter [84]. Chmielewski i Dymarski [31] zaproponowali dołączenie pewnej liczby wektorów stałych do słownika adaptacyjnego, przetwarzanego w pierwszym etapie kodowania. Poprawia to jakość mowy w transjentach, gdy słownik adaptacyjny zawiera sekwencje niedopasowane do kodowanego wektora mowy. Innym rozwiązaniem jest użycie różnych słowników dla mowy dźwięcznej, bezdźwięcznej, transjentów (tzw. CELP z klasyfikacją fonetyczną, patrz Wang i Gersho [202], Paksoy, Srinivasan i Gersho [160], Lupini, Hassanein i Cuperman [125], Dymarski [50]).

Zagadnienia projektowania słowników dla wielostopniowych VQ i CELP zostaną omówione w p.3.4.5.

3.4.2 Przeszukiwanie słowników i tworzenie modelu sygnału mowy - sformułowanie zadania

W kwantyzatorach wektorowych i koderach CELP z modelowaniem wielostopniowym, do modelowania sygnału percepcyjnego jest wykorzystywany schemat kształt - wzmocnienie z osobnym wzmocnieniem dla każdego wybranego wektora (wzór (3.28)):

$$\tilde{\mathbf{y}} = \sum_{k=1}^K g_{m_k}^k \mathbf{f}_{j_k}^k + \tilde{\mathbf{y}}_0$$

lub z pojedynczym wzmocnieniem dla sumy wektorów (wzór (3.30)):

$$\tilde{\mathbf{y}} = g \sum_{k=1}^K s^k \mathbf{f}_{j_k}^k + \tilde{\mathbf{y}}_0$$

W kwantyzatorze wektorowym należy pominąć "ringing" $\tilde{\mathbf{y}}_0$, natomiast w CELP należy uwzględnić, że $\mathbf{f}_{j_k}^k = \mathbf{H}' \mathbf{c}_{j_k}^k$ jest odfiltrowanym wektorem $\mathbf{c}_{j_k}^k$ ze słownika \mathbf{C}^k .

Zadanie modelowania dla i -go wektora sygnału percepcyjnego $\mathbf{y}(i)$ (dla uproszczenia indeks " i " będzie pomijany) polega na znalezieniu wektorów $\mathbf{f}_{j_k}^k$, $k = 1 \dots K$ i odpowiadających im wzmocnień $g_{m_k}^k$ (lub $g s^k$) tak, aby błąd modelowania: $D(i) = \|\mathbf{y} - \tilde{\mathbf{y}}\|^2$ był jak najmniejszy. Podstawiając $\tilde{\mathbf{y}}$ ze wzoru (3.28) otrzymuje się

$$D(i) = \|\mathbf{y}_0 - \sum_{k=1}^K g_{m_k}^k \mathbf{f}_{j_k}^k\|^2 \quad (3.57)$$

gdzie $\mathbf{y}_0 = \mathbf{y} - \tilde{\mathbf{y}}_0$ jest wektorem docelowym (ang. *target vector*). Analogicznie, podstawiając $\tilde{\mathbf{y}}$ ze wzoru 3.30 otrzymuje się

$$D(i) = \|\mathbf{y}_0 - g \sum_{k=1}^K s^k \mathbf{f}_{j_k}^k\|^2 \quad (3.58)$$

Modelowanie wymaga przeszukania K słowników kształtów sygnału percepcyjnego $\mathbf{F}^1, \dots, \mathbf{F}^K$ (kwantyzację wzmocnień też można interpretować jako przeszukiwanie słowników zawierających skalary - poziomy kwantyzacji). Jednoczesne przeszukiwanie tych słowników, tzn. testowanie wszystkich możliwych kombinacji, nie jest możliwe ze względu na dużą złożoność obliczeniową. Należy w związku z tym poszukiwać algorytmów suboptymalnych, umożliwiających uzyskanie dobrego modelu $\tilde{\mathbf{y}}$ przy wykonywaniu stosunkowo niewielkiej liczby operacji arytmetycznych.

W p.3.4.3 zostaną omówione takie algorytmy dla modelu opartego na kombinacji liniowej wektorów (minimalizacja kryterium (3.57)), a w p.3.4.4 dla modelu opartego na sumie (lub różnicy) wektorów pomnożonej przez pojedynczy współczynnik wzmocnienia (minimalizacja kryterium (3.58)).

3.4.3 Lokalnie optymalne algorytmy modelowania wielostopniowego

Jak już wspomniano w poprzednim punkcie, istnieje optymalny algorytm wyboru kształtów sygnału percepcyjnego $\mathbf{f}_{j_k}^k$ i odpowiadających im wzmocnień $g_{m_k}^k$, gwarantujący minimum błędu modelowania (3.57). Wymaga on jednak sprawdzenia wszystkich możliwości wyboru kształtów i wzmocnień z odpowiednich słowników, co jest nierealne (z wyłączeniem najprostszyc przypadków) ze względu na dużą złożoność obliczeniową. W związku z tym realizuje się algorytmy iteracyjne, polegające na tym, że w każdym z K etapów modelowania do modelu dołącza się nowy wektor kształtu.

Takie algorytmy są z reguły suboptymalne (nie gwarantują minimum błędu modelowania). Wyjątkiem jest sytuacja, gdy wybierane w kolejnych etapach wektory są ortogonalne w stosunku do wektorów wybieranych w innych etapach. Rozważmy przypadek, gdy wszystkie wektory wybierane są ze słownika \mathbf{F} , zawierającego wektory ortonormalne. Takich N - wymiarowych wektorów może być co najwyżej N . Modelowanie polega na rzutowaniu wektora docelowego \mathbf{y}_0 na wybierane wektory ze słownika - odpowiednie (nieskwantowane)

wzmocnienia wynoszą: $g_k = [\mathbf{f}_{j_k}]^t \mathbf{y}_0$. Odpowiada to wyznaczeniu K niezależnych od siebie (ze względu na ortogonalność wektorów słownika) modeli typu kształt - wzmocnienie, według zasad opisanych w p.3.3.1. W szczególności, w każdym etapie wybiera się wektor \mathbf{f}_{j_k} , tworzący najmniejszy kąt ostry ϕ (maksimum $\cos^2 \phi$) z wektorem docelowym \mathbf{y}_0 . Jeżeli wykorzystana są wszystkie ($K = N$) wektory ortonormalnego słownika \mathbf{F} , ($\mathbf{F}^t \mathbf{F} = \mathbf{I}$), to wówczas wzmocnienia g_k można uważać za składowe przetransformowanego wektora \mathbf{y}_0 : $\mathbf{g} = \mathbf{F}^t \mathbf{y}_0$. Kwantyzację wzmocnień można zinterpretować jako kodowanie współczynników transformaty, w wyniku kwantowania otrzymuje się wektor $\tilde{\mathbf{g}}$. Mnożąc wektory słownika \mathbf{F} przez skwantowane wzmocnienia otrzymuje się model $\tilde{\mathbf{y}} = \mathbf{F} \tilde{\mathbf{g}}$. Gdyby wzmocnienia nie były kwantowane, to modelowanie byłoby bezbłędne:

$$\tilde{\mathbf{y}} = \mathbf{F} \mathbf{g} = \mathbf{F} \mathbf{F}^t \mathbf{y}_0 = \mathbf{I} \mathbf{y}_0 = \mathbf{y}_0 \quad (3.59)$$

Do problemu kodowania transformaty powrócimy w dalszej części rozprawy.

W ogólnym przypadku słowniki nie zawierają ortogonalnych wektorów i algorytmy iteracyjne budowania modelu $\tilde{\mathbf{y}}$ mają charakter suboptymalny. Najprostszy algorytm, który będzie tu nazywany **standardowym algorytmem iteracyjnym**, polega na rzutowaniu, na wybrany ze słownika wektor, błędu otrzymanego w poprzednim etapie modelowania. Po $k - 1$ -szym etapie wektor błędu wynosi:

$$\mathbf{y}_{k-1} = \mathbf{y}_0 - \sum_{i=1}^{k-1} g_{m_i}^i \mathbf{f}_{j_i}^i \quad (3.60)$$

W k -tym etapie poszukuje się $\mathbf{f}_{j_k}^k$ i $g_{m_k}^k$, minimalizujących kwadrat normy wektora błędu:

$$\|\mathbf{y}_k\|^2 = \|\mathbf{y}_0 - \sum_{i=1}^k g_{m_i}^i \mathbf{f}_{j_i}^i\|^2 = \|\mathbf{y}_{k-1} - g_{m_k}^k \mathbf{f}_{j_k}^k\|^2 \quad (3.61)$$

Jest to zagadnienie modelowania wektorowego typu kształt - wzmocnienie, którego rozwiązanie omówiono w p.3.3.1. Wynika ono z przepisania wzoru (3.61) w postaci analogicznej do (3.35):

$$\|\mathbf{y}_k\|^2 = \mathbf{y}_{k-1}^t \mathbf{y}_{k-1} - \frac{[\mathbf{y}_{k-1}^t \mathbf{f}_{j_k}^k]^2}{\|\mathbf{f}_{j_k}^k\|^2} + \|\mathbf{f}_{j_k}^k\|^2 [g_{m_k}^k - g^k]^2 \quad (3.62)$$

gdzie $g^k = \frac{\mathbf{y}_{k-1}^t \mathbf{f}_{j_k}^k}{\|\mathbf{f}_{j_k}^k\|^2}$ jest dokładnym (nieskwantowanym) współczynnikiem wzmocnienia w k -tym etapie modelowania.

Ze słownika \mathbf{F}^k wybrany zostanie ten wektor, który tworzy minimalny kąt ostry ϕ z wektorem \mathbf{y}_{k-1} , co oznacza maksymalizację kryterium:

$$\cos^2 \phi = \frac{[\mathbf{y}_{k-1}^t \mathbf{f}_{j_k}^k]^2}{\|\mathbf{f}_{j_k}^k\|^2 \|\mathbf{y}_{k-1}\|^2} \quad (3.63)$$

Czynnik $\|\mathbf{y}_{k-1}\|^2$ w (3.63) można pominąć, gdyż nie zmienia się on dla kolejnych, testowanych wektorów słownika \mathbf{F}^k .

Dla wybranego wektora $\mathbf{f}_{j_k}^k$ wyznacza się dokładną wartość wzmocnienia g^k , którą następnie kwantuje się, wybierając najbliższy poziom kwantyzacji $g_{m_k}^k$.

W sytuacji, gdy wszystkie wektory pochodzą z tego samego słownika \mathbf{F} (np. modelowanie wieloimpulsowe), można otrzymać równoważną postać przedstawionego algorytmu nie wymagającą przetwarzania wektorów, tylko ich iloczynów skalarnych (korelacji wzajemnych). Zależność $\mathbf{y}_k = \mathbf{y}_{k-1} - g_{m_k}^k \mathbf{f}_{j_k}^k$ można przepisać w postaci

$$\mathbf{y}_k = \mathbf{y}_{k-1} - g_{m_k} \mathbf{f}_{j_k} = \mathbf{y}_{k-1} - \frac{\mathbf{f}_{j_k}^t \mathbf{y}_{k-1}}{\|\mathbf{f}_{j_k}\|^2} \mathbf{f}_{j_k},$$

w której górny indeks pominięto, gdyż w każdym etapie przeszukuje się ten sam słownik kształtów i wzmocnień, oraz podstawiono niekwantowaną wartość wzmocnienia g^k . Po obustronnym pomnożeniu przez transponowany wektor $\mathbf{f}_j \in \mathbf{F}$ otrzymuje się wzór, według którego aktualizuje się iloczyny skalarne:

$$\mathbf{f}_j^t \mathbf{y}_k = \mathbf{f}_j^t \mathbf{y}_{k-1} - \frac{\mathbf{f}_j^t \mathbf{f}_{j_k}}{\|\mathbf{f}_{j_k}\|^2} \mathbf{f}_{j_k}^t \mathbf{y}_{k-1} \quad (3.64)$$

Algorytm przetwarzania korelacji wzajemnych został zaproponowany przez Atala i Remde [7] i zastosowany w [9] do tworzenia pobudzenia impulsowego. Można przedstawić go w następujący sposób:

Mając dany słownik \mathbf{F} i wektor docelowy \mathbf{y}_0 oblicza się dla każdego wektora $\mathbf{f}_j \in \mathbf{F}$ iloczyny skalarne $\mathbf{f}_j^t \mathbf{y}_0$ i $\mathbf{f}_j^t \mathbf{f}_j$.

Dla $k = 1, \dots, K$:

- wybiera się wektor o indeksie j_k , zapewniający maksimum wartości $\frac{[\mathbf{f}_j^t \mathbf{y}_{k-1}]^2}{\mathbf{f}_j^t \mathbf{f}_j}$
- dla wybranego wektora oblicza się niekwantowane wzmocnienie: $g^k = \frac{\mathbf{f}_{j_k}^t \mathbf{y}_{k-1}}{\mathbf{f}_{j_k}^t \mathbf{f}_{j_k}}$
- dla każdego wektora \mathbf{f}_j ze słownika oblicza się

$$\mathbf{f}_j^t \mathbf{y}_k = \mathbf{f}_j^t \mathbf{y}_{k-1} - r(k, j) \mathbf{f}_{j_k}^t \mathbf{y}_{k-1}$$

$$\text{gdzie } r(k, j) = \frac{\mathbf{f}_j^t \mathbf{f}_{j_k}}{\mathbf{f}_{j_k}^t \mathbf{f}_{j_k}} = \frac{\mathbf{f}_j^t \mathbf{f}_{j_k}}{\|\mathbf{f}_{j_k}\|^2}$$

W ostatnim etapie, dla $k = K$, punkt końcowy (aktualizacja iloczynów skalarnych) można pominąć.

Standardowy algorytm iteracyjny jest daleki od optymalnego przede wszystkim z tej przyczyny, że otrzymany model $\tilde{\mathbf{y}}$ nie jest w ogólnym przypadku rzutem ortogonalnym wektora docelowego \mathbf{y}_0 na podprzestrzeń rozpiętą na wybranych wektorach $\mathbf{f}_{j_1}^1, \dots, \mathbf{f}_{j_K}^K$. Rzut ortogonalny otrzymuje się jedynie w przypadku, gdy wybrane wektory tworzą bazę ortogonalną K -wymiarowej podprzestrzeni (rys.3.9). Oczywiście można, po zakończeniu algorytmu, obliczyć nowe wartości niekwantowanych wzmocnień g^1, \dots, g^K tak, aby model $\tilde{\mathbf{y}} = \sum_{k=1}^K g^k \mathbf{f}_{j_k}^k$ stanowił rzut ortogonalny \mathbf{y}_0 na podprzestrzeń rozpiętą na wektorach $\mathbf{f}_{j_1}^1, \dots, \mathbf{f}_{j_K}^K$. Dalsze zmniejszenie błędu jest możliwe, gdy opisaną operację wykonuje się po zakończeniu każdego kroku

algorytmu (istnieje wtedy większa szansa na wybór lepszego wektora w następnym kroku). Znaczną poprawę osiągnięto dopiero dzięki modyfikacji zaproponowanej przez N. Moreau i autora niniejszej rozprawy [141]. W k -tej iteracji, istniejąca już podprzestrzeń $\mathbf{f}_{j_1}^1, \dots, \mathbf{f}_{j_{k-1}}^{k-1}$ jest uzupełniana wektorem $\mathbf{f}_{j_k}^k$ w taki sposób, aby rzut ortogonalny \mathbf{y}_0 na otrzymaną k -wymiarową podprzestrzeń był jak najdłuższy (czyli błąd modelowania jak najmniejszy). Osiąga się w ten sposób właściwość **lokalnej optymalności** algorytmu modelowania: w każdej iteracji istniejąca podprzestrzeń jest uzupełniana najlepszym wektorem. Nie należy tego mylić z globalną optymalnością - dodając w każdym kroku jeden wektor nie otrzymuje się najlepszej z możliwych K -wymiarowych podprzestrzeni rozpiętych na wektorach $\mathbf{f}_{j_k}^k, k = 1, \dots, K$.

Realizacja k -tego kroku lokalnie optymalnego algorytmu modelowania mogłaby polegać na obliczeniu, dla kolejnych wektorów - kandydatów \mathbf{f}_j^k , wszystkich wzmocnień g^1, \dots, g^k - byłoby to jednak rozwiązanie kosztowne ze względu na dużą złożoność obliczeniową. Wymienieni autorzy zaproponowali szereg bardziej efektywnych realizacji tego algorytmu. Pierwsza polega na progresywnym tworzeniu ortogonalnej bazy w omawianej podprzestrzeni. Po wybraniu pierwszego wektora $\mathbf{f}_{j_1}^1$ (zapewniającego minimalny błąd modelowania, tzn. minimalny kąt ostry z \mathbf{y}_0) umieszczamy go w bazie ortogonalnej, jako $\bar{\mathbf{f}}_{j_1}^1$. Następnie ortogonalizuje się względem niego wszystkie wektory $\mathbf{f}_j^2, \dots, \mathbf{f}_j^K$ należące do słowników $\mathbf{F}^2, \dots, \mathbf{F}^K$, które mogą jeszcze wejść do tej podprzestrzeni. Operację ortogonalizacji można opisać wzorem:

$$\begin{aligned} \bar{\mathbf{f}}_j^m &= \mathbf{f}_j^m - \frac{(\bar{\mathbf{f}}_{j_1}^1)^t \mathbf{f}_j^m}{\|\bar{\mathbf{f}}_{j_1}^1\|^2} \bar{\mathbf{f}}_{j_1}^1 \\ &= \mathbf{f}_j^m - \frac{(\bar{\mathbf{f}}_{j_1}^1) (\bar{\mathbf{f}}_{j_1}^1)^t}{\|\bar{\mathbf{f}}_{j_1}^1\|^2} \mathbf{f}_j^m \\ &= \mathbf{P}(\bar{\mathbf{f}}_{j_1}^1) \mathbf{f}_j^m \end{aligned} \quad (3.65)$$

gdzie $\mathbf{P}(\mathbf{f}) = \mathbf{I} - \frac{\mathbf{f} \mathbf{f}^t}{\|\mathbf{f}\|^2}$ jest operatorem ortogonalizacji względem \mathbf{f} .

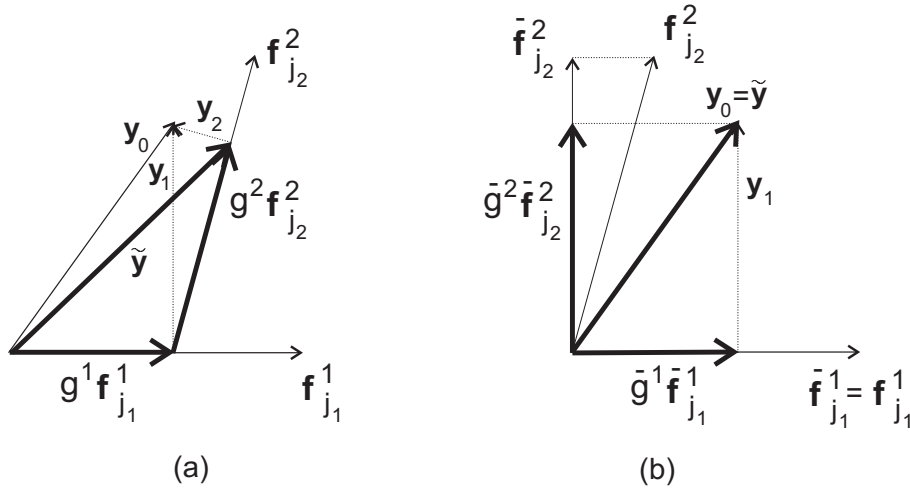
Zortogonalizowany słownik można wykorzystać w algorytmie rzutowania błędu, co prowadzi do wyboru lokalnie optymalnego wektora $\bar{\mathbf{f}}_{j_2}^2$. Następnie dokonuje się ortogonalizacji względem $\bar{\mathbf{f}}_{j_2}^2$, itd. Otrzymany po K krokach model $\tilde{\mathbf{y}} = \sum_{k=1}^K \bar{g}^k \bar{\mathbf{f}}_{j_k}^k$ jest już ortogonalnym rzutem \mathbf{y}_0 na podprzestrzeń, uzupełnianą w kolejnych krokach lokalnie optymalnymi wektorami.

Załóżmy teraz, że wszystkie wektory pochodzą z tego samego słownika \mathbf{F} (pobudzenie mieszane lub wieloimpulsowe). W k -tym kroku algorytmu modelowania słownik ten podlega ortogonalizacji względem wybranego wektora $\bar{\mathbf{f}}_{j_k}^k$. Oznacza to, że słownik $\bar{\mathbf{F}}^k$ zostaje przekształcony w $\bar{\mathbf{F}}^{k+1}$ (zgodnie z tą notacją pierwotny słownik \mathbf{F} oznaczamy przez $\bar{\mathbf{F}}^1$, czyli $\bar{\mathbf{F}}^1 = \mathbf{F}$). Ortogonalizacja wektora $\bar{\mathbf{f}}_j^k \in \bar{\mathbf{F}}^k$ polega na odjęciu rzutu ortogonalnego na $\bar{\mathbf{f}}_{j_k}^k$:

$$\bar{\mathbf{f}}_j^{k+1} = \bar{\mathbf{f}}_j^k - \frac{(\bar{\mathbf{f}}_{j_k}^k)^t \bar{\mathbf{f}}_j^k}{\|\bar{\mathbf{f}}_{j_k}^k\|^2} \bar{\mathbf{f}}_{j_k}^k = \bar{\mathbf{f}}_j^k - \bar{r}(k, j) \bar{\mathbf{f}}_{j_k}^k \quad (3.66)$$

gdzie $\bar{r}(k, j) = \frac{(\bar{\mathbf{f}}_{j_k}^k)^t \bar{\mathbf{f}}_j^k}{\|\bar{\mathbf{f}}_{j_k}^k\|^2}$. Powyższe równanie można przepisać w postaci

$$\bar{\mathbf{f}}_j^k = \bar{r}(k, j) \bar{\mathbf{f}}_{j_k}^k + \bar{\mathbf{f}}_j^{k+1} \quad (3.67)$$



Rysunek 3.9: Modelowanie 2-wymiarowego wektora y_0 : (a)- standardowy algorytm iteracyjny, (b)- algorytm lokalnie optymalny z ortogonalizacją słownika

gdzie pierwszy składnik jest rzutem ortogonalnym $\bar{\mathbf{f}}_j^k$ na wybrany wektor $\bar{\mathbf{f}}_{j_k}^k$, a drugi jest prostopadły do wybranego wektora. Stosując powyższe równanie rekurencyjnie, dochodzi się do wektora $\bar{\mathbf{f}}_j^1 = \mathbf{f}_j$:

$$\mathbf{f}_j = \bar{\mathbf{f}}_j^1 = \bar{r}(1, j) \bar{\mathbf{f}}_{j_1}^1 + \cdots + \bar{r}(k, j) \bar{\mathbf{f}}_{j_k}^k + \bar{\mathbf{f}}_j^{k+1} \quad (3.68)$$

gdzie $j \neq j_1 \neq \cdots \neq j_k$.

Zakładając, że wektor wybrany w k -tej iteracji znajduje się w k -tej kolumnie macierzy \mathbf{F} (tzn. w każdej iteracji dokonuje się permutacji kolumn), można słownik pierwotny $\bar{\mathbf{F}}^1 = \mathbf{F}$ wyrazić następującym wzorem:

$$\bar{\mathbf{F}}^1 = \bar{\mathbf{F}}^{k+1} \bar{\mathbf{R}}^k \quad (3.69)$$

gdzie

$$\bar{\mathbf{R}}^k = \begin{bmatrix} 1 & \bar{r}(1, 2) & \bar{r}(1, 3) & \cdots & \bar{r}(1, k) & \bar{r}(1, k+1) & \cdots & \bar{r}(1, L) \\ 0 & 1 & \bar{r}(2, 3) & \cdots & \bar{r}(2, k) & \bar{r}(2, k+1) & \cdots & \bar{r}(2, L) \\ \vdots & \vdots & \vdots & & \vdots & \vdots & & \vdots \\ 0 & 0 & 0 & \cdots & 1 & \bar{r}(k, k+1) & \cdots & \bar{r}(k, L) \\ 0 & 0 & 0 & \cdots & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & & \vdots & \vdots & & \vdots \\ 0 & 0 & 0 & \cdots & 0 & 0 & \cdots & 1 \end{bmatrix}$$

Zapisane wyżej równanie macierzowe odpowiada ortogonalizacji Grama - Schmidta k pierwszych kolumn macierzy $\bar{\mathbf{F}}^1 = \mathbf{F}$ z rozszerzeniem do $L - k$ pozostałych kolumn. Według Goluba i Van Loana nazywa się to zmodyfikowaną dekompozycją Grama - Schmidta [80], gdyż w standardowej procedurze Grama - Schmidta transformacji podlegają tylko kolejne kolumny.

Można teraz zmodyfikować standardowy algorytm iteracyjny wyboru wektorów ze słownika \mathbf{F} i odpowiadających im wzmocnień w taki sposób, aby nadać mu cechę lokalnej optymalności. Wystarczy w tym celu, w k -tym etapie obliczeń, wykorzystać zortogonalizowany słownik $\bar{\mathbf{F}}^k$, zamiast oryginalnego słownika \mathbf{F} . Otrzymamy w ten sposób **zmodyfikowany algorytm Grama - Schmidta**. Zauważmy, że nie trzeba teraz rzutować błędu modelowania na kolejne ortogonalne wektory - identyczny efekt daje rzutowanie wektora docelowego \mathbf{y}_0 (rys.3.9b). Algorytm można zapisać w następujących krokach:

Mając dany słownik $\bar{\mathbf{F}}^1 = \mathbf{F}$ i wektor docelowy \mathbf{y}_0 oblicza się dla każdego wektora $\bar{\mathbf{f}}_j^1 \in \bar{\mathbf{F}}^1$ iloczynny skalarny $(\bar{\mathbf{f}}_j^1)^t \mathbf{y}_0$ i $(\bar{\mathbf{f}}_j^1)^t \bar{\mathbf{f}}_j^1$.

Dla $k = 1, \dots, K$:

- wybiera się wektor $\bar{\mathbf{f}}_{j_k}^k$, zapewniający maksimum wartości $\frac{[(\bar{\mathbf{f}}_j^k)^t \mathbf{y}_0]^2}{(\bar{\mathbf{f}}_j^k)^t (\bar{\mathbf{f}}_j^k)}$ (oznacza to wybór wektora tworzącego najmniejszy kąt ostry z \mathbf{y}_0)
- dla wybranego wektora oblicza się niekwantowane wzmocnienie: $\bar{g}^k = \frac{(\bar{\mathbf{f}}_{j_k}^k)^t \mathbf{y}_0}{\|\bar{\mathbf{f}}_{j_k}^k\|^2}$
- dla każdego wektora $\bar{\mathbf{f}}_j^k$ ze słownika $\bar{\mathbf{F}}^k$

– oblicza się $\bar{r}(k, j) = \frac{(\bar{\mathbf{f}}_j^k)^t \bar{\mathbf{f}}_j^k}{\|\bar{\mathbf{f}}_{j_k}^k\|^2}$

– przeprowadza się ortogonalizację słownika $\bar{\mathbf{F}}^k$ względem $\bar{\mathbf{f}}_{j_k}^k$, rezultatem jest słownik $\bar{\mathbf{F}}^{k+1}$ (wzór (3.66)):

$$\bar{\mathbf{f}}_j^{k+1} = \bar{\mathbf{f}}_j^k - \bar{r}(k, j) \bar{\mathbf{f}}_{j_k}^k$$

– aktualizuje się iloczyny skalarne:

$$(\bar{\mathbf{f}}_j^{k+1})^t \mathbf{y}_0 = (\bar{\mathbf{f}}_j^k)^t \mathbf{y}_0 - \bar{r}(k, j) (\bar{\mathbf{f}}_{j_k}^k)^t \mathbf{y}_0$$

oraz

$$(\bar{\mathbf{f}}_j^{k+1})^t (\bar{\mathbf{f}}_j^{k+1}) = (\bar{\mathbf{f}}_j^k)^t (\bar{\mathbf{f}}_j^k) - 2\bar{r}(k, j) (\bar{\mathbf{f}}_j^k)^t (\bar{\mathbf{f}}_{j_k}^k) + \bar{r}^2(k, j) (\bar{\mathbf{f}}_{j_k}^k)^t (\bar{\mathbf{f}}_{j_k}^k)$$

co upraszcza się do postaci

$$(\bar{\mathbf{f}}_j^{k+1})^t (\bar{\mathbf{f}}_j^{k+1}) = (\bar{\mathbf{f}}_j^k)^t (\bar{\mathbf{f}}_j^k) - \bar{r}^2(k, j) (\bar{\mathbf{f}}_{j_k}^k)^t (\bar{\mathbf{f}}_{j_k}^k)$$

W ostatnim etapie, dla $k = K$, końcowy punkt (ortogonalizacja słownika i aktualizacja iloczynów skalarnych) można pominąć.

Obliczone wzmocnienia $\bar{g}^1, \dots, \bar{g}^K$ odpowiadają zortogonalizowanym wektorom, które nie są znane a priori po stronie odbiorczej. Jeśli chcemy uniknąć przeprowadzania ortogonalizacji słownika po stronie odbiorczej, to należy wyznaczyć wzmocnienia odpowiadające wektorom z pierwotnego słownika \mathbf{F} . Zakładając, że wybrane wektory słownika znajdują się w kolejnych K kolumnach macierzy \mathbf{F} , można zapisać:

$$\tilde{\mathbf{y}} = \mathbf{F} [g^1 \dots g^K 0 \dots 0]^t = \bar{\mathbf{F}}^K [\bar{g}^1 \dots \bar{g}^K 0 \dots 0]^t \quad (3.70)$$

Ponieważ (wzór (3.69)) $\mathbf{F} = \bar{\mathbf{F}}^1 = \bar{\mathbf{F}}^K \bar{\mathbf{R}}^{K-1}$, zatem otrzymuje się układ równań:

$$\begin{bmatrix} \bar{g}^1 \\ \bar{g}^2 \\ \vdots \\ \bar{g}^{K-1} \\ \bar{g}^K \end{bmatrix} = \begin{bmatrix} 1 & \bar{r}(1,2) & \cdots & \bar{r}(1,K) \\ 0 & 1 & \cdots & \bar{r}(2,K) \\ \vdots & \vdots & & \vdots \\ 0 & 0 & \cdots & \bar{r}(K-1,K) \\ 0 & 0 & \cdots & 1 \end{bmatrix} \begin{bmatrix} g^1 \\ g^2 \\ \vdots \\ g^{K-1} \\ g^K \end{bmatrix} \quad (3.71)$$

Rozwiązanie układu równań liniowych (3.71) jest trywialne, gdyż macierz układu jest macierzą trójkątną górną, zawierającą na przekątnej elementy jednostkowe:

$$g^k = \bar{g}^k$$

dla $k = K - 1, \dots, 1$

$$g^k = \bar{g}^k - \sum_{n=k+1}^K \bar{r}(k,n) g^n$$

Zmodyfikowany algorytm Grama - Schmidta ma tę wadę, że nie udało się w nim uniknąć bezpośredniego działania na wektorach słownika (w procesie ortogonalizacji), co zwiększa złożoność obliczeniową. Bezpośredniej ortogonalizacji słownika można uniknąć, stosując następującą modyfikację algorytmu. W k -tym kroku mamy do czynienia ze słownikiem $\bar{\mathbf{F}}^k$, którego wektory były już ortogonalizowane względem $\bar{\mathbf{f}}_{j_1}^1, \dots, \bar{\mathbf{f}}_{j_{k-1}}^{k-1}$, stąd

$$\bar{\mathbf{f}}_j^k = \mathbf{f}_j - \sum_{n=1}^{k-1} \bar{r}(n,j) \bar{\mathbf{f}}_{j_n}^n \quad (3.72)$$

Podstawiając (3.72) do wzoru na $\bar{r}(k,j)$, otrzymuje się

$$\bar{r}(k,j) = \frac{(\bar{\mathbf{f}}_{j_k}^k)^t \bar{\mathbf{f}}_j^k}{(\bar{\mathbf{f}}_{j_k}^k)^t (\bar{\mathbf{f}}_{j_k}^k)} = \frac{(\bar{\mathbf{f}}_{j_k}^k)^t \mathbf{f}_j}{(\bar{\mathbf{f}}_{j_k}^k)^t (\bar{\mathbf{f}}_{j_k}^k)} \quad (3.73)$$

gdyż wektor $\bar{\mathbf{f}}_{j_k}^k$ jest prostopadły do $\bar{\mathbf{f}}_{j_1}^1, \dots, \bar{\mathbf{f}}_{j_{k-1}}^{k-1}$. Podstawiając (3.72), dla $j = j_k$, do (3.73) otrzymuje się

$$\begin{aligned} \bar{r}(k,j) &= \frac{[\mathbf{f}_{j_k} - \sum_{n=1}^{k-1} \bar{r}(n,j_k) \bar{\mathbf{f}}_{j_n}^n]^t \mathbf{f}_j}{(\bar{\mathbf{f}}_{j_k}^k)^t (\bar{\mathbf{f}}_{j_k}^k)} \\ &= \frac{\mathbf{f}_{j_k}^t \mathbf{f}_j - \sum_{n=1}^{k-1} \bar{r}(n,j_k) (\bar{\mathbf{f}}_{j_n}^n)^t \mathbf{f}_j}{(\bar{\mathbf{f}}_{j_k}^k)^t (\bar{\mathbf{f}}_{j_k}^k)} \end{aligned} \quad (3.74)$$

Podstawiając jeszcze raz (3.73) otrzymuje się:

$$\bar{r}(k,j) = \frac{\mathbf{f}_{j_k}^t \mathbf{f}_j - \sum_{n=1}^{k-1} \bar{r}(n,j_k) \bar{r}(n,j) (\bar{\mathbf{f}}_{j_n}^n)^t (\bar{\mathbf{f}}_{j_n}^n)}{(\bar{\mathbf{f}}_{j_k}^k)^t (\bar{\mathbf{f}}_{j_k}^k)} \quad (3.75)$$

W ten sposób można wyeliminować bezpośrednie działanie na wektorach, a zwłaszcza uniknąć bezpośredniej ortogonalizacji słownika. Jediną operacją na wektorach jest obliczenie

iloczynów skalarnych $\mathbf{f}_{jk}^t \mathbf{f}_j$. W kolejnych krokach aktualizuje się rekursywnie iloczyny skalarne $(\bar{\mathbf{f}}_j^k)^t \mathbf{y}_0$ i $(\bar{\mathbf{f}}_j^k)^t (\bar{\mathbf{f}}_j^k)$. Tę wersję algorytmu lokalnie optymalnego będziemy nazywać **rekursywnym zmodyfikowanym algorytmem Grama - Schmidta**:

Mając dany słownik $\bar{\mathbf{F}}^1 = \mathbf{F}$ i wektor docelowy \mathbf{y}_0 oblicza się dla każdego wektora $\bar{\mathbf{f}}_j^1 \in \bar{\mathbf{F}}^1$ iloczyny skalarne $(\bar{\mathbf{f}}_j^1)^t \mathbf{y}_0$ i $(\bar{\mathbf{f}}_j^1)^t \bar{\mathbf{f}}_j^1$.

Dla $k = 1, \dots, K$:

- wybiera się lokalnie optymalny wektor \mathbf{f}_{jk} , zapewniający maksimum wartości

$$\frac{[(\bar{\mathbf{f}}_j^k)^t \mathbf{y}_0]^2}{(\bar{\mathbf{f}}_j^k)^t (\bar{\mathbf{f}}_j^k)}$$

- oblicza się niekwantowane wzmocnienie, odpowiadające zortogonalizowanemu wektorowi $\bar{\mathbf{f}}_{jk}^k$:

$$\bar{g}^k = \frac{(\bar{\mathbf{f}}_{jk}^k)^t \mathbf{y}_0}{(\bar{\mathbf{f}}_{jk}^k)^t (\bar{\mathbf{f}}_{jk}^k)}$$

- dla każdego wektora \mathbf{f}_j ze słownika \mathbf{F} (można pominąć w ostatnim etapie, gdy $k = K$)
 - oblicza się $\bar{r}(k, j)$ ze wzoru (3.75) - jedyną operacją na wektorach jest obliczenie $\mathbf{f}_{jk}^t \mathbf{f}_j$, kwadraty norm $(\bar{\mathbf{f}}_{jn}^n)^t (\bar{\mathbf{f}}_{jn}^n)$ są aktualizowane rekursywnie;
 - aktualizuje się iloczyny skalarne, tak jak w zmodyfikowanym algorytmie Grama - Schmidta:

$$(\bar{\mathbf{f}}_j^{k+1})^t \mathbf{y}_0 = (\bar{\mathbf{f}}_j^k)^t \mathbf{y}_0 - \bar{r}(k, j) (\bar{\mathbf{f}}_{jk}^k)^t \mathbf{y}_0$$

oraz

$$(\bar{\mathbf{f}}_j^{k+1})^t (\bar{\mathbf{f}}_j^{k+1}) = (\bar{\mathbf{f}}_j^k)^t (\bar{\mathbf{f}}_j^k) - \bar{r}^2(k, j) (\bar{\mathbf{f}}_{jk}^k)^t (\bar{\mathbf{f}}_{jk}^k)$$

Na koniec oblicza się wzmocnienia odpowiadające wektorom oryginalnego (niezortogonalizowanego) słownika, rozwiązując układ równań (3.71).

W koderach CELP słownik \mathbf{F} zawiera odfiltrowane wektory $\mathbf{f}_j = \mathbf{H}'\mathbf{c}_j$. Ich obliczenie nie jest konieczne, gdyż występujące w algorytmie iloczyny skalarne mogą być wyrażone w funkcji wektorów \mathbf{c}_j :

$$\mathbf{f}_j^t \mathbf{f}_j = \mathbf{c}_j^t (\mathbf{H}')^t \mathbf{H}' \mathbf{c}_j$$

$$\mathbf{f}_j^t \mathbf{y}_0 = \mathbf{c}_j^t [(\mathbf{H}')^t \mathbf{y}_0]$$

Ma to duże znaczenie w przypadku, gdy wektory \mathbf{c}_j są "rzadkie", tzn. zawierają większość składowych o wartościach zerowych. Właściwość tę można wykorzystać w algorytmie modelowania wektora docelowego \mathbf{y}_0 . Zgodnie ze wzorem (3.65) ortogonalizacja wektorów słownika w k -tym kroku algorytmu może być przedstawiona w następujący sposób:

$$\bar{\mathbf{f}}_j^{k+1} = \mathbf{P}(\bar{\mathbf{f}}_{jk}^k) \bar{\mathbf{f}}_j^k = \mathbf{P}(\bar{\mathbf{f}}_{jk}^k) \cdots \mathbf{P}(\bar{\mathbf{f}}_{j1}^1) \mathbf{f}_j \quad (3.76)$$

Powyższą zależność można wykorzystać do aktualizacji iloczynów skalarnych

$$\begin{aligned} (\bar{\mathbf{f}}_j^{k+1})^t \bar{\mathbf{f}}_j^{k+1} &= (\mathbf{H}' \mathbf{c}_j)^t \mathbf{P}^t(\bar{\mathbf{f}}_{j_1}^1) \cdots \mathbf{P}^t(\bar{\mathbf{f}}_{j_k}^k) \mathbf{P}(\bar{\mathbf{f}}_{j_k}^k) \cdots \mathbf{P}(\bar{\mathbf{f}}_{j_1}^1) (\mathbf{H}' \mathbf{c}_j) \\ &= \mathbf{c}_j^t \left[\mathbf{H}'^t \mathbf{P}(\bar{\mathbf{f}}_{j_1}^1) \cdots \mathbf{P}(\bar{\mathbf{f}}_{j_k}^k) \mathbf{H}' \right] \mathbf{c}_j \end{aligned} \quad (3.77)$$

gdyż operatory ortogonalizacji są opisane macierzami symetrycznymi oraz komutują ze sobą. Oczywiście jest, że nie jest konieczna dwukrotna ortogonalizacja względem tego samego wektora - stąd uproszczenie wzoru (3.77). Podobnie aktualizuje się iloczyn skalarny $(\bar{\mathbf{f}}_j^{k+1})^t \mathbf{y}_0$:

$$\begin{aligned} (\bar{\mathbf{f}}_j^{k+1})^t \mathbf{y}_0 &= \left[\mathbf{P}(\bar{\mathbf{f}}_{j_k}^k) \cdots \mathbf{P}(\bar{\mathbf{f}}_{j_1}^1) (\mathbf{H}' \mathbf{c}_j) \right]^t \mathbf{y}_0 \\ &= \mathbf{c}_j^t \left[\mathbf{H}'^t \mathbf{P}(\bar{\mathbf{f}}_{j_1}^1) \cdots \mathbf{P}(\bar{\mathbf{f}}_{j_k}^k) \right] \mathbf{y}_0 \end{aligned} \quad (3.78)$$

W przypadku, gdy wektory \mathbf{c}_j są "rzadkie", korzystanie ze wzorów (3.77) i (3.78) może mieć pewne zalety. Algorytm oparty na rekursywnym obliczaniu (3.77) i (3.78) nazwiemy **algorytmem filtracji odwrotnej**, gdyż przetwarzane są nieodfiltrowane odpowiedniki wektorów słownika \mathbf{F} .

Podobne podejście zaproponował Taniguchi i inni [193], stosując odwrotną filtrację w celu otrzymania nieodfiltrowanych odpowiedników zortogonalizowanych wektorów:

$$\begin{aligned} \bar{\mathbf{f}}_j^{k+1} &= \mathbf{P}(\bar{\mathbf{f}}_{j_k}^k) \bar{\mathbf{f}}_j^k \\ \mathbf{H}' \bar{\mathbf{c}}_j^{k+1} &= \mathbf{P}(\bar{\mathbf{f}}_{j_k}^k) \mathbf{H}' \bar{\mathbf{c}}_j^k \\ \bar{\mathbf{c}}_j^{k+1} &= (\mathbf{H}')^{-1} \mathbf{P}(\bar{\mathbf{f}}_{j_k}^k) \mathbf{H}' \bar{\mathbf{c}}_j^k \end{aligned}$$

Podejście to charakteryzuje się większą złożonością obliczeniową niż proponowany algorytm filtracji odwrotnej.

3.4.4 Modelowanie wielostopniowe z pojedynczym wzmocnieniem

Zgodnie ze wzorem (3.58) model wektora docelowego \mathbf{y}_0 otrzymuje się w następującej postaci:

$$\tilde{\mathbf{y}} = g \sum_{k=1}^K s^k \mathbf{f}_{j_k}^k$$

Modelowanie z pojedynczym współczynnikiem wzmocnienia odpowiada konkatencji K krótkich wektorów, sumowaniu wektorów (kodery VSELP [78, 79]), a także pobudzeniu algebraicznemu, w szczególności ternarnemu. W przypadku koderów CELP wektory $\mathbf{f}_{j_k}^k$ stanowią odfiltrowane wersje wektorów ze słowników pobudzeń: $\mathbf{f}_{j_k}^k = \mathbf{H}' \mathbf{c}_{j_k}^k$.

Wybrane wektory $\mathbf{f}_{j_k}^k$, $k = 1, \dots, K$ tworzą jednowymiarową podprzestrzeń opisaną wektorem

$$\mathbf{f} = \sum_{k=1}^K s^k \mathbf{f}_{j_k}^k$$

Wybiera się ją w ten sposób, aby była ona najbliższa w stosunku do wektora docelowego \mathbf{y}_0 , co oznacza minimalizację kąta między \mathbf{y}_0 i \mathbf{f} , minimalizację odległości \mathbf{y}_0 od rzutu ortogonalnego $g \mathbf{f}$ lub maksymalizację długości rzutu wektora \mathbf{y}_0 na \mathbf{f} . Sprowadza się to do maksymalizacji kryterium (punkt 3.3.1):

$$\frac{[\mathbf{f}^t \mathbf{y}_0]^2}{\mathbf{f}^t \mathbf{f}} = \frac{[\sum_{k=1}^K s^k (\mathbf{f}_{j_k}^k)^t \mathbf{y}_0]^2}{\sum_{k=1}^K \sum_{m=1}^K s^k s^m (\mathbf{f}_{j_k}^k)^t \mathbf{f}_{j_m}^m} \quad (3.79)$$

Dla koderów CELP, podstawiając $\mathbf{f}_{j_k}^k = \mathbf{H}' \mathbf{c}_{j_k}^k$, otrzymuje się

$$\frac{[\sum_{k=1}^K s^k (\mathbf{c}_{j_k}^k)^t \Psi]^2}{\sum_{k=1}^K \sum_{m=1}^K s^k s^m (\mathbf{c}_{j_k}^k)^t \Phi \mathbf{c}_{j_m}^m} \quad (3.80)$$

gdzie $\Psi = (\mathbf{H}')^t \mathbf{y}_0$, $\Phi = (\mathbf{H}')^t \mathbf{H}'$.

Dla koderów CELP z pobudzeniem ternarnym wektor $s^k \mathbf{c}_{j_k}^k$ ma jedną niezerową (j_k -tą) składową o wartości $s^k = \pm 1$. Wówczas suma K takich wektorów (nazwijmy ją \mathbf{c}^K) ma K składowych niezerowych o wartościach s^k na pozycjach j_k ($k = 1 \dots K$). W tej sytuacji można zapisać:

$$\sum_{k=1}^K s^k (\mathbf{c}_{j_k}^k)^t \Psi = \sum_{k=1}^K s^k \Psi(j_k)$$

oraz

$$\sum_{k=1}^K \sum_{m=1}^K s^k s^m (\mathbf{c}_{j_k}^k)^t \Phi \mathbf{c}_{j_m}^m = \sum_{k=1}^K \sum_{m=1}^K s^k s^m \Phi(j_k, j_m)$$

co po podstawieniu do (3.80) daje

$$\frac{[\sum_{k=1}^K s^k \Psi(j_k)]^2}{\sum_{k=1}^K \sum_{m=1}^K s^k s^m \Phi(j_k, j_m)} \quad (3.81)$$

Jednowymiarową podprzestrzeń, opisaną wektorem \mathbf{f} , można utworzyć na wiele sposobów, przetestowanie ich wszystkich jest najczęściej niemożliwe ze względu na złożoność obliczeniową. Np. N -wymiarowy wektor pobudzenia ternarnego zawierającego K impulsów o amplitudach ± 1 można skonstruować na

$$L = 2^K \frac{N!}{K! (N-K)!}$$

sposobów. Dla $N = 40$ otrzymuje się: dla $K = 4, L = 1.5 \cdot 10^6$, dla $K = 5, L = 2.1 \cdot 10^7$, dla $K = 6, L = 2.5 \cdot 10^8$, dla $K = 7, L = 2.4 \cdot 10^9$. Oczywiście należy poszukiwać algorytmów suboptymalnych, albo nałożyć ograniczenia na położenia impulsów, aby zredukować liczbę kombinacji. Np. w koderze GSM-EFR [69] każdy z 10 impulsów ma tylko 8 możliwych położenia.

Najszybsze algorytmy konstruowania pobudzenia ternarnego otrzymamy, ignorując wartość mianownika we wzorze (3.81) i maksymalizując licznik [178, 194]. W tym wypadku

testowanie położenia impulsów jest zbędne: impulsy zostają rozmieszczone w miejscach j_k odpowiadających największym (co do wartości bezwzględnej) składowym wektora Ψ i mają amplitudy

$$s^k = \text{sgn}(\Psi(j_k))$$

Wyniki są zachęcające, ale jedynie dla pobudzenia ternarnego o strukturze regularnej (impulsy równoodległe). W [178] zanotowano spadek SNR jedynie o 0.6 dB w stosunku do algorytmu opartego na przetestowaniu wszystkich kombinacji. W przypadku pobudzenia o strukturze nieregularnej spadek jakości przekracza 2 dB. Ten algorytm może być zastosowany jedynie do konstruowania pobudzenia ternarnego, w innych przypadkach (np. konkatencja krótkich wektorów) nie ma on sensu.

Inne algorytmy suboptymalne polegają na przeszukiwaniu podzbioru wektorów pobudzenia otaczającego wektor maksymalizujący licznik wyrażenia (3.81) [180]. W [115] impulsy wstawia się kolejno, przeszukując podzbiór wektorów otaczający wektor maksymalizujący kryterium (3.81) w poprzedniej iteracji. Złożoność obliczeniowa takiego algorytmu jest nawet o 3 rzędy wielkości mniejsza od algorytmu optymalnego (przeszukanie wszystkich kombinacji), lecz jest zmienna, co może być niewygodne w zastosowaniach telekomunikacyjnych (przetwarzanie w czasie rzeczywistym).

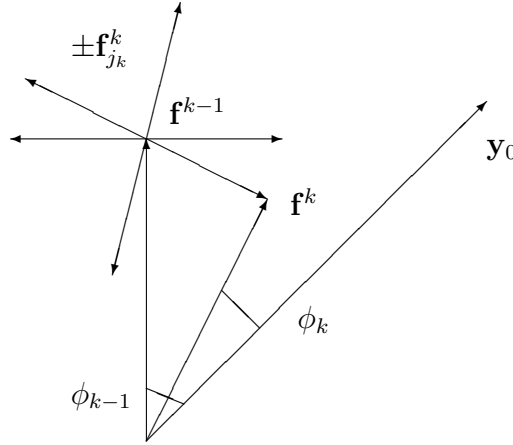
W [56, 145] autor niniejszej rozprawy wraz z N. Moreau zaproponowali inny algorytm suboptymalny, wymagający przetestowania mniej niż $2NK$ wektorów (dla pobudzenia ternarnego). Algorytm ten może być zastosowany we wszelkich koderach wielostopniowych z pojedynczym wzmocnieniem (przetestowano go na pobudzeniu ternarnym [56] i na pobudzeniu zbudowanym metodą konkatencji [145]). Polega on na **sukcesywnej minimalizacji kąta** między wektorem docelowym \mathbf{y}_0 i tworzoną jednowymiarową podprzestrzenią \mathbf{f} .

Algorytm ten ma cechy lokalnej optymalności: w każdej z K iteracji dołącza się najlepszy wektor, pod warunkiem jednak, że nie wolno zmieniać decyzji podjętych w poprzednich iteracjach. I tak, w k -tej iteracji dodaje się do istniejącego wektora, definiującego jednowymiarową podprzestrzeń $\mathbf{f}^{k-1} = \sum_{i=1}^{k-1} s^i \mathbf{f}_{j_i}^i$, wektor $s^k \mathbf{f}_{j_k}^k = \pm \mathbf{f}_{j_k}^k$, w taki sposób, aby zminimalizować kąt między otrzymanym wektorem $\mathbf{f}^k = \sum_{i=1}^k s^i \mathbf{f}_{j_i}^i$ i wektorem docelowym \mathbf{y}_0 (rys. 3.10).

W każdej iteracji wybiera się wektor $\mathbf{f}_{j_k}^k$ i znak s^k z tym, że w pierwszej iteracji istnieje dowolność wyboru znaku ($\pm \mathbf{f}_{j_k}^k$ daje tę samą wartość kryterium (3.79), gdyż znak można przenieść do wzmocnienia g). I tak, można narzucić $s^1 = 1$ (jeśli wzmocnienie g może być ujemne), lub wybrać s^1 tak, aby wymusić kąt ostry między $s^1 \mathbf{f}_{j_1}^1$ i \mathbf{y}_0 (wówczas $g > 0$).

Dla pobudzenia ternarnego (kryterium (3.81)) w k -tej iteracji testuje się $N - k + 1$ pozycji impulsu (j_k) i 2 znaki (amplitudy) s^k . W sumie liczba obliczeń kryterium nie przekracza $2KN$.

Algorytm sukcesywnej minimalizacji kąta nie gwarantuje monotonicznego zmniejszania się kąta, ale nie jest wskazane jego zatrzymanie w przypadku, gdy kąt w danej iteracji wzrośnie. Taki wzrost ma najczęściej charakter lokalny, zmniejszenie się kąta następuje w kolejnych iteracjach (porównano oba warianty - różnice w SNR nie przekraczały 0.3 dB, głównie na korzyść wariantu "bez zatrzymania"). Poza tym wybór mniejszej liczby wektorów prowadzi do zwiększenia przepływności binarnej (trzeba przesyłać informację o tym, ile wektorów wybrano).



Rysunek 3.10: Algorytm sukcesywnej minimalizacji kąta

W realizacjach praktycznych etap modelowania z pojedynczym wzmocnieniem (np. pobudzenie ternarne) poprzedzony jest predykcją długookresową (LTP). Oznacza to wybór, w pierwszym etapie obliczeń, wektora $\mathbf{f}_{j_1}^1$ ze słownika adaptacyjnego wraz z odpowiednim wzmocnieniem g_1 . W pozostałych $K - 1$ etapach wykorzystuje się wspólne wzmocnienie g_2 :

$$\tilde{\mathbf{y}} = g_1 \mathbf{f}_{j_1}^1 + g_2 \sum_{k=2}^K s^k \mathbf{f}_{j_k}^k \quad (3.82)$$

Oznacza to, że poszukuje się 2-wymiarowej podprzestrzeni opisanej wektorami $\mathbf{f}_{j_1}^1$ i $\mathbf{f} = \sum_{k=2}^K s^k \mathbf{f}_{j_k}^k$ tak, aby błąd rzutowania ortogonalnego wektora docelowego \mathbf{y}_0 na tę podprzestrzeń był jak najmniejszy. Oznacza to minimalizację kąta między \mathbf{y}_0 i poszukiwaną podprzestrzenią (płaszczyzną), lub maksymalizację długości rzutu wektora docelowego na tę płaszczyznę.

Ze względu na złożoność obliczeniową, w pierwszym etapie wybiera się odpowiedni wektor $\mathbf{f}_{j_1}^1$ ze słownika adaptacyjnego (minimalizując kąt z wektorem docelowym). W kolejnych etapach tworzy się dwuwymiarową podprzestrzeń, opisaną wektorem $\mathbf{f}_{j_1}^1$ i budowanym wektorem $\mathbf{f}^k = \sum_{i=2}^k s^i \mathbf{f}_{j_i}^i$. Badano dwa warianty tego algorytmu. Wariant standardowy polegał na wyznaczeniu, w pierwszym etapie obliczeń, wzmocnienia g_1 i określeniu błędu modelowania

$$\mathbf{y}_1 = \mathbf{y}_0 - g_1 \mathbf{f}_{j_1}^1$$

Następnie wykonywano algorytm sukcesywnej minimalizacji kąta między \mathbf{y}_1 i tworzoną jednowymiarową podprzestrzenią opisaną wektorem $\mathbf{f}^k = \sum_{i=2}^k s^i \mathbf{f}_{j_i}^i$, $k = 2 \dots K$. Ten wariant nie ma cechy lokalnej optymalności (budując jednowymiarową podprzestrzeń nie uwzględnia się wektora ze słownika adaptacyjnego, a otrzymany model nie jest nawet rzutem ortogonalnym wektora \mathbf{y}_0 na płaszczyznę).

Drugi wariant ma cechę lokalnej optymalności: po wybraniu wektora ze słownika adaptacyjnego buduje się wektor $\mathbf{f}^k = \sum_{i=2}^k s^i \mathbf{f}_{j_i}^i$, tak, aby w każdym etapie obliczeń zachować jak najmniejszy kąt między \mathbf{y}_0 i płaszczyzną opisaną wektorami $\mathbf{f}_{j_1}^1$ i \mathbf{f}^k . Najłatwiej tego dokonać, ortogonalizując wszystkie kandydujące wektory \mathbf{f}_j^k względem wektora ze słownika adaptacyjnego, wybranego w pierwszym etapie. Otrzymuje się w ten sposób wektory zortogonalizowane

$$\bar{\mathbf{f}}_j^k = \mathbf{P}(\mathbf{f}_{j_1}^1) \mathbf{f}_j^k$$

Następnie wykonuje się algorytm sukcesywnej minimalizacji kąta między \mathbf{y}_0 i tworzoną jednowymiarową podprzestrzenią opisaną wektorem $\bar{\mathbf{f}}^k = \sum_{i=2}^k s^i \bar{\mathbf{f}}_{j_i}^i$, $k = 2 \dots K$.

W przypadku kodera CELP o pobudzeniu ternarnym można podstawić $\mathbf{f}_j^k = \mathbf{H}' \mathbf{c}_j^k$, oraz $\bar{\mathbf{f}}_j^k = \mathbf{P}(\mathbf{f}_{j_1}^1) \mathbf{H}' \mathbf{c}_j^k$, lub w skrócie $\bar{\mathbf{f}}_j^k = \mathbf{P} \mathbf{H}' \mathbf{c}_j^k$. W tej sytuacji minimalizacja kąta między \mathbf{y}_0 i tworzoną jednowymiarową podprzestrzenią opisaną wektorem $\bar{\mathbf{f}}^k = \sum_{i=2}^k s^i \bar{\mathbf{f}}_{j_i}^i$, $k = 2 \dots K$ oznacza maksymalizację, w k -tym etapie obliczeń, następującego kryterium:

$$\begin{aligned} \frac{[\mathbf{y}_0^t \bar{\mathbf{f}}^k]^2}{(\bar{\mathbf{f}}^k)^t \bar{\mathbf{f}}^k} &= \frac{[\mathbf{y}_0^t \mathbf{P} \mathbf{H}' \mathbf{c}^k]^2}{(\mathbf{P} \mathbf{H}' \mathbf{c}^k)^t \mathbf{P} \mathbf{H}' \mathbf{c}^k} = \frac{[(\mathbf{y}_0^t \mathbf{P} \mathbf{H}') \mathbf{c}^k]^2}{(\mathbf{c}^k)^t \mathbf{H}'^t \mathbf{P}^t \mathbf{P} \mathbf{H}' \mathbf{c}^k} = \\ &= \frac{[(\mathbf{H}'^t \mathbf{P} \mathbf{y}_0)^t \mathbf{c}^k]^2}{(\mathbf{c}^k)^t \mathbf{H}'^t \mathbf{P} \mathbf{H}' \mathbf{c}^k} \end{aligned} \quad (3.83)$$

gdzie podstawiono $\bar{\mathbf{f}}^k = \mathbf{P} \mathbf{H}' \mathbf{c}^k$, $\mathbf{c}^k = \sum_{i=2}^k s^i \mathbf{c}_{j_i}^i$ oraz $\mathbf{P}^t \mathbf{P} = \mathbf{P} \mathbf{P} = \mathbf{P}$.

Niech Ψ' oznacza wektor $\mathbf{H}'^t \mathbf{P} \mathbf{y}_0$ i Φ' macierz $\mathbf{H}'^t \mathbf{P} \mathbf{H}'$. Kryterium (3.83) można teraz przepisać w postaci podobnej do (3.81):

$$\frac{[\sum_{i=1}^k s^i \Psi'(j_i)]^2}{\sum_{i=1}^k \sum_{m=1}^k s^i s^m \Phi'(j_i, j_m)} \quad (3.84)$$

Obliczanie wektorów Ψ' i macierzy Φ' w (3.84) musi się odbywać częściej niż obliczanie wektorów Ψ i macierzy Φ w (3.81), gdyż te pierwsze zależą od \mathbf{y}_0 , podczas gdy te drugie zależały tylko od współczynników predykcji.

3.4.5 Projektowanie słowników

Zagadnienia projektowania słowników dla SGVQ i CELP z modelowaniem jednostopniowym omówiono w p.3.3.2. Stosowany jest uogólniony algorytm Lloyda jak na rys.3.8. Centroidy dla słownika kształtów i wzmocnień obliczane są, dla SGVQ, ze wzorów (3.41) i (3.42). Jeżeli narzuci się jednostkową normę wektorów kształtu, to wówczas centroidy obliczane są ze wzoru (3.43). Jeśli założy się, że błąd kwantyzacji wzmocnień jest nieskończenie mały, to słownik kształtów otrzymuje się rozwiązując równania (3.48). W przypadku kodera CELP zamiast wzorów (3.41) i (3.42) stosuje się wzory (3.53) i (3.55).

W przypadku modelowania wielostopniowego zagadnienie projektowania słowników przedstawia się następująco: Dla danego zbioru wektorów percepcyjnego sygnału akustycznego,

tworzącego sekwencję treningową $\{\mathbf{y}(i)\}$ poszukuje się słowników kształtów $\{\mathbf{f}_j^1\} \dots \{\mathbf{f}_j^K\}$ (dla kodera CELP $\{\mathbf{c}_j^1\} \dots \{\mathbf{c}_j^K\}$) i słowników wzmocnień $\{g_m^1\} \dots \{g_m^K\}$. Kryterium jest minimalny błąd kodowania całej sekwencji treningowej $D = \sum_i D(i)$.

W większości zastosowań pierwszy etap modelowania jest etapem predykcji długookresowej (p.3.2.3). Oznacza to, że słownik $\{\mathbf{f}_j^1\}$ (dla CELP $\{\mathbf{c}_j^1\}$) jest słownikiem adaptacyjnym, zawierającym opóźnione sekwencje próbek skwantowanego sygnału percepcyjnego $\tilde{\mathbf{y}}$ (dla kodera CELP, skwantowanego sygnału resztkowego $\tilde{\mathbf{r}}$ - patrz rys.3.7). Słownik adaptacyjny $\{\mathbf{f}_j^1\}$ nie podlega wówczas projektowaniu, jednak utrudnia projektowanie pozostałych słowników $\{\mathbf{f}_j^2\} \dots \{\mathbf{f}_j^K\}$ z tego względu, że zawiera on sekwencje sygnału skwantowanego, a więc zależy od wszystkich słowników. Projektowanie powinno odbywać się **w pętli zamkniętej** - zmiana jakiegokolwiek słownika prowadzi bowiem do modyfikacji słownika adaptacyjnego, co z kolei prowadzi do konieczności "przeprojektowania" wszystkich słowników.

Można uprościć algorytm projektowania "rozrywając" pętlę, tzn. wypełniając słownik adaptacyjny opóźnionymi sekwencjami próbek nieskwantowanego sygnału percepcyjnego \mathbf{y} (dla kodera CELP, nieskwantowanego sygnału resztkowego \mathbf{r}). Mówi się wtedy o projektowaniu **w pętli otwartej** [38].

Klasyczny algorytm projektowania słowników K -stopniowego SGVQ polega na K -krotnym zastosowaniu uogólnionego algorytmu Lloyda - rys.3.8. W pierwszym etapie, stosując jako sekwencję treningową $\{\mathbf{y}_0(i)\}$ zbiór wektorów sygnału percepcyjnego $\{\mathbf{y}(i)\}$, projektuje się słownik kształtów $\{\mathbf{f}_j^1\}$ i wzmocnień $\{g_m^1\}$. Jeżeli słownik kształtów ma być adaptacyjny, to wówczas wypełnia się go sekwencjami próbek sygnału percepcyjnego (pętla otwarta), a projektuje się jedynie kwantyzator wzmocnień. Po zaprojektowaniu słowników i przeprowadzeniu kwantyzacji sekwencji treningowej $\{\mathbf{y}_0(i)\}$ oblicza się błąd kwantyzacji w pierwszym etapie modelowania:

$$\mathbf{y}_1(i) = \mathbf{y}_0(i) - g_{m_1(i)}^1 \mathbf{f}_{j_1(i)}^1$$

Zbiór wektorów $\{\mathbf{y}_1(i)\}$ stanowi sekwencję treningową do projektowania słowników kształtów $\{\mathbf{f}_j^2\}$ i wzmocnień $\{g_m^2\}$ wykorzystywanych w drugim etapie modelowania. Następnie oblicza się błędy kwantowania po dwóch etapach:

$$\mathbf{y}_2(i) = \mathbf{y}_1(i) - g_{m_2(i)}^2 \mathbf{f}_{j_2(i)}^2$$

i przystępuje się do projektowania słowników wykorzystywanych w trzecim etapie (o ile $K > 2$).

W każdym etapie projektowania centroidy dla słownika kształtów i wzmocnień obliczane są, dla SGVQ, z odpowiednio zmodyfikowanych wzorów (3.41), (3.42). Np. w drugim etapie projektowania

$$\mathbf{f}_j^{2,nowy} = \frac{\sum_{i, \mathbf{y}_1(i) \in P(\mathbf{f}_j^2)} [g_{m_2(i)}^2 \mathbf{y}_1(i)]}{\sum_{i, \mathbf{y}_1(i) \in P(\mathbf{f}_j^2)} (g_{m_2(i)}^2)^2} \quad (3.85)$$

$$g_m^{2,nowy} = \frac{\sum_{i, \mathbf{y}_1(i) \in P(g_m^2)} [\mathbf{y}_1^t(i) \mathbf{f}_{j_2(i)}^2]}{\sum_{i, \mathbf{y}_1(i) \in P(g_m^2)} [(\mathbf{f}_{j_2(i)}^2)^t \mathbf{f}_{j_2(i)}^2]} \quad (3.86)$$

Dla SGVQ z predykcją długookresową, po zaprojektowaniu słowników w pętli otwartej, można uruchomić koder w pętli zamkniętej. Umożliwia to powtórne zaprojektowanie kwantyzatora wzmocnień dla pierwszego etapu, oraz wszystkich słowników dla kolejnych etapów. Proces ten można powtarzać wielokrotnie.

Opisany tu klasyczny algorytm projektowania słowników SGVQ nie bierze pod uwagę ortogonalizacji słowników kształtów względem wektorów wybranych w poprzednich etapach modelowania. Już słownik wykorzystywany w drugim etapie, $\{\mathbf{f}_j^2\}$, powinien być zortogonalizowany względem wektora $\mathbf{f}_{j_1(i)}^1$ (wzór (3.65)):

$$\bar{\mathbf{f}}_j^2 = \mathbf{P}(\mathbf{f}_{j_1(i)}^1) \mathbf{f}_j^2 \quad (3.87)$$

gdzie $\mathbf{P}(\mathbf{f}) = \mathbf{I} - \frac{\mathbf{f} \mathbf{f}^t}{\|\mathbf{f}\|^2}$ jest operatorem ortogonalizacji względem \mathbf{f} .

Projektując słowniki wykorzystywane w drugim etapie, należy dążyć do minimalizacji błędu kwantowania sekwencji treningowej:

$$D = \sum_i \|\mathbf{y}_1(i) - \bar{g}_{m_2(i)}^2 \bar{\mathbf{f}}_{j_2(i)}^2\|^2 = \sum_i \|\mathbf{y}_1(i) - \bar{g}_{m_2(i)}^2 \mathbf{P}(\mathbf{f}_{j_1(i)}^1) \mathbf{f}_{j_2(i)}^2\|^2 \quad (3.88)$$

Odpowiedni centroid dla kształtu można teraz obliczyć rozwiązując układ równań $\frac{\partial D}{\partial \bar{\mathbf{f}}_j^2} = \mathbf{0}$, który, po przekształceniach, można zapisać w następującej postaci:

$$\left[\sum_{i, \mathbf{y}_1(i) \in P(\mathbf{f}_j^2)} (\bar{g}_{m_2(i)}^2)^2 \mathbf{P}(\mathbf{f}_{j_1(i)}^1) \right] \mathbf{f}_j^{2, \text{nowy}} = \sum_{i, \mathbf{y}_1(i) \in P(\mathbf{f}_j^2)} \bar{g}_{m_2(i)}^2 \mathbf{P}(\mathbf{f}_{j_1(i)}^1) \mathbf{y}_1(i) \quad (3.89)$$

Sumowanie we wzorze (3.89) przeprowadza się dla wszystkich wektorów sekwencji treningowej, spełniających warunek $\mathbf{y}_1(i) \in P(\mathbf{f}_j^2)$, w wyniku ostatnio dokonanego podziału. Należy odróżnić oznaczenie komórki Voronoi'a (P) od oznaczenia operatora ortogonalizacji (\mathbf{P}). W wyprowadzeniu powyższego wzoru wzięto pod uwagę właściwość symetrii ($\mathbf{P}^t = \mathbf{P}$) oraz idempotencji ($\mathbf{P}\mathbf{P} = \mathbf{P}$) tego operatora.

Podobnie można obliczyć centroid \bar{g}_m^2 rozwiązując układ równań $\frac{\partial D}{\partial \bar{g}_m^2} = 0$:

$$\bar{g}_m^{2, \text{nowy}} = \frac{\sum_{i, \mathbf{y}_1(i) \in P(\bar{g}_m^2)} [\mathbf{y}_1^t(i) \mathbf{P}(\mathbf{f}_{j_1(i)}^1) \mathbf{f}_{j_2(i)}^2]}{\sum_{i, \mathbf{y}_1(i) \in P(\bar{g}_m^2)} \|\mathbf{P}(\mathbf{f}_{j_1(i)}^1) \mathbf{f}_{j_2(i)}^2\|^2} \quad (3.90)$$

Należy zwrócić uwagę, że projektowany słownik wzmocnień "współpracuje" ze zortogonalizowanym słownikiem kształtów. Zaproponowany algorytm projektowania słowników, uwzględniający ich ortogonalizację, ma zastosowanie w wielostopniowym SGVQ. Odpowiednie wzory dla wielostopniowego kodera CELP można znaleźć w [206].

3.4.6 Porównanie algorytmów modelowania

W celu porównania algorytmów przeszukiwania i projektowania słowników zasymulowano dwustopniowy kwantyzator wektorowy typu kształt- wzmocnienie (SGVQ) o przepływności

binarnej 9.6 kbit/s, w którym pierwszy etap stanowi predykcja długookresowa (LTP) [64]. Ze względu na złożoność obliczeniową wymiar przetwarzanych wektorów N ograniczono do 20. Dla częstotliwości próbkowania 8 kHz implikuje to kwantowanie z rozdzielczością 24 bitów/wektor. Bity podzielono w równym stosunku między oba etapy modelowania. Osobnym problemem jest podział bitów, w obrębie każdego etapu, między kwantyzatory kształtu i wzmocnienia. Wyniki odpowiednich eksperymentów podano w tab.3.3. Najlepsze wyniki

Tabela 3.3: Dwustopniowy SGVQ z LTP o przepływności binarnej 9.6 kbit/s - SNR_{seg} dla różnych rozdziałów bitów między kwantyzatory kształtu i wzmocnienia (L_s - liczba wektorów kształtu, L_g - liczba poziomów kwantyzacji wzmocnienia)

LTP		SGVQ		SNR_{seg} [dB]
L_s	L_g	L_s	L_g	
128	32	128	32	11.29
128	32	64	64	10.94
128	32	256	16	11.67
256	64	256	16	11.38

osiągnięto kodując kształt w słowniku adaptacyjnym w 7 bitach, odpowiednie wzmocnienie w 5 bitach, kształt w słowniku stałym w 8 bitach, a odpowiednie wzmocnienie w 4 bitach. Wartości te nie są krytyczne, zostały jednak utrzymane w symulacjach, których wyniki podano w tab.3.4.

Tabela 3.4: Dwustopniowy SGVQ z LTP o przepływności binarnej 9.6 kbit/s - SNR_{seg} [dB] dla różnych algorytmów przeszukiwania i projektowania słowników

algorytm modelowania	algorytm projekt. słowników	jeden kwantyz. wzmocnienia		wiele kwantyz. wzmocnienia	
		fraza treningowa	inna fraza	fraza treningowa	inna fraza
standardowy	standardowy, w pętli otwartej	11.67	10.66	12.14	11.26
z korekcją wzmocnienia	standardowy, w pętli otwartej	11.95	11.04	12.43	11.67
ortogonalizacja	standardowy, w pętli otwartej	12.12	11.30	12.58	11.75
ortogonalizacja	ortogonalizacja, w pętli otwartej	12.60	11.72	12.80	11.82
ortogonalizacja	ortogonalizacja, w pętli zamkniętej	12.90	11.72	12.98	12.03

W tab.3.4 zamieszczono wyniki dla sekwencji treningowej (wykorzystanej do projektowania słowników) i innej frazy, nie należącej do sekwencji treningowej. Przetestowano również

podejście, opisane w p.3.3.3: w drugim etapie modelowania wykorzystano wiele kwantyzatorów wzmocnienia (dla każdego kształtu osobny kwantyzator wzmocnienia). Porównano następujące algorytmy modelowania sygnału (przeszukiwania słowników): standardowy algorytm iteracyjny (rzutowanie błędu otrzymanego w poprzednim etapie modelowania), algorytm iteracyjny z korekcją wzmocnień (po wykonaniu algorytmu w wersji standardowej, oblicza się nowe wartości niekwantowanych wzmocnień g^1, g^2 tak, aby model $\tilde{\mathbf{y}} = \sum_{k=1}^2 g^k \mathbf{f}_{j_k}^k$ stanowił rzut ortogonalny \mathbf{y}_0 na podprzestrzeń, tym wypadku płaszczyznę) oraz algorytm lokalnie optymalny (ortogonalizacja słownika wykorzystywanego w drugim etapie względem wektora wybranego w pierwszym etapie modelowania).

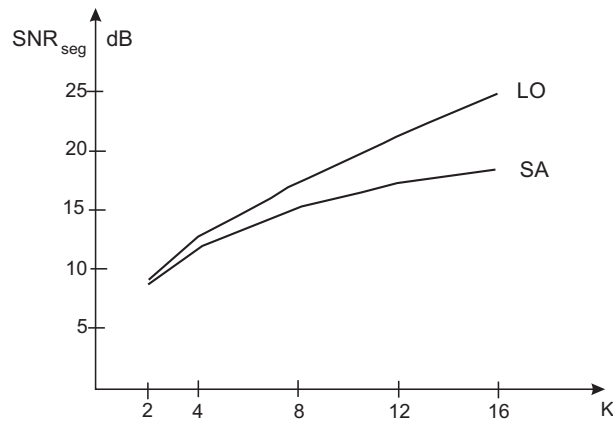
Słowniki wykorzystywane w drugim etapie projektowano metodą standardową (dla SGVQ z pojedynczym kwantyzatorem wzmocnienia wg wzorów (3.85), (3.86)), następnie uwzględniono ortogonalizację słownika kształtów (wzory (3.89), (3.90)), w końcu przeprojektowano słowniki w pętli zamkniętej. Kolejne modyfikacje prowadziły do stopniowej poprawy jakości sygnału (zwiększenie SNR_{seg}).

Podobne badania, przeprowadzone dla innych fraz mowy, udokumentowano w [63]. W pierwszym etapie zasymulowano klasyczną wersję kodera SGVQ z LTP, bez ortogonalizacji słownika. Zastosowano standardowy algorytm modelowania oraz słowniki obliczane w układzie otwartym. Dla testowanej frazy sygnału mowy otrzymano segmentowy stosunek mocy sygnału do mocy zniekształceń $SNR_{seg} = 12.5$ dB. Samo przeprowadzenie ortogonalizacji słownika poprawiło jakość sygnału w niewielkim stopniu ($SNR_{seg} = 12.8$ dB). Zastosowanie nowego algorytmu projektowania słowników, tzn. nowych sposobów obliczania centroidów (z uwzględnieniem ortogonalizacji), wpłynęło na poprawę SNR_{seg} do 13.5 dB. Dalszą poprawę jakości sygnału mowy osiągnięto przeprowadzając projektowanie słowników w układzie zamkniętym ($SNR = 14.3$ dB). Należy zwrócić uwagę na fakt, że poprawę jakości sygnału mowy uzyskano bez zwiększenia przepływności binarnej - nieznacznie tylko komplikując układ kodera (konieczność przeprowadzania ortogonalizacji).

Badania porównawcze algorytmów modelowania sygnału przeprowadzono w szerszym zakresie na zasymulowanych koderach CELP [145]. Podstawowy wariant kodera zawiera K etapów (stopni) modelowania. W pierwszym etapie wykorzystywany jest słownik adaptacyjny (predykcja długookresowa) a w pozostałych etapach ten sam słownik stały. Słownik stały ma strukturę "gęstą" (wypełniony jest sekwencją o rozkładzie gaussowskim) lub "rzadką" (w 90% jest wypełniony zerami). Zarówno słownik adaptacyjny, jak i stały zawiera 128 wektorów 40-wymiarowych. Długość ramki, w której oblicza się parametry filtra predykcyjnego, wynosi 20 ms (160 próbek pobieranych z częstotliwością 8 kHz). W ramce mieszczą się 4 wektory o wymiarze $N = 40$.

Przepływność binarną kodera CELP oszacowano na $(2.25 + 2.2 K)$ kbit/s. Złożoność obliczeniową, przy modelowaniu sygnału z wykorzystaniem standardowego algorytmu iteracyjnego oraz słownika stałego o strukturze "gęstej", oszacowano na $(5.2 + 1.02(K - 2) + 0.17K)$ kbit/s. Wzór jest słuszny dla $K > 1$. Jakość sygnału badano, obliczając SNR w ujęciu segmentowym, dla mowy męskiej i żeńskiej o łącznym czasie trwania 24 s.

Na rys.3.11 porównano standardowy i lokalnie optymalny algorytm modelowania. Przewaga tego ostatniego zaznacza się wyraźniej dla koderów CELP o wielu etapach modelowania, przy czym nawet dla $K = 2$ lub 3 nie jest do pominięcia.



Rysunek 3.11: SNR_{seg} dla K - stopniowego kodera CELP: (SA) - standardowy algorytm iteracyjny, (LO) - algorytm lokalnie optymalny z ortogonalizacją słownika

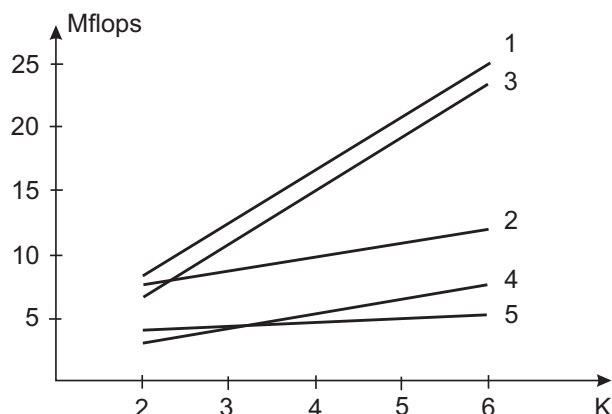
Lokalnie optymalny algorytm modelowania może być zaimplementowany na wiele sposobów, różniących się złożonością obliczeniową. Porównanie tych implementacji (rys.3.12) wypada zdecydowanie na korzyść rekursywnego zmodyfikowanego algorytmu Grama - Schmidta, zarówno dla słownika o strukturze "gęstej", jak i "rzadkiej". Dla słownika o strukturze "rzadkiej" dobrze spisuje się algorytm filtracji odwrotnej - ten sam algorytm dla słownika o strukturze "gęstej" jest bardzo nieefektywny (złożoność obliczeniowa > 40 Mflops, tego wariantu nie umieszczono na wykresie).

Za podstawę do dalszych badań przyjęto koder CELP o $K = 3$ etapach modelowania. Jego przepływność binarna wynosi $2.25 + 2.2 \cdot 3 = 8.85$ kbit/s. Za rozwiązanie standardowe (odniesienia) uznano koder wykorzystujący standardowy iteracyjny algorytm modelowania sygnału, charakteryzujący się złożonością obliczeniową 6.8 Mflops. W wyniku symulacji (wykorzystano wspomniane frazy o długości 24 s) otrzymano $SNR_{seg} = 10.4$ dB. Ten wynik umieszczono na rys.3.13 pod numerem 1, traktując go jako punkt odniesienia. Badając inne kodery CELP o zbliżonej przepływności binarnej, notowano przyrost SNR_{seg} względem wspomnianego punktu odniesienia.

Porównując kodery nr 2 i 3 można się przekonać o wyższości rekursywnego, zmodyfikowanego algorytmu Grama - Schmidta nad dokonywaniem korekcji wzmacnień w każdej iteracji.

Połączenie słownika adaptacyjnego i stałego w jeden słownik "mieszany" prowadzi do poprawy jakości sygnału, jednak kosztem niewielkiego wzrostu przepływności binarnej (kodery 4 i 5).

W celu zmniejszenia złożoności obliczeniowej nadano słownikowi stałemu strukturę "nakładkową", "wycinając" kolejne wektory z długiej sekwencji gaussowskiej, według zasady pokazanej na rys.3.14. Taki słownik daje się łatwo filtrować, gdyż kolejne wektory zawierają prawie taką samą sekwencję próbek sygnału. Filtrowanie kolejnego wektora wymaga uwzględnienia tylko jednej próbki, tworzącej pierwszą składową tego wektora. Tak zrealizowany koder nr 6 jest mniej złożony obliczeniowo, jednak zauważa się spadek jakości mowy



Rysunek 3.12: Złożoność obliczeniowa różnych implementacji lokalnie optymalnego przeszukiwania słownika w K -stopniowym koderze CELP: (1) - prosta ortogonalizacja słownika o strukturze "gęstej", (2) - rekursywny zmodyfikowany algorytm Grama - Schmidta, słownik o strukturze "gęstej", (3) - prosta ortogonalizacja słownika o strukturze "rzadkiej", (4) - filtracja odwrotna, słownik "rzadki", (5) - rekursywny zmodyfikowany algorytm Grama - Schmidta, słownik o strukturze "rzadkiej"

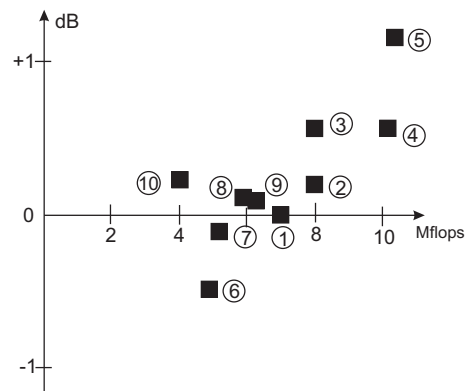
wskutek dużej korelacji wektorów słownika. Ten spadek można skompensować stosując jednocześnie rekursywny zmodyfikowany algorytm Grama - Schmidta (koder nr 8).

Inną metodę zmniejszenia złożoności obliczeniowej zaproponowano w [197]. Polega ona na filtrowaniu słownika w sposób uproszczony, drogą przetwarzania autokorelacji filtrowanych wektorów i autokorelacji odpowiedzi impulsowej filtru. Praktycznie nie zauważa się pogorszenia jakości mowy (koder nr 7), a nawet można osiągnąć niewielką jej poprawę, stosując jednocześnie rekursywny zmodyfikowany algorytm Grama - Schmidta (koder nr 9).

Porównując odpowiednie kodery korzystające ze standardowego i lokalnie optymalnego algorytmu modelowania (tzn. kodery 1-3, 4-5, 6-8, 7-9) stwierdza się poprawę jakości mowy w wyniku stosowania lokalnie optymalnego wyboru wektorów.

Szukając dróg prowadzących do zmniejszenia złożoności obliczeniowej zasymulowano koder CELP, w którym 40-wymiarowy wektor pobudzenia powstaje przez konkatenację czterech 10-wymiarowych wektorów. Wszystkim 10-wymiarowym wektorom przypisany jest ten sam współczynnik wzmocnienia, można jednak zmieniać polaryzację każdego wektora. W procesie modelowania sygnału korzysta się z algorytmu sukcesywnej minimalizacji kąta, opisanego w p.3.4.4. Algorytm ten daje znaczną redukcję złożoności obliczeniowej bez pogorszenia jakości sygnału (koder nr10). Porównując SNR_{seg} należy zwrócić uwagę na nieco większą przepływność binarną tego kodera (9.2 kbit/s).

Algorytm sukcesywnej minimalizacji kąta znalazł również zastosowanie w koderze CELP z predykcją długookresową i pobudzeniem ternarnym, opisanym w [56]. Przepływność binarna kodera wynosi 9.6 kbit/s, długość wektora pobudzenia wynosi $N = 40$ (5 ms). Każdy wektor zawiera 4 impulsy o amplitudach ± 1 . W tabl.3.5 porównano trzy algorytmy modelowania: algorytm optymalny (wymagający przetestowania wszystkich położenia impulsów w



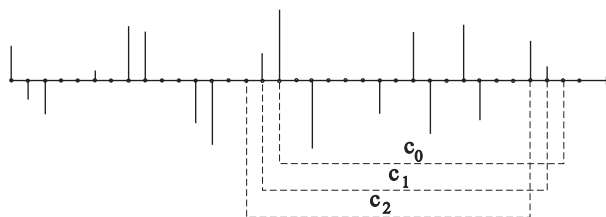
Rysunek 3.13: Jakość (SNR_{seg} względem rozwiązania standardowego) i złożoność obliczeniowa (Mflops) koderów CELP o zbliżonej przepływności binarnej: (1) - standardowy algorytm iteracyjny (8.85 kbit/s), (2) - korekcja wzmacnień po każdej iteracji (8.85 kbit/s), (3) - rekursywny zmodyfikowany algorytm Grama - Schmidta (8.85 kbit/s), (4) - standardowy algorytm iteracyjny ze słownikiem "mieszanym" (9.45 kbit/s), (5) - rekursywny zmodyfikowany algorytm Grama - Schmidta ze słownikiem "mieszanym" (9.45 kbit/s), (6) - słownik o strukturze "nakładkowej" (8.85 kbit/s), (7) - uproszczona filtracja słownika (8.85 kbit/s), (8) - rekursywny zmodyfikowany algorytm Grama - Schmidta ze słownikiem o strukturze "nakładkowej" (8.85 kbit/s), (9) - rekursywny zmodyfikowany algorytm Grama - Schmidta z uproszczoną filtracją słownika (8.85 kbit/s), (10) - konkatencja krótkich wektorów (9.2 kbit/s)

obrębie wektora, tzn. 731120 przypadków), algorytm sukcesywnej minimalizacji kąta (308 testów) oraz algorytm polegający na maksymalizacji licznika wyrażenia (3.81) (nie wymaga testowania). Badania przeprowadzono dla 4 fraz o czasie trwania około 6 sekund. Z badań wynika, że zaproponowany algorytm sukcesywnej minimalizacji kąta bardzo niewiele ustępuje algorytmowi optymalnemu, a jednocześnie zapewnia małą złożoność obliczeniową. Algorytm maksymalizacji licznika jest jeszcze prostszy, lecz prowadzi do zauważalnego pogorszenia jakości mowy.

Testowane kodery CELP są K -stopniowe, w pierwszym etapie wykonuje się predykcję długookresową, a w kolejnych $K - 1$ etapach rozmieszcza się impulsy. Słownik impulsów może być zortogonalizowany względem wektora wybranego w pierwszym etapie, wymaga to zastąpienia kryterium (3.81) przez kryterium (3.84). W tab.3.6 porównano oba warianty: bez i z ortogonalizacją. Ortogonalizacja zawsze zapewniała wzrost SNR_{seg} .

3.4.7 Przykład realizacji praktycznej

Przykładem rozwiązania praktycznego jest koder o przepływnościach 2.4/4.8/8 kbit/s, przeznaczony dla transmisji utajnionej mowy w sieciach publicznych i komórkowych [55, 66]. Koder został zrealizowany w Zakładzie Systemów Teletransmisyjnych Instytutu Telekomu-



Rysunek 3.14: Słownik o strukturze "rzadkiej" i "nakładkowej"

Tabela 3.5: CELP o przepływności 9.6 kbit/s z LTP i słownikiem ternarnym - SNR_{seg} [dB] dla trzech algorytmów modelowania

Algorytm	frazą			
	1	2	3	4
min. kąta	13.4	11.3	14.1	10.9
optymalny	13.6	11.5	14.2	11.1
max. licznika	11.6	9.9	11.9	9.3

nikacji Politechniki Warszawskiej, we współpracy z Wojskowym Instytutem Łączności w Zegrzu i Zakładem Projektowo- Wdrożeniowym TEL-KA z Warszawy.

Koder zawiera dwa warianty CELP o przepływnościach 4.8 kbit/s i 8 kbit/s (z uzupełnieniem do 9.6 kbit/s) oraz wokoder predykcyjny 2.4 kbit/s (o wokoderach będzie mowa w jednym z następnych rozdziałów pracy). Koder jest przystosowany do działania w sieci PSTN i GSM. Implementacji koderu dokonano na zmiennoprzecinkowym procesorze sygnałowym TMS320C31.

Jedną z najbardziej udanych wersji koderu CELP o przepływności 4.8 kbit/s jest koder stanowiący podstawę standardu amerykańskiego 1016-CELP [12]. Standard ten powstawał w okresie, kiedy nie było zmiennoprzecinkowych procesorów sygnałowych - jego projektanci przewidywali implementacje w oparciu o wyspecjalizowane rozwiązania sprzętowe lub szybkie procesory stałoprzecinkowe. Wybrany wariant koderu CELP ma wiele cech koderu amerykańskiego, jednak uwzględniono w nim niektóre późniejsze rozwiązania (np. klasyfikację sygnału i postfiltrację). Wybrany koder CELP 8 kbit/s ma wiele cech standardów ITU-T G729 [100] i G723.1 [99] (np. długości ramek i wektorów), jednak zachował wiele cech standardu 1016-CELP (np. słownik stochastyczny). Ze względu na zastosowania jego przepływność binarna została powiększona do 9.6 kbit/s.

Koder dla transmisji utajnionej mowy i danych został pomyślany jako urządzenie, za pomocą którego można przesyłać mowę i dane zarówno pomiędzy abonentami telefonicznej sieci stacjonarnej jak i ruchomej sieci GSM. Do realizacji powyższej idei wykorzystano usługę transmisji danych, dostępną abonentom telefonicznej sieci stacjonarnej, abonentom sieci komórkowej systemu GSM oraz pomiędzy abonentami obydwu sieci.

Przy współpracy z siecią stacjonarną interfejs komunikacyjny spełnia rolę modemu o prze-

Tabela 3.6: CELP z LTP bez i z ortogonalizacją - SNR_{seg} [dB] w funkcji K-1 - liczby niezerowych wartości w 40-próbkowym wektorze pobudzenia

K-1	fraza 1		fraza 3	
	bez ortog.	ortog.	bez ortog.	ortog.
1	9.7	10.0	10.2	10.4
2	11.3	11.8	11.8	12.3
3	12.8	13.1	13.0	13.7
4	13.4	13.9	14.1	14.5
5	13.8	14.6	14.6	15.2
6	14.4	14.9	15.2	15.9
7	14.9	15.4	15.6	16.1
8	15.1	15.8	15.9	16.4

plywności do 14.4 kbit/s (standard V.32bis). Szybkość transmisji jest wybierana w funkcji jakości zestawionego łącza i może wynosić 9.6 kbit/s, 4.8 kbit/s lub 2.4 kbit/s. Podczas pracy poprzez sieć komórkową interfejs komunikacyjny umożliwia komunikację z telefonem komórkowym GSM. Telefon komórkowy jest wykorzystywany w trybie transmisji danych o szybkości do 9.6 kbit/s.

W nadajniku sygnał mowy poddawany jest obróbce wstępnej, mającej na celu odfiltrowanie składowej stałej i przydźwięku sieci, korekcję charakterystyki mikrofonu, oraz automatyczną regulację poziomu sygnału. W każdej ramce (30 ms, 240 próbek) dokonuje się jednokrotnego (dla wersji 4.8 kbit/s) lub dwukrotnego (dla wersji 8 kbit/s) obliczenia parametrów predykcyjnych, tzw. współczynników widma prążkowego ($LSF_1, LSF_2 \dots, LSF_{10}$). Dla wersji 8 kbit/s sześć razy w ramce (co 5 ms, 40 próbek) wyznaczane są wektory pobudzenia. Długości wektorów odpowiadają standardowi G729. Dla wersji 4.8 kbit/s wektory pobudzenia są 60-wymiarowe (7.5 ms) i są czterokrotnie wyznaczane w ramce. Długości wektorów są takie same, jak w standardzie G723.1.

W zaprojektowanym koderze stosuje się słownik stochastyczny o specjalnej strukturze, umożliwiającej jego szybką filtrację i przeszukiwanie - rys.3.14. Liczy on $L_s = 512$ wektorów (opcjonalnie 256). Dla mowy dźwięcznej o okresie tonu krtaniowego $T_0 < 40$ lub $T_0 < 60$ próbek (tzn. mniejszym od długości wektora) przeprowadza się "periodyzację" wektorów słownika, powtarzając ten sam ciąg próbek z okresem równym opóźnieniu w predykcji długookresowej.

W koderze 8 kbit/s zastosowano schemat kodowania parametrów przedstawiony w tab.3.7. Przepływność trochę przekracza 8 kbit/s, co jest dopuszczalne (pełna przepływność wynosi 9.6 kbit/s, wraz z zabezpieczeniem kodowym). W razie konieczności można przepływność łatwo zredukować do 8 kbit/s, kosztem obniżenia dokładności kodowania LSF. W koderze 4.8 kbit/s (tab.3.8) przepływność jest nieco mniejsza, po dodaniu 1 bitu osiąga wartość nominalną.

Odtworzony po stronie odbiorczej sygnał podlega jeszcze tzw. adaptacyjnej postfiltracji,

Tabela 3.7: Schemat kodowania parametrów w 30 ms ramce - przepływność 8.1 kbit/s

parametr	liczba bitów
LSF_1, \dots, LSF_{10}	$2 \cdot (4, 5, 5, 4, 4, 4, 4, 4, 3, 3)$
indeks LTP	6*8
indeks stoch.	6*9
wzmocnienie LTP	6*5
wzmocnienie stoch.	6*5
synchronizacja	1
razem	243

Tabela 3.8: Schemat kodowania parametrów w 30 ms ramce - przepływność 4.767 kbit/s

parametr	liczba bitów
LSF_1, \dots, LSF_{10}	3,4,4,4,4,3,3,3,3,3
indeks LTP	4*8
indeks stoch.	4*9
wzmocnienie LTP	4*5
wzmocnienie stoch.	4*5
synchronizacja	1
razem	143

w celu wzmocnienia zakresów formantowych (koncentracji energetycznych na skali częstotliwości) kosztem zakresów międzyformantowych, mniej istotnych z punktu widzenia zrozumiałości mowy. Postfiltracja w większym stopniu wzmacnia mowę niż szum kwantyzacji, ponadto szum jest mniej dokuczliwy dzięki zjawisku maskowania sygnałem mowy. W procesie postfiltracji nie można dopuścić do wzrostu energii sygnału - konieczny jest algorytm kontroli wzmocnienia w postfiltrze.

Badania jakości sygnału mowy przeprowadzono metodami obiektywnymi, mierząc SNR_{seg} . Wyniki badań koderów CELP dla niektórych fraz testowych podano w tab.3.9. Duża wartość SNR_{seg} dla sygnału sinusoidalnego świadczy o prawidłowym działaniu algorytmów koderów CELP.

Większe znaczenie, przy tak małej przepływności binarnej, mają badania oparte na odsłuchach. Przeprowadzono (pod kierownictwem dr S.Kuli) porównawcze badania wyrazistości logatomowej, obejmując nimi, oprócz trzech omawianych koderów, także koder GSM i analogowe łącze telefoniczne. Badania przeprowadzono używając 1200 logatomów zebranych w 12 listach po sto logatomów każda. W badaniach uczestniczyły łącznie 3 osoby, z tego dwie osoby były słuchaczami, zaś jedna osoba czytała listy logatomowe (lektor). Test przeprowadzono w rzeczywistych warunkach korzystania z połączeń telefonicznych, a więc w warunkach nie odpowiadających normom pomiarowym. Z tego względu wyniki (tabl.3.10) mają jedynie znaczenie porównawcze. Badania porównawcze wyrazistości logatomowej wykazują dobrą jakość mowy dla kodera 8 kbit/s (porównywalną z koderem GSM-FR), nieco gorszą jakość

Tabela 3.9: SNR w ujęciu segmentowym [dB] dla kilku sygnałów testowych

fraza	CELP 4.8 kbit/s	CELP 8 kbit/s
sinus 350 Hz	34.5	37.4
m. żeńska 1	10.6	13.2
m. żeńska 2	8.5	10.2
m. żeńska 3	8.5	11.2
m. żeńska 4	10.7	13.8
m. męska 1	7.4	9.9
m. męska 2	7.4	9.7
m. męska 3	9.2	11.9

mowy dla kodera 4.8 kbit/s oraz bardziej wyraźny spadek jakości przy szybkości 2.4 kbit/s.

Tabela 3.10: Badanie porównawcze wyrazistości logatomowej

koder	przepływność [kbit/s]	rozpoznawalność
telefon analog.		82.7 %
GSM	13	66.7%
CELP	8.1	63.5%
CELP	4.8	61.3%
wokoder	2.4	50.5%

Koder jest odporny na przekłamania: zniekształcenia sygnału mowy są wyraźnie zauważalne dopiero przy BER około 0.05.

Rozdział 4

Kodowanie transformaty z wykorzystaniem kwantyzacji wektorowej

4.1 Wstęp

Kodowanie z wykorzystaniem transformaty (najczęściej w dziedzinie częstotliwości) jest zagadnieniem szerokim, jego omówienie wymagałoby napisania odrębnej rozprawy. Z tego względu poruszone tu będą jedynie pewne aspekty tego zagadnienia, związane z wykorzystaniem kwantyzacji wektorowej.

Dzięki dekorelującym właściwościom transformaty, podział na części przetransformowanego wektora i kwantowanie wektorowe każdej części z osobna prowadzi do mniejszych zniekształceń niż analogiczna operacja wykonana na wektorze nietransformowanym (tzw. zysk transformaty). Problemem obliczenia zysku kodowania transformaty zajmowali się Soman i Vaidyanathan [186] dla kwantowania skalarnego oraz Li i Zhang [119] dla kwantowania wektorowego (z wykorzystaniem PCVQ). Autor rozprawy oraz N. Moreau i in. dokonali porównania zysku kodowania dla kilku transformat, w tym dla zaproponowanego przez nich układu filtrów o strukturze hierarchicznej opisanego w [144].

Problemem odpowiedniego podziału przetransformowanego wektora, w celu poprawienia zysku transformaty, zajmowali się Li i Zhang [119, 118]. Problem rozdziału zasobu bitów między kwantyzatory wektorowe operujące w dziedzinie transformaty, a także problem łącznego określenia wymiarów kwantowanych wektorów i przydzielenia im optymalnej liczby bitów został rozwiązany przez Cupermana [34, 33]. Moreau i Dymarski [143, 145], a także Dymarski i Ratre [63, 64] badali kwantyzatory wektorowe typu kształt - wzmocnienie (SGVQ) z wykorzystaniem transformaty Karhunen - Loevego, charakteryzującej się najlepszymi właściwościami dekorelującymi i oferującej największy zysk transformaty.

Jest wiele publikacji dotyczących koderów sygnałów akustycznych w dziedzinie częstotliwości. Kodery transformaty są opisane m.in. w pracy Mahieux [126], patrz też Mahieux i Petit [127], Davidson, Fiedler i Antill [35]. Chang, Gray i May [23] oraz Chan i Gersho [21, 22] wykorzystują transformatę i kwantowanie wektorowe w podprzedziałach częstotliwości. W koderze TPC (*transform predictive coding*, patrz Chen i Wang [30], Chen [29]) oraz koderze TCX (*transform coded excitation*, patrz Lefebvre, Salami, Laflamme i Adoul [117]) łączy się kodowanie transformaty z predykcją długookresową i adaptacyjną preemfazą /deemfazą (w oparciu o predykcję krótkookresową). Podobny schemat znajdziemy w pracy Moreau i Dymarskiego [144], z tym, że wykorzystano transformatę z nakładaniem, opisaną w pracach Malvara [131, 129, 130]. W toku dalszych prac ci sami autorzy opracowali koder transformaty o przepływności 64 kbit/s i zmniejszonym opóźnieniu [103, 148]. Opracowali również wersję "hierarchiczną" tego koder, umożliwiającą prawidłowy odbiór sygnału akustycznego przy zmniejszonej przepływności binarnej, w wyniku pogorszenia właściwości transmisyjnych kanału (np. utraty części transmitowanych pakietów) [146, 58, 147] - patrz p.4.3.

4.2 Wyznaczanie zysku transformaty

4.2.1 Analiza problemu w postaci ogólnej

W p.3.1.2 wykazano, że kwantyzator wektorowy jest koderem asymptotycznie optymalnym. Czy stosowanie transformaty wraz z kwantyzacją wektorową ma w takiej sytuacji sens? Najpierw wykażemy, że najprostsze rozwiązanie, tj. transformacja N - wymiarowych wektorów, a potem ich kwantowanie wektorowe, nie prowadzi do zmiany SNR .

Niech \mathbf{T} będzie macierzą stopnia $N \times N$, opisującą transformatę unitarną (tzn. $\mathbf{T}\mathbf{T}^t = \mathbf{T}^t\mathbf{T} = \mathbf{I}$, gdzie \mathbf{I} - macierz jednostkowa). Transformację wektora \mathbf{x}^1 można opisać następująco:

$$\mathbf{y} = \mathbf{T} \mathbf{x} \quad (4.1)$$

Z zachowania normy:

$$\mathbf{y}^t \mathbf{y} = \mathbf{x}^t \mathbf{T}^t \mathbf{T} \mathbf{x} = \mathbf{x}^t \mathbf{I} \mathbf{x} = \mathbf{x}^t \mathbf{x} \quad (4.2)$$

wynika, że średnia energia przypadająca na składową wektora \mathbf{y} jest taka sama, jak średnia energia przypadająca na składową wektora \mathbf{x} , czyli moc (równa wariancji, ze względu na zerową wartość średnią) sygnału wejściowego:

$$\sigma_y^2 = \frac{1}{N} E[\mathbf{y}^t \mathbf{y}] = \frac{1}{N} E[\mathbf{x}^t \mathbf{x}] = \frac{1}{N} N \sigma_x^2 = \sigma_x^2 \quad (4.3)$$

Z kolei z zależności odnoszących się do macierzy korelacji:

$$\mathbf{R}^N(\mathbf{y}) = E[\mathbf{y}\mathbf{y}^t] = \mathbf{T} E[\mathbf{x}\mathbf{x}^t] \mathbf{T}^t = \mathbf{T} \mathbf{R}^N(\mathbf{x}) \mathbf{T}^t \quad (4.4)$$

wynika zachowanie wyznacznika tej macierzy:

$$\det \mathbf{R}^N(\mathbf{y}) = \det[\mathbf{T} \mathbf{R}^N(\mathbf{x}) \mathbf{T}^t] = \det[\mathbf{T}\mathbf{T}^t] \det \mathbf{R}^N(\mathbf{x}) = \det[\mathbf{I}] \det \mathbf{R}^N(\mathbf{x}) = \det \mathbf{R}^N(\mathbf{x}) \quad (4.5)$$

Kwantując bezpośrednio wektor \mathbf{x} przy użyciu kwantyzatora wektorowego o rozdzielczości b bitów na składową, otrzymuje się energię błędu kwantyzacji przypadającą na składową wektora, jak we wzorze (3.8):

$$\sigma_e^2(b, N) = c(N) \left[\det \mathbf{R}^N(\mathbf{x}) \right]^{1/N} 2^{-2b}$$

Odpowiednia wartość SNR dana jest wzorem (3.9):

$$SNR(b, N) = \frac{\sigma_x^2}{\sigma_e^2(b, N)} = c^{-1}(N) \frac{\sigma_x^2}{\left[\det \mathbf{R}^N(\mathbf{x}) \right]^{1/N}} 2^{2b}$$

Kwantując przetransformowany wektor \mathbf{y} , otrzymuje się energię błędu kwantyzacji, przypadającą na składową tego wektora, równą (na podstawie (4.5)) $\sigma_e^2(b, N)$:

$$\sigma_e^2(\mathbf{T}, b, N) = c(N) \left[\det \mathbf{R}^N(\mathbf{y}) \right]^{1/N} 2^{-2b} = c(N) \left[\det \mathbf{R}^N(\mathbf{x}) \right]^{1/N} 2^{-2b} = \sigma_e^2(b, N) \quad (4.6)$$

¹Zakłada się, że wektor próbek sygnału akustycznego \mathbf{x} ma zerową wartość średnią: $E(\mathbf{x}) = \mathbf{0}$. Wynika stąd równość mocy $E[\mathbf{x}^t \mathbf{x}]/N$ i wariancji $E[(\mathbf{x} - E(\mathbf{x}))^t (\mathbf{x} - E(\mathbf{x}))]/N$, oraz równość macierzy korelacji $E[\mathbf{x}\mathbf{x}^t]$ i kowariancji $E[(\mathbf{x} - E(\mathbf{x}))(\mathbf{x} - E(\mathbf{x}))^t]$. Te równości obowiązują również dla przetransformowanego wektora $\mathbf{y} = \mathbf{T}\mathbf{x}$, gdyż z $E(\mathbf{x}) = \mathbf{0}$ wynika $E(\mathbf{y}) = \mathbf{0}$.

Ostatecznie, na podstawie (4.3) i (4.6) stwierdza się równość wartości SNR w procesie kwantowania wektorów \mathbf{y} i \mathbf{x} :

$$SNR(\mathbf{T}, b, N) = \frac{\sigma_y^2}{\sigma_e^2(\mathbf{T}, b, N)} = \frac{\sigma_x^2}{\sigma_e^2(b, N)} = SNR(b, N) \quad (4.7)$$

Te wyniki można uzasadnić intuicyjnie ze względu na fakt, że transformatę unitarną rozpatruje się jako zmianę układu współrzędnych. Optymalny kwantyzator wektorowy operujący w dziedzinie transformaty (\mathbf{y}) wykorzystuje przetransformowany słownik optymalnego kwantyzatora wektora wejściowego (\mathbf{x}) - czyli ten sam słownik, wyrażony jedynie w innym układzie współrzędnych.

Poprawę SNR , a więc zysk ze stosowania transformaty można uzyskać w sytuacji, gdy transformowane bloki $\bar{\mathbf{x}}$, tworzące wektory o wymiarze \bar{N} , są dłuższe od wektorów przetwarzanych w kwantowaniu wektorowym (tzn. $\bar{N} > N$). W praktyce przetransformowany wektor

$$\bar{\mathbf{y}} = \bar{\mathbf{T}} \bar{\mathbf{x}} \quad (4.8)$$

dzieli się na M części, kwantowanych w osobnych kwantyzatorach wektorowych. Załóżmy na początek, że te części są równe, czyli $\bar{N} = NM$.

Największą wartość SNR można osiągnąć, kwantując \bar{N} -wymiarowy wektor $\bar{\mathbf{x}}$ w całości (transformata nie byłaby wówczas potrzebna):

$$SNR(b, \bar{N}) \geq SNR(b, N)$$

Rozwiązanie takie może być nierealne, ze względu na złożoność obliczeniową kwantyzatora wektorowego. Kwantowanie N -wymiarowych części wektora $\bar{\mathbf{y}}$ może być dokładniejsze, niż kwantowanie N -wymiarowych wektorów \mathbf{x} (ze względu na "dekorelujące" właściwości transformaty), natomiast złożoność obliczeniowa obu rozwiązań jest porównywalna (o ile istnieją szybkie algorytmy obliczania transformaty). Jeśli przez $SNR(\bar{\mathbf{T}}, b, N)$ oznaczy się SNR kwantowania wektorowego N -wymiarowych części wektora $\bar{\mathbf{y}}$, to spełniona jest nierówność:

$$SNR(b, \bar{N}) \geq SNR(\bar{\mathbf{T}}, b, N) \geq SNR(b, N) \quad (4.9)$$

Zysk transformaty przejawia się w odpowiedniej poprawie SNR :

$$G(\bar{\mathbf{T}}) = \frac{SNR(\bar{\mathbf{T}}, b, N)}{SNR(b, N)} = \frac{\sigma_e^2(b, N)}{\sigma_e^2(\bar{\mathbf{T}}, b, N)} \quad (4.10)$$

Równość wynika z warunku zachowania normy (4.3). Dla $N = 1$ otrzymuje się zysk transformaty przy kwantowaniu skalarnym.

Należy teraz wyznaczyć średnią energię błędu kwantyzacji na składową wektora $\bar{\mathbf{y}}$, czyli $\sigma_e^2(\bar{\mathbf{T}}, b, N)$, przenoszoną przez transformatę odwrotną do wektora $\bar{\mathbf{x}}$, jako moc błędu kwantowania przetwarzanego sygnału. Odpowiednie N -wymiarowe części wektora $\bar{\mathbf{y}}$ oznaczymy przez $\bar{\mathbf{y}}_1, \bar{\mathbf{y}}_2, \dots, \bar{\mathbf{y}}_M$:

$$\bar{\mathbf{y}} = \begin{bmatrix} \bar{\mathbf{y}}_1 \\ \vdots \\ \bar{\mathbf{y}}_M \end{bmatrix} = \bar{\mathbf{T}} \bar{\mathbf{x}} \quad (4.11)$$

Każdy z N -wymiarowych wektorów jest kwantowany przy użyciu osobnego VQ ze zoptymalizowanym słownikiem. Odpowiednie słowniki liczą

$$L_1 = 2^{b_1 N}, \dots, L_M = 2^{b_M N}$$

wektorów. Rozdzielczości tych kwantyzatorów, b_1, \dots, b_M spełniają warunek

$$\sum_{k=1}^M b_k = bM \quad (4.12)$$

gdzie b - średnia liczba bitów na składową $\bar{\mathbf{y}}$, czyli rozdzielczość kwantowania sygnału. Poszukiwaną średnią energią błędu kwantyzacji na składową wektora $\bar{\mathbf{y}}$ można teraz zapisać wzorem:

$$\sigma_e^2(\bar{\mathbf{T}}, b, N) = \frac{c(N)}{M} \sum_{k=1}^M [\det \mathbf{R}^N(\bar{\mathbf{y}}_k)]^{1/N} 2^{-2b_k} \quad (4.13)$$

gdzie $\mathbf{R}^N(\bar{\mathbf{y}}_k)$ jest macierzą kowariancji wektora $\bar{\mathbf{y}}_k$ (równą macierzy korelacji, ze względu na zerową wartość średnią tego wektora).

Wartość $\sigma_e^2(\bar{\mathbf{T}}, b, N)$ zależy oczywiście od przetwarzanego sygnału (poprzez macierze kowariancji $\mathbf{R}^N(\bar{\mathbf{y}}_k)$) oraz od rozdziału bitów między M kwantyzatorów (poprzez b_1, \dots, b_M). Będziemy poszukiwać optymalnego rozdziału bitów, minimalizującego moc błędu kwantowania opisaną wzorem (4.13) z warunkiem (4.12).

Ze względu na to, że średnia arytmetyczna zbioru liczb rzeczywistych nieujemnych jest większa lub równa średniej geometrycznej (równość występuje w przypadku, gdy wszystkie te liczby są jednakowe), spełniona jest następująca nierówność:

$$\frac{c(N)}{M} \sum_{k=1}^M [\det \mathbf{R}^N(\bar{\mathbf{y}}_k)]^{1/N} 2^{-2b_k} \geq c(N) \left[\prod_{k=1}^M \det \mathbf{R}^N(\bar{\mathbf{y}}_k) \right]^{1/NM} 2^{-2(b_1 + \dots + b_M)/M} \quad (4.14)$$

którą, ze względu na (4.12), można przepisać w postaci:

$$\frac{c(N)}{M} \sum_{k=1}^M [\det \mathbf{R}^N(\bar{\mathbf{y}}_k)]^{1/N} 2^{-2b_k} \geq c(N) \left[\prod_{k=1}^M \det \mathbf{R}^N(\bar{\mathbf{y}}_k) \right]^{1/NM} 2^{-2b} \quad (4.15)$$

Równość występuje w przypadku, gdy dla każdego k (czyli dla $k = 1, \dots, M$)

$$[\det \mathbf{R}^N(\bar{\mathbf{y}}_k)]^{1/N} 2^{-2b_k} = \left[\prod_{m=1}^M \det \mathbf{R}^N(\bar{\mathbf{y}}_m) \right]^{1/NM} 2^{-2b} = \text{const} \quad (4.16)$$

Stąd optymalny rozdział bitów:

$$b_k = b + \frac{1}{2N} \log_2 [\det \mathbf{R}^N(\bar{\mathbf{y}}_k)] - \frac{1}{2NM} \log_2 \left[\prod_{m=1}^M \det \mathbf{R}^N(\bar{\mathbf{y}}_m) \right], \quad k = 1, \dots, M \quad (4.17)$$

Dla optymalnego rozdziału bitów moc błędu kwantyzacji jest równa

$$\sigma_e^2(\bar{\mathbf{T}}, b, N) = c(N) \left[\prod_{k=1}^M \det \mathbf{R}^N(\bar{\mathbf{y}}_k) \right]^{1/NM} 2^{-2b} \quad (4.18)$$

Zysk transformaty $\bar{\mathbf{T}}$ dla optymalnego rozdziału bitów wynosi:

$$G(\bar{\mathbf{T}}) = \frac{[\det \mathbf{R}^N(\mathbf{x})]^{1/N}}{\left[\prod_{k=1}^M \det \mathbf{R}^N(\bar{\mathbf{y}}_k) \right]^{1/NM}} \leq \frac{c(N) [\det \mathbf{R}^N(\mathbf{x})]^{1/N}}{c(\bar{N}) [\det \mathbf{R}^{\bar{N}}(\bar{\mathbf{x}})]^{1/\bar{N}}} \quad (4.19)$$

Powyższe rozważania można uogólnić na przypadek, gdy wektor $\bar{\mathbf{y}}$ podzielono na wektory składowe o różnej długości N_1, \dots, N_M , tak, że

$$\sum_{k=1}^M N_k = \bar{N} \quad (4.20)$$

Do kwantowania tych wektorów składowych wykorzystano zoptymalizowane słowniki liczące odpowiednio

$$L_1 = 2^{b_1 N_1}, \dots, L_M = 2^{b_M N_M} \quad (4.21)$$

wektorów. Cały wektor $\bar{\mathbf{y}}$ jest kwantowany z wykorzystaniem

$$b_1 N_1 + \dots + b_M N_M = b \bar{N} \quad (4.22)$$

bitów. Energia błędu kwantyzacji przypadająca na składową wektora $\bar{\mathbf{y}}_k$ wynosi:

$$\sigma_{e,k}^2 = c(N_k) [\det \mathbf{R}^{N_k}(\bar{\mathbf{y}}_k)]^{1/N_k} 2^{-2b_k}, \quad k = 1, \dots, M \quad (4.23)$$

Energię błędu kwantowania całego wektora $\bar{\mathbf{y}}$ (gdy kwantuje się osobno M jego części) można wyrazić wzorem:

$$\sum_{k=1}^M N_k \sigma_{e,k}^2 = \sum_{k=1}^M N_k c(N_k) [\det \mathbf{R}^{N_k}(\bar{\mathbf{y}}_k)]^{1/N_k} 2^{-2b_k} \quad (4.24)$$

Średnia energia błędu przypadająca na składową wektora $\bar{\mathbf{y}}$, przenoszona przez transformatę odwrotną jako moc błędu kwantowania sygnału, wynosi:

$$\sigma_e^2(\bar{\mathbf{T}}, b, N_1, \dots, N_M) = \sum_{k=1}^M \frac{N_k}{\bar{N}} c(N_k) [\det \mathbf{R}^{N_k}(\bar{\mathbf{y}}_k)]^{1/N_k} 2^{-2b_k} \quad (4.25)$$

Zadanie minimalizacji $\sigma_e^2(\bar{\mathbf{T}}, b, N_1, \dots, N_M)$ względem b_1, \dots, b_M i N_1, \dots, N_M (zadanie rozdziału bitów i długości kwantowanych wektorów) z warunkami (4.20) i (4.22) nie jest dobrze postawione. Wynik takiej optymalizacji jest trywialny - cała dostępna długość \bar{N} i wszystkie zasoby binarne $b\bar{N}$ zostaną przydzielone jednemu wektorowi, transformacja nie będzie wówczas potrzebna, a moc szumu kwantowania osiągnie minimum równe

$$\sigma_e^2(b, \bar{N}) = c(\bar{N}) [\det \mathbf{R}^{\bar{N}}(\bar{\mathbf{x}})]^{1/\bar{N}} 2^{-2b} \quad (4.26)$$

Realizacja takiego kwantowania nie byłaby możliwa przy większych wartościach \bar{N} , ze względu na złożoność obliczeniową.

Aby zadanie minimalizacji $\sigma_e^2(\bar{\mathbf{T}}, b, N_1, \dots, N_M)$ miało sens, należy dodać warunek, ograniczający złożoność obliczeniową. Mając dane długości kwantowanych wektorów N_1, \dots, N_M , można poszukiwać optymalnego rozdziału bitów b_1, \dots, b_M spełniającego warunek (4.22). Posługując się metodą mnożników Lagrange'a otrzymuje się (dla $k = 1, \dots, M$):

$$\sigma_{e,k}^2 = c(N_k) [\det \mathbf{R}^{N_k}(\bar{\mathbf{y}}_k)]^{1/N_k} 2^{-2b_k} = \sigma_e^2(\bar{\mathbf{T}}, b, N_1, \dots, N_M) = \text{const} \quad (4.27)$$

czyli taką samą moc błędu kwantowania każdego wektora składowego. Rozdzielczości poszczególnych kwantyzatorów wynoszą:

$$b_k = \frac{1}{2} \log_2 c(N_k) + \frac{1}{2N_k} \log_2 [\det \mathbf{R}^{N_k}(\bar{\mathbf{y}}_k)] - \sum_{m=1}^M \left[\frac{N_m}{2\bar{N}} \log_2 c(N_m) + \frac{1}{2\bar{N}} \log_2 [\det \mathbf{R}^{N_m}(\bar{\mathbf{y}}_m)] \right] + b, \quad k = 1, \dots, M \quad (4.28)$$

Dodatkowo należy sformułować wymaganie, aby liczby bitów przydzielone poszczególnym kwantyzatorom, czyli $N_1 b_1, \dots, N_M b_M$ były nieujemnymi liczbami całkowitymi.

Ograniczenie na złożoność obliczeniową powinno polegać na narzuceniu takiej samej liczby wektorów we wszystkich słownikach (wzór (4.21)), czyli $L_1 = L_2 = \dots = L_M$, co oznacza, że $b_1 N_1 = b_2 N_2 = \dots = b_M N_M$. Z takim ograniczeniem zadanie minimalizacji błędu nie jest trywialne i było rozwiązywane w pracach [33, 145, 64]. Po rozwiązaniu takiego zadania można obliczyć zysk transformaty, porównując moc błędu kwantowania (4.25) z mocą szumu kwantowania wektora $\bar{\mathbf{x}}$ z taką samą rozdzielczością b i zbliżoną złożonością obliczeniową (podział $\bar{\mathbf{x}}$ na M równych części o długości $N = \bar{N}/M$):

$$G(\bar{\mathbf{T}}) = \frac{\sigma_e^2(b, N)}{\sigma_e^2(\bar{\mathbf{T}}, b, N_1, \dots, N_M)} \quad (4.29)$$

4.2.2 Obliczanie zysku transformat stosowanych w praktyce

Znając macierz transformaty unitarnej $\bar{\mathbf{T}}$, sposób ekstrakcji wektorów składowych z wektora $\bar{\mathbf{y}}$, oraz właściwości statystyczne sygnału wejściowego można określić macierze kowariancji poszczególnych wektorów składowych, a w dalszej kolejności - moc zniekształceń kwantowania (wzory (4.13),(4.18),(4.25)) i zysk transformaty (wzory (4.10),(4.19),(4.29)). Rozpatrzmy sposób postępowania w praktycznych sytuacjach.

1. Jeżeli kwantowanie wektorowe ma się odbywać w podzakresach częstotliwości, to wówczas można wykorzystać dyskretną transformatę Fouriera

$$\bar{\mathbf{T}} = DFT(\bar{N}) = \frac{1}{\sqrt{\bar{N}}} \begin{bmatrix} 1 & 1 & \dots & 1 \\ 1 & e^{j\frac{2\pi}{\bar{N}}} & \dots & e^{j\frac{2\pi}{\bar{N}}(\bar{N}-1)} \\ \dots & \dots & \dots & \dots \\ 1 & e^{j\frac{2\pi}{\bar{N}}(\bar{N}-1)} & \dots & e^{j\frac{2\pi}{\bar{N}}(\bar{N}-1)(\bar{N}-1)} \end{bmatrix} \quad (4.30)$$

lub dyskretną transformatę kosinusoidalną

$$\bar{\mathbf{T}} = DCT(\bar{N}) = \frac{1}{\sqrt{\bar{N}}} \begin{bmatrix} 1 & \sqrt{2} \cos[\frac{\pi}{2\bar{N}}(1)(1)] & \cdots & \sqrt{2} \cos[\frac{\pi}{2\bar{N}}(1)(\bar{N}-1)] \\ 1 & \sqrt{2} \cos[\frac{\pi}{2\bar{N}}(3)(1)] & \cdots & \sqrt{2} \cos[\frac{\pi}{2\bar{N}}(3)(\bar{N}-1)] \\ \cdots & \cdots & \cdots & \cdots \\ 1 & \sqrt{2} \cos[\frac{\pi}{2\bar{N}}(2\bar{N}-1)(1)] & \cdots & \sqrt{2} \cos[\frac{\pi}{2\bar{N}}(2\bar{N}-1)(\bar{N}-1)] \end{bmatrix} \quad (4.31)$$

Szerokości M podzakresów częstotliwości mogą być równe (kwantyzatory wektorowe operują na wektorach o jednakowej długości $N = \bar{N}/M$) lub nierówne - proporcjonalne do N_1, \dots, N_M . W przypadku DFT należy oczywiście uwzględnić fakt, że wektor $\bar{\mathbf{y}}$ zawiera składowe parami sprzężone (poza składowymi o indeksie 0 i $\bar{N}/2$).

- Jeżeli każdy z N - wymiarowych wektorów \mathbf{x} jest transformowany osobno, z wykorzystaniem transformaty unitarnej \mathbf{T} , a potrzebne jest jednoczesne przetwarzanie M kolejnych wektorów, to macierz $\bar{\mathbf{T}}$ można utworzyć następująco:

$$\bar{\mathbf{T}} = \bar{\mathbf{T}}(M) = \begin{bmatrix} \mathbf{T} & 0 & \cdots & 0 \\ \cdots & \cdots & \cdots & \cdots \\ 0 & \cdots & 0 & \mathbf{T} \end{bmatrix} \quad (4.32)$$

gdzie \mathbf{T} jest macierzą o wymiarach $N \times N$ (w przypadku transformat "z nakładaniem", tzw. *lapped transforms* [130], liczba kolumn może być kilkakrotnie większa od liczby wierszy).

Wykorzystanie transformaty $\bar{\mathbf{T}}(M)$, a następnie niezależna kwantyzacja wektorowa M otrzymanych N - wymiarowych wektorów nie ma większego sensu - jak już wykazano (równanie (4.7)), nie prowadzi to do poprawy SNR .

- Podział przetransformowanego wektora $\bar{\mathbf{y}}$ na M części nie musi się odbywać na zasadzie konkatencji (tzn. wektory $\bar{\mathbf{y}}_k, k = 1, \dots, M$ są utworzone z kolejnych składowych $\bar{\mathbf{y}}$). Można również przetwarzać wektory polifazowe, co oznacza zastosowanie transformacji zmieniającej kolejność składowych:

$$\bar{\mathbf{P}}(M) = \begin{bmatrix} \mathbf{P}_{0,0} & \cdots & \mathbf{P}_{0,N-1} \\ \cdots & \cdots & \cdots \\ \mathbf{P}_{M-1,0} & \cdots & \mathbf{P}_{M-1,N-1} \end{bmatrix} \quad (4.33)$$

gdzie $\mathbf{P}_{i,j}$ mają wymiary $N \times M$ i zawierają po jednej składowej niezerowej, równej 1. Np. w celu ekstrakcji $M = 2$ wektorów polifazowych o wymiarze $N = 3$ bezpośrednio z $\bar{N} = 6$ -wymiarowego wektora $\bar{\mathbf{x}}$ należy zastosować transformację opisaną wzorem:

$$\bar{\mathbf{y}} = \begin{bmatrix} x_0 \\ x_2 \\ x_4 \\ x_1 \\ x_3 \\ x_5 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_0 \\ x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \end{bmatrix} \quad (4.34)$$

Oczywiście wykonywanie takiej operacji na wektorze $\bar{\mathbf{x}}$ nie ma sensu, gdyż składowe wektorów polifazowych są ze sobą słabiej skorelowane niż kolejne próbki sygnału. Podobnie nie ma sensu osobne transformowanie takich wektorów polifazowych powstałych z $\bar{\mathbf{x}}$.

4. W przeciwieństwie do powyższego, sensownym i często stosowanym rozwiązaniem jest transformacja M kolejnych N -wymiarowych wektorów (z wykorzystaniem tej samej transformaty unitarnej \mathbf{T}), a następnie ekstrakcja N wektorów polifazowych o wymiarze M . Odpowiada to kwantyzacji wektorowej sygnałów subpasmowych (każdy wektor polifazowy zawiera składowe o tym samym indeksie, pochodzące z kolejno transformowanych wektorów $\mathbf{T}\mathbf{x}$). Odpowiednia transformata wynosi:

$$\bar{\mathbf{T}} = \bar{\mathbf{P}}(N) \bar{\mathbf{T}}(M) \quad (4.35)$$

gdzie $\bar{\mathbf{T}}(M)$ - transformacja M wektorów N -wymiarowych, $\bar{\mathbf{P}}(N)$ - ekstrakcja N wektorów M -wymiarowych, tworzących sygnały subpasmowe.

5. Metoda zaproponowana przez Li i Zhanga [119, 118] może być opisana macierzą transformaty

$$\bar{\mathbf{T}} = \bar{\mathbf{P}}(M) \bar{\mathbf{T}}(N) \bar{\mathbf{P}}(N) = \bar{\mathbf{P}}^t(N) \bar{\mathbf{T}}(N) \bar{\mathbf{P}}(N) \quad (4.36)$$

Polega ona na ekstrakcji N wektorów polifazowych (M -wymiarowych), niezależnej transformacji każdego z tych wektorów, a następnie ekstrakcji M wektorów N -wymiarowych.

4.2.3 Transformaty "z nakładaniem"

W celu poprawienia rozdzielczości w dziedzinie częstotliwości, Malvar zaproponował transformaty "z nakładaniem" (ang. *lapped transforms*), opisane macierzami prostokątnymi [131, 129, 130].

Macierz transformaty ma wymiar $m \times lm$:

$$\mathbf{T} = \mathbf{T}(m, l) = [\mathbf{T}_0 \cdots \mathbf{T}_{l-1}] = \begin{bmatrix} h_0(0) & \cdots & h_0(lm-1) \\ \cdots & \cdots & \cdots \\ h_{m-1}(0) & \cdots & h_{m-1}(lm-1) \end{bmatrix} \quad (4.37)$$

gdzie k -ty wiersz $h_k(n)$, $n = 0, \dots, lm-1$ może być traktowany jako odpowiedź impulsowa k -tego filtru analizy częstotliwościowej:

$$h_k(n) = v(n) \sqrt{\frac{2}{m}} \cos\left[\left(n + \frac{m+1}{2}\right)\left(k + \frac{1}{2}\right)\frac{\pi}{m}\right], \quad n = 0, \dots, lm-1 \quad (4.38)$$

Okno $v(n)$, $n = 0, \dots, lm-1$ jest dla $l = 2$ określone wzorem

$$v(n) = -\sin\left[\left(n + \frac{1}{2}\right)\frac{\pi}{2m}\right]$$

ma wymiar nieskończony. Jeśli kwantuje się M wektorów m -wymiarowych jednocześnie (np. kwantyzacja wektorowa w podpasmach), to można rozpatrywać jedynie mM wierszy tej macierzy. Macierz $\bar{\mathbf{T}}_m$ składa się z l różnych macierzy kwadratowych o wymiarze $m \times m$: $\mathbf{T}_0 \mathbf{T}_1 \cdots \mathbf{T}_{l-1}$. Można wykazać, że macierz $\bar{\mathbf{T}}_m$ jest unitarna i zachowuje energię transformowanego sygnału.

Aby wyodrębnić k -tą składową kolejnych m -wymiarowych wektorów transformaty (np. sygnał k -tego podpasma częstotliwości, $k = 0, \dots, m - 1$) można wykorzystać macierz ekstrakcji $\bar{\mathbf{P}}(k, m)$, np.

$$\bar{\mathbf{P}}(k = 1, m = 4) = \begin{bmatrix} \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \cdots & 0 & 1 & 0 & 0 & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ \cdots & \cdots & \cdots & \cdots & \cdots & 0 & 1 & 0 & 0 & 0 & \cdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \end{bmatrix} \quad (4.41)$$

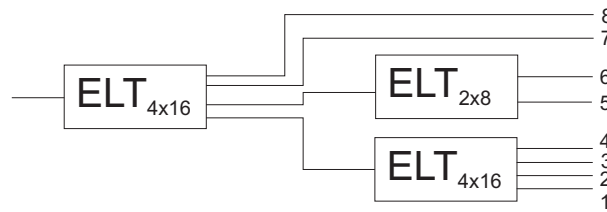
Operację transformacji kolejnych m -wymiarowych wektorów oraz ekstrakcji k -tej składowej można opisać macierzą (dla przykładu podstawiono $m = 4$):

$$\bar{\mathbf{P}}(k, m) \bar{\mathbf{T}}_m = \begin{bmatrix} \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \cdots & 0 & h_k(0) & h_k(1) & h_k(2) & h_k(3) & h_k(4) & \cdots & \cdots & \cdots & \cdots \\ \cdots & \cdots & \cdots & \cdots & \cdots & 0 & h_k(0) & h_k(1) & h_k(2) & \cdots & \cdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \end{bmatrix} \quad (4.42)$$

Operację tę traktuje się jako filtrację sygnału wejściowego w k -tym (z m dostępnych) filtrze pasmowym o odpowiedzi impulsowej $h_k(0), \dots, h_k(lm - 1)$ wraz z obniżeniem (m -krotnym) częstotliwości próbkowania. Zapis tej operacji w formie macierzowej (równanie (4.42)) jest potrzebny do obliczenia macierzy kowariancji wektorów utworzonych z sygnałów subpasmych i obliczenia mocy błędu kwantowania tych wektorów.

Transformaty z nakładaniem wprowadzają większe opóźnienie algorytmiczne² niż transformaty opisane macierzą kwadratową. Dla klasycznej transformaty, opisanej macierzą $m \times m$, wynosi ono m próbek sygnału wejściowego. W przypadku transformaty opisanej macierzą $m \times lm$ opóźnienie wynosi ml próbek. Jeśli chcemy jednocześnie przetwarzać (kwantyzacja wektorowa) M próbek sygnału subpasmych, to opóźnienie wzrasta do $Mm + (l - 1)m$ próbek sygnału wejściowego. Sposób na zmniejszenie tego opóźnienia podano w pracach [147, 58]. Nałożenie się okien czasowych, wynoszące $r = (l - 1)m$ próbek i wynikające stąd opóźnienie algorytmiczne, można regulować przez odpowiednie kształtowanie wierszy macierzy $\mathbf{T}(m, l)$. Wiersze powinny się zerować na brzegach z zachowaniem warunku idealnej rekonstrukcji. Np. dla MLT ($l = 2$) efektywne nałożenie się okien (r próbek) można regulować w zakresie od 0 do m . Przypadek $r = 0$ odpowiada klasycznej DCT, natomiast dla $r = m$ otrzymuje się MLT. Opóźnienie algorytmiczne wynosi $m + r$, jednak nie warto go zanadto zmniejszać ze względu na efekty brzegowe i pogorszenie rozdzielczości w dziedzinie częstotliwości. Ostatecznie w pracach [147, 58] zaproponowano $r = m/2$ jako rozsądny kompromis między rozdzielczością w dziedzinie częstotliwości a opóźnieniem algorytmicznym.

²nie uwzględniające czasu transmisji oraz szybkości obliczeń



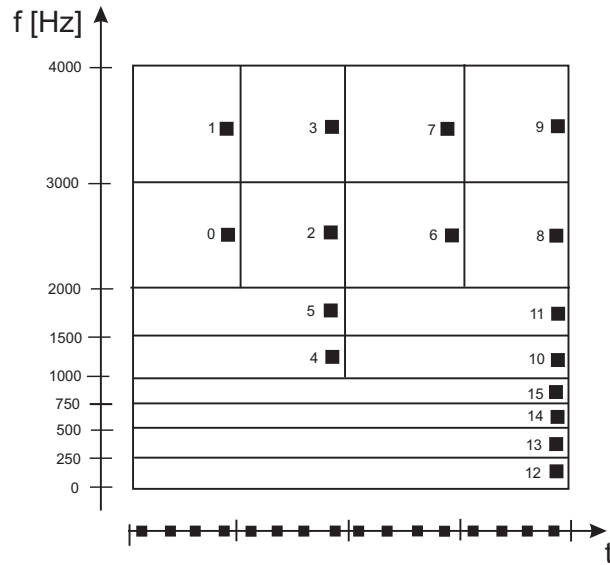
Rysunek 4.2: Zestaw filtrów o strukturze hierarchicznej

4.2.4 Zestaw filtrów o zmiennej rozdzielczości

Ze względu na właściwości ludzkiego ucha, korzystne jest zachowanie większej rozdzielczości w zakresie małych częstotliwości a mniejszej rozdzielczości w zakresie większych częstotliwości [209], [107]. Wymagania te spełnia (dla sygnału telefonicznego) zestaw filtrów o strukturze hierarchicznej [144]. Układ filtrów został zrealizowany drogą wielokrotnego wykorzystania transformaty ELT [130] - patrz rys.4.2. Pasma telefoniczne $[0, 4 \text{ kHz}]$ jest dzielone na 4 podpasma o szerokości 1 kHz za pomocą transformaty ELT opisanej macierzą stopnia 4×16 , co jest równoważne zastosowaniu 4 filtrów o odpowiedziach impulsowych zapisanych we wierszach macierzy i czterokrotnym obniżeniu częstotliwości próbkowania. Pierwsze podpasmo $[0, 1 \text{ kHz}]$ dzieli się na 4 części w identyczny sposób jak poprzednio, natomiast drugie podpasmo $[1, 2 \text{ kHz}]$ dzieli się na 2 części z wykorzystaniem transformaty ELT opisanej macierzą stopnia 2×8 co jest równoważne zastosowaniu 2 filtrów o odpowiedziach impulsowych zapisanych we wierszach macierzy i dwukrotnym obniżeniu częstotliwości próbkowania. I tak, z 16 próbek sygnału wejściowego otrzymuje się po 1 próbce sygnału wyjściowego pierwszych 4 podpasm ($4 \cdot 4 = 16$ -krotne obniżenie częstotliwości próbkowania), po 2 próbki sygnału wyjściowego podpasm 5-6 ($4 \cdot 2 = 8$ -krotne obniżenie częstotliwości próbkowania) i po 4 próbki sygnału wyjściowego podpasm 7-8 (4-krotne obniżenie częstotliwości próbkowania). Rozkład czasowo - częstotliwościowy tych 16 próbek pokazano na rys 4.3.

Bezpośrednia kwantyzacja wektorowa 16 próbek z rys 4.3 nie ma sensu (nie prowadzi do poprawy SNR), gdyż układ filtrów zachowuje się jak transformata unitarna i można pokazać, że zachowuje energię sygnału. Zysk można osiągnąć, kwantyzując wektorowo sygnały w poszczególnych podpasmach. Załóżmy, że wszystkie kwantyzatory wektorowe operują na wektorach N - wymiarowych. Wówczas należy przetwarzać bloki liczące $16N$ próbek (ze względu na 16 - krotne obniżenie częstotliwości próbkowania w podpasmach 1-4). W konsekwencji otrzymuje się po 1 wektorze (N - wymiarowym) w podpasmach 1-4, po 2 wektory w podpasmach 5-6 i po 4 wektory w podpasmach 7-8. Zgodnie z założeniem, zapewniona jest duża rozdzielczość w dziedzinie częstotliwości (i mała - w dziedzinie czasu) w podpasmach 1-4. W podpasmach 7-8 jest odwrotnie. W sumie z bloku $16N$ próbek sygnału wejściowego tworzy się $M = 16 N$ - wymiarowych wektorów sygnałów podpasmowych.

Do kwantyzacji wektorowej należy wykorzystać 8 różnych kwantyzatorów wektorowych - po jednym na każde podpasmo. Aby obliczyć zysk oferowany przez zestaw filtrów z omówionymi kwantyzatorami wektorowymi należy skorzystać ze wzorów (4.18) i (4.19), co wymaga obliczenia macierzy kowariancji wektorów utworzonych z sygnałów podpasmowych $\mathbf{R}^N(\bar{\mathbf{y}}_k)$.



Rysunek 4.3: Rozkład czasowo-częstotliwościowy 16 próbek na wyjściach zestawu filtrów o strukturze hierarchicznej

I tak, N - wymiarowe wektory utworzone z sygnałów podpasm 7 i 8 wyrażają się wzorami

$$\begin{aligned}\bar{\mathbf{y}}_7 &= \bar{\mathbf{P}}(3,4)\bar{\mathbf{T}}_4 \bar{\mathbf{x}} \\ \bar{\mathbf{y}}_8 &= \bar{\mathbf{P}}(4,4)\bar{\mathbf{T}}_4 \bar{\mathbf{x}}\end{aligned}\quad (4.43)$$

Macierz ekstrakcji k -tej składowej jest zdefiniowana wzorem (4.41) z tym, że składowe wektora są ponumerowane od 1 wzwyż. Odpowiednie macierze kowariancji otrzymuje się ze wzorów:

$$\begin{aligned}\mathbf{R}^N(\bar{\mathbf{y}}_7) &= \bar{\mathbf{P}}(3,4)\bar{\mathbf{T}}_4 E[\bar{\mathbf{x}}\bar{\mathbf{x}}^t] [\bar{\mathbf{P}}(3,4)\bar{\mathbf{T}}_4]^t \\ \mathbf{R}^N(\bar{\mathbf{y}}_8) &= \bar{\mathbf{P}}(4,4)\bar{\mathbf{T}}_4 E[\bar{\mathbf{x}}\bar{\mathbf{x}}^t] [\bar{\mathbf{P}}(4,4)\bar{\mathbf{T}}_4]^t\end{aligned}\quad (4.44)$$

W podpasmach $k = 1 \dots 4$ przetwarzanie jest dwuetapowe

$$\bar{\mathbf{y}}_k = \bar{\mathbf{P}}(k,4)\bar{\mathbf{T}}_4\bar{\mathbf{P}}(1,4)\bar{\mathbf{T}}_4 \bar{\mathbf{x}}$$

skąd

$$\mathbf{R}^N(\bar{\mathbf{y}}_k) = \bar{\mathbf{P}}(k,4)\bar{\mathbf{T}}_4\bar{\mathbf{P}}(1,4)\bar{\mathbf{T}}_4 E[\bar{\mathbf{x}}\bar{\mathbf{x}}^t] [\bar{\mathbf{P}}(k,4)\bar{\mathbf{T}}_4\bar{\mathbf{P}}(1,4)\bar{\mathbf{T}}_4]^t\quad (4.45)$$

Z kolei w podpasmach 5 i 6, w drugim etapie, stosowana jest macierz ELT stopnia 2×8 :

$$\begin{aligned}\bar{\mathbf{y}}_5 &= \bar{\mathbf{P}}(1,2)\bar{\mathbf{T}}_2\bar{\mathbf{P}}(2,4)\bar{\mathbf{T}}_4 \bar{\mathbf{x}} \\ \bar{\mathbf{y}}_6 &= \bar{\mathbf{P}}(2,2)\bar{\mathbf{T}}_2\bar{\mathbf{P}}(2,4)\bar{\mathbf{T}}_4 \bar{\mathbf{x}}\end{aligned}\quad (4.46)$$

Stąd macierze kowariancji

$$\mathbf{R}^N(\bar{\mathbf{y}}_5) = \bar{\mathbf{P}}(1,2)\bar{\mathbf{T}}_2\bar{\mathbf{P}}(2,4)\bar{\mathbf{T}}_4 E[\bar{\mathbf{x}}\bar{\mathbf{x}}^t] [\bar{\mathbf{P}}(1,2)\bar{\mathbf{T}}_2\bar{\mathbf{P}}(2,4)\bar{\mathbf{T}}_4]^t$$

$$\mathbf{R}^N(\bar{\mathbf{y}}_6) = \bar{\mathbf{P}}(2, 2)\bar{\mathbf{T}}_2\bar{\mathbf{P}}(2, 4)\bar{\mathbf{T}}_4 E[\bar{\mathbf{x}}\bar{\mathbf{x}}^t] [\bar{\mathbf{P}}(2, 2)\bar{\mathbf{T}}_2\bar{\mathbf{P}}(2, 4)\bar{\mathbf{T}}_4]^t \quad (4.47)$$

Ze wzorów (4.44), (4.45) i (4.47) wynika stopień macierzy kowariancji $E[\bar{\mathbf{x}}\bar{\mathbf{x}}^t]$ - wynosi on odpowiednio $4N + 12$, $16N + 60$ i $8N + 28$. Filtrując blok $16N$ próbek sygnału wejściowego z wykorzystaniem zestawu filtrów o strukturze hierarchicznej $\bar{\mathbf{T}}_{ELT}$ otrzymuje się $M = 16$ N - wymiarowych wektorów podlegających kwantowaniu wektorowemu. Stosując optymalny rozdział bitów między 8 kwantyzatorów operujących w poszczególnych podpasmach, można posłużyć się wzorem (4.18) dla obliczenia mocy błędu kwantyzacji:

$$\sigma_e^2(\bar{\mathbf{T}}_{ELT}, b, N) = c(N) \left[\prod_{k=1}^4 \det \mathbf{R}^N(\bar{\mathbf{y}}_k) \left[\prod_{k=5}^6 \det \mathbf{R}^N(\bar{\mathbf{y}}_k) \right]^2 \left[\prod_{k=7}^8 \det \mathbf{R}^N(\bar{\mathbf{y}}_k) \right]^4 \right]^{1/16N} 2^{-2b}$$

który można przepisać w postaci

$$\sigma_e^2(\bar{\mathbf{T}}_{ELT}, b, N) = c(N) \left[\prod_{k=1}^4 \det \mathbf{R}^N(\bar{\mathbf{y}}_k) \right]^{1/16N} \left[\prod_{k=5}^6 \det \mathbf{R}^N(\bar{\mathbf{y}}_k) \right]^{1/8N} \left[\prod_{k=7}^8 \det \mathbf{R}^N(\bar{\mathbf{y}}_k) \right]^{1/4N} 2^{-2b} \quad (4.48)$$

Aby określić zysk ze stosowania kwantowania wektorowego sygnałów podpasmych, należy porównać otrzymaną moc błędu kwantyzacji z mocą błędu kwantowania wektorowego N -wymiarowych wektorów sygnału wejściowego (wzór (4.10)):

$$G(\bar{\mathbf{T}}_{ELT}) = \frac{\sigma_e^2(b, N)}{\sigma_e^2(\bar{\mathbf{T}}_{ELT}, b, N)}$$

Ostatecznie otrzymuje się

$$G(\bar{\mathbf{T}}_{ELT}) = \frac{[\det \mathbf{R}^N(\mathbf{x})]^{1/N}}{\left[\prod_{k=1}^4 \det \mathbf{R}^N(\bar{\mathbf{y}}_k) \right]^{1/16N} \left[\prod_{k=5}^6 \det \mathbf{R}^N(\bar{\mathbf{y}}_k) \right]^{1/8N} \left[\prod_{k=7}^8 \det \mathbf{R}^N(\bar{\mathbf{y}}_k) \right]^{1/4N}} \quad (4.49)$$

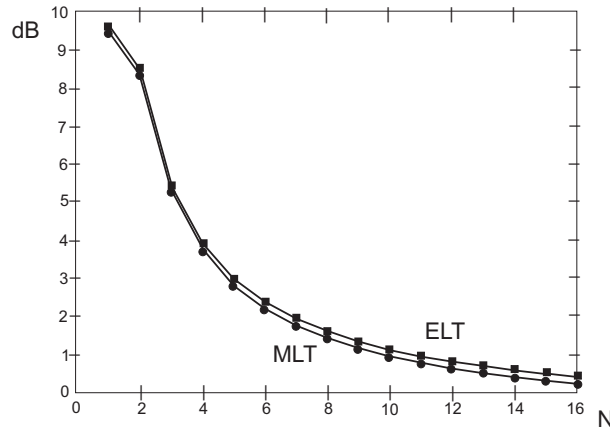
Przeprowadzono przykładowe obliczenia dla procesu AR(2), otrzymanego przez filtrację nieskorelowanego sygnału gaussowskiego z wykorzystaniem filtru:

$$\frac{1}{A(z)} = \frac{1}{(1 - \gamma e^{j\theta} z^{-1})(1 - \gamma e^{-j\theta} z^{-1})} = \frac{1}{1 + a_1 z^{-1} + a_2 z^{-2}} \quad (4.50)$$

podstawiając $\gamma = 0.96$ i $\theta = \pi/4$. Wyniki przedstawiono na rys. 4.4.

Oczywiście największy zysk przetwarzania w podpasmach jest widoczny dla kwantowania skalarne ($N = 1$). Im większy jest wymiar wektora, tym lepsze są wyniki bezpośredniego kwantowania wektorowego sygnału wejściowego i tym mniej widoczny jest zysk wynikający z filtracji. Należy pamiętać, że kwantyzator wektorowy jest koderem asymptotycznie optymalnym i dla $N \rightarrow \infty$ osiąga największy SNR - w tej sytuacji transformata niczego już nie wnosi.

Porównanie transformat MLT i ELT wykazuje nieznaczną przewagę tej ostatniej.



Rysunek 4.4: Zysk [dB] zestawu filtrów o strukturze hierarchicznej (opartej na MLT i ELT) w funkcji wymiaru kwantowanych wektorów N

4.2.5 Transformata optymalna

Zysk transformaty określony jest wzorem (4.19) (dla kwantyzatorów wektorowych przetwarzających wektory o tej samej długości) i wzorem (4.29) (dla kwantyzatorów wektorowych operujących na wektorach o różnej długości). Jeżeli założymy, że dokonano optymalnego rozdziału bitów między M kwantyzatorów wektorowych, to wówczas zysk dla danego sygnału wejściowego (zbioru wektorów $\bar{\mathbf{x}}$) zależy od zastosowanej transformaty $\bar{\mathbf{T}}$. Dla uproszczenia założymy, że kwantyzatory wektorowe operują na wektorach o stałej długości $N = \bar{N}/M$ (analizę można rozszerzyć na przypadek ogólny - wektorów o różnej długości). Dążąc do maksymalizacji zysku transformaty należy, zgodnie ze wzorem (4.18), dążyć do minimalizacji iloczynu wyznaczników macierzy kowariancji kwantowanych wektorów:

$$\min \prod_{k=1}^M \det \mathbf{R}^N(\bar{\mathbf{y}}_k) \quad (4.51)$$

Macierze $\mathbf{R}^N(\bar{\mathbf{y}}_k)$ są rozłożone wzdłuż głównej przekątnej macierzy kowariancji wektorów $\bar{\mathbf{y}}$:

$$\mathbf{R}^{\bar{N}}(\bar{\mathbf{y}}) = E\{\bar{\mathbf{y}}\bar{\mathbf{y}}^t\} = \begin{bmatrix} E\{\bar{\mathbf{y}}_1\bar{\mathbf{y}}_1^t\} & \cdots & E\{\bar{\mathbf{y}}_1\bar{\mathbf{y}}_M^t\} \\ \cdots & \cdots & \cdots \\ E\{\bar{\mathbf{y}}_M\bar{\mathbf{y}}_1^t\} & \cdots & E\{\bar{\mathbf{y}}_M\bar{\mathbf{y}}_M^t\} \end{bmatrix} \quad (4.52)$$

Macierze kowariancji wektorów $\bar{\mathbf{x}}$ i $\bar{\mathbf{y}} = \bar{\mathbf{T}}\bar{\mathbf{x}}$ są związane zależnością³

$$\mathbf{R}^{\bar{N}}(\bar{\mathbf{y}}) = \bar{\mathbf{T}} \mathbf{R}^{\bar{N}}(\bar{\mathbf{x}}) \bar{\mathbf{T}}^t \quad (4.53)$$

³Wzór (4.53) jest słuszny zarówno dla macierzy kowariancji, jak i macierzy korelacji. Niezależnie od tego, ze względu na zerową wartość średnią wektorów, obie macierze są równe.

Zgodnie ze wzorem (4.5) wyznaczniki macierzy $\mathbf{R}^{\bar{N}}(\bar{\mathbf{y}})$ i $\mathbf{R}^{\bar{N}}(\bar{\mathbf{x}})$ są równe. Wyznacznik $\mathbf{R}^{\bar{N}}(\bar{\mathbf{y}})$ można zapisać w postaci iloczynu wyrażenia (4.51) i wyznacznika pewnej macierzy \mathbf{R}' :

$$\det \mathbf{R}^{\bar{N}}(\bar{\mathbf{y}}) = (\det \mathbf{R}') \prod_{k=1}^M \det \mathbf{R}^N(\bar{\mathbf{y}}_k) \quad (4.54)$$

Ponieważ $\det \mathbf{R}^{\bar{N}}(\bar{\mathbf{y}}) = \det \mathbf{R}^{\bar{N}}(\bar{\mathbf{x}})$ zależy jedynie od przetwarzanego sygnału, zatem minimalizacja wyrażenia (4.51) oznacza maksymalizację $\det \mathbf{R}'$.

Należy teraz znaleźć wyrażenie opisujące macierz \mathbf{R}' . Zauważmy, że macierze kowariancji $\mathbf{R}^N(\bar{\mathbf{y}}_k) = E\{\bar{\mathbf{y}}_k \bar{\mathbf{y}}_k^t\}$ są symetryczne i dodatnio półokreślone - założmy, że są one określone dodatnio. Można je przedstawić w następującej postaci:

$$E\{\bar{\mathbf{y}}_k \bar{\mathbf{y}}_k^t\} = \mathbf{A}_k \mathbf{D}_k^2 \mathbf{A}_k^t = (\mathbf{A}_k \mathbf{D}_k)(\mathbf{A}_k \mathbf{D}_k)^t = \mathbf{B}_k \mathbf{B}_k^t \quad (4.55)$$

gdzie \mathbf{D}_k jest macierzą diagonalną zawierającą pierwiastki wartości własnych (dodatnich) macierzy kowariancji $E\{\bar{\mathbf{y}}_k \bar{\mathbf{y}}_k^t\}$. W związku z tym można zapisać:

$$\mathbf{R}^{\bar{N}}(\bar{\mathbf{y}}) = \begin{bmatrix} \mathbf{B}_1 & 0 & \cdots & 0 \\ 0 & \cdots & \cdots & \cdots \\ \cdots & \cdots & \cdots & \cdots \\ \cdots & \cdots & \cdots & 0 \\ 0 & \cdots & 0 & \mathbf{B}_M \end{bmatrix} \begin{bmatrix} \mathbf{I} & \mathbf{r}'_{1,2} & \cdots & \mathbf{r}'_{1,M} \\ \mathbf{r}'_{2,1} & \mathbf{I} & \cdots & \cdots \\ \cdots & \cdots & \cdots & \cdots \\ \mathbf{r}'_{M,1} & \cdots & \cdots & \mathbf{I} \end{bmatrix} \begin{bmatrix} \mathbf{B}_1^t & 0 & \cdots & 0 \\ 0 & \cdots & \cdots & \cdots \\ \cdots & \cdots & \cdots & \cdots \\ \cdots & \cdots & \cdots & 0 \\ 0 & \cdots & 0 & \mathbf{B}_M^t \end{bmatrix} \quad (4.56)$$

gdzie dla $k \neq l$

$$\mathbf{r}'_{k,l} = \mathbf{B}_k^{-1} E\{\bar{\mathbf{y}}_k \bar{\mathbf{y}}_l^t\} \mathbf{B}_l^{-t} \quad (4.57)$$

gdzie $\mathbf{B}_l^{-t} = (\mathbf{B}_l^t)^{-1}$.

Nietrudno zauważyć, że macierz zawierająca podmacierze $\mathbf{r}'_{k,l}$ jest poszukiwaną macierzą \mathbf{R}' , tzn.

$$\mathbf{R}^{\bar{N}}(\bar{\mathbf{y}}) = \mathbf{B} \mathbf{R}' \mathbf{B}^t \quad (4.58)$$

gdyż

$$\det \mathbf{R}^{\bar{N}}(\bar{\mathbf{y}}) = \det^2 \mathbf{B} \det \mathbf{R}' = \det \mathbf{R}' \prod_{k=1}^M \det^2 \mathbf{B}_k \quad (4.59)$$

a (na podstawie (4.55)) $\det^2 \mathbf{B}_k = \det \mathbf{R}^N(\bar{\mathbf{y}}_k)$, co spełnia równanie (4.54).

Macierz \mathbf{R}' , której wyznacznik należy maksymalizować, jest macierzą symetryczną:

$$\mathbf{r}'_{k,l}{}^t = (\mathbf{B}_k^{-1} E\{\bar{\mathbf{y}}_k \bar{\mathbf{y}}_l^t\} \mathbf{B}_l^{-t})^t = \mathbf{B}_l^{-1} E\{\bar{\mathbf{y}}_l \bar{\mathbf{y}}_k^t\} \mathbf{B}_k^{-t} = \mathbf{r}'_{l,k} \quad (4.60)$$

i dodatnio określoną, gdyż dla każdego wektora \mathbf{a}

$$\mathbf{a}^t \mathbf{R}^{\bar{N}}(\bar{\mathbf{y}}) \mathbf{a} = (\mathbf{B}^t \mathbf{a})^t \mathbf{R}' (\mathbf{B}^t \mathbf{a}) > 0 \quad (4.61)$$

Niech $\lambda_1 \cdots \lambda_{NM}$ będą wartościami własnymi macierzy \mathbf{R}' - są one rzeczywiste i dodatnie, ich iloczyn jest równy wyznacznikowi macierzy a ich suma jest równa śladowi macierzy, czyli NM . W związku z tym spełniona jest nierówność:

$$\det \mathbf{R}' = \prod_{k=1}^{NM} \lambda_k \leq \left(\frac{1}{NM} \sum_{k=1}^{NM} \lambda_k \right)^{NM} = 1 \quad (4.62)$$

Maksimum wartości wyznacznika występuje dla macierzy diagonalnej, której wszystkie wartości własne są równe 1. Oznacza to, że dla $k \neq l$ $\mathbf{r}'_{k,l} = 0$, a więc

$$E\{\bar{\mathbf{y}}_k \bar{\mathbf{y}}_l^t\} = \mathbf{B}_k \mathbf{r}'_{k,l} \mathbf{B}_l^t = 0 \quad (4.63)$$

Ze wzoru (4.52) wynika, że macierz kowariancji wektorów $\bar{\mathbf{y}}$, czyli $\mathbf{R}^{\bar{N}}(\bar{\mathbf{y}})$ składa się z położonych wzdłuż przekątnej macierzy $\mathbf{R}^N(\bar{\mathbf{y}}_k)$ (są to macierze kowariancji poszczególnych części wektora $\bar{\mathbf{y}}$) oraz z elementów zerowych. Oznacza to, że **transformata optymalna $\bar{\mathbf{T}}_{opt}$ dekoreluje wektory przetwarzane w różnych kwantyzatorach wektorowych, z możliwością zachowania korelacji wewnętrznych $E\{\bar{\mathbf{y}}_k \bar{\mathbf{y}}_k^t\}$.**

Na podstawie (4.54), dla transformaty optymalnej spełniona jest równość:

$$\prod_{k=1}^M \det \mathbf{R}^N(\bar{\mathbf{y}}_k) = \det \mathbf{R}^{\bar{N}}(\bar{\mathbf{y}}) = \det \mathbf{R}^{\bar{N}}(\bar{\mathbf{x}}) \quad (4.64)$$

Moc błędu kwantyzacji wyraża się wzorem (4.18), który można przepisać w postaci:

$$\sigma_e^2(\bar{\mathbf{T}}_{opt}, b, N) = c(N) [\det \mathbf{R}^{\bar{N}}(\bar{\mathbf{x}})]^{1/NM} 2^{-2b} \quad (4.65)$$

Podobnie wzór na zysk transformaty (4.19) można zastosować do transformaty optymalnej:

$$G(\bar{\mathbf{T}}_{opt}) = \frac{[\det \mathbf{R}^N(\mathbf{x})]^{1/N}}{[\det \mathbf{R}^{\bar{N}}(\bar{\mathbf{x}})]^{1/NM}} = \frac{[\det \mathbf{R}^N(\mathbf{x})]^{1/N}}{[\det \mathbf{R}^{\bar{N}}(\bar{\mathbf{x}})]^{1/\bar{N}}} \leq \frac{c(N) [\det \mathbf{R}^N(\mathbf{x})]^{1/N}}{c(\bar{N}) [\det \mathbf{R}^{\bar{N}}(\bar{\mathbf{x}})]^{1/\bar{N}}} \quad (4.66)$$

Nasuują się następujące spostrzeżenia:

- Przykładem transformaty optymalnej jest transformata Karhunen-Loevego (KLT), dekorelująca wszystkie składowe wektora $\bar{\mathbf{y}}$. Wówczas macierz kowariancji przetworzonych wektorów $\bar{\mathbf{y}}$ jest macierzą diagonalną, zawierającą na przekątnej wariancje poszczególnych składowych;
- Dla $N = const$ zysk transformaty optymalnej $G(\bar{\mathbf{T}}_{opt})$ rośnie w funkcji M - zależnie od przetwarzanego sygnału;
- Dla $M = const$ zysk transformaty optymalnej $G(\bar{\mathbf{T}}_{opt})$ maleje w funkcji N . Dla $N \rightarrow \infty$ $G(\bar{\mathbf{T}}_{opt}) \rightarrow 1$;
- Dla $N = 1$ (kwantyzacja skalarna współczynników transformaty) jedyną transformatą optymalną jest KLT. Zysk takiej transformaty wynosi:

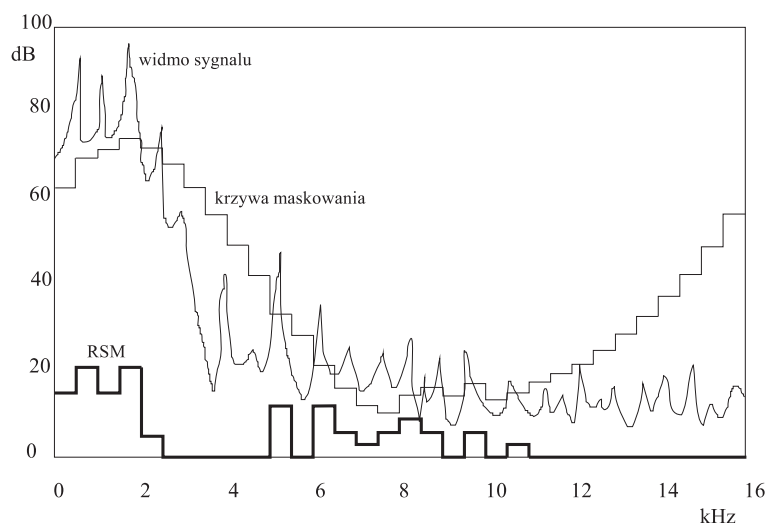
$$G(\bar{\mathbf{T}}_{opt}) = \frac{\sigma_x^2}{[\det \mathbf{R}^M(\bar{\mathbf{x}})]^{1/M}} = \frac{(\sum_{k=1}^M \sigma_k^2)/M}{(\prod_{k=1}^M \sigma_k^2)^{1/M}} \quad (4.67)$$

gdzie σ_x^2 - wariancja sygnału wejściowego, $\sigma_1^2 \dots \sigma_M^2$ - wariancje składowych przetworzonego M - wymiarowego wektora;

- Aby przekonać się, czy bardziej korzystna jest bezpośrednia kwantyzacja wektorowa N - wymiarowych wektorów \mathbf{x} , czy też optymalna transformacja (KLT) N - wymiarowych wektorów, a następnie kwantyzacja skalarna poszczególnych składowych, należy porównać $\sigma_e^2(b, N)$ z $\sigma_e^2(\bar{\mathbf{T}}_{opt}, b, 1)$ tworząc stosunek:

$$\frac{\sigma_e^2(\bar{\mathbf{T}}_{opt}, b, 1)}{\sigma_e^2(b, N)} = \frac{c(1)}{c(N)} \quad (4.68)$$

Stosunek ten jest większy od jedności dla $N > 1$. Kwantyzacja wektorowa jest bardziej efektywną techniką kodowania sygnału niż transformacja i kwantyzacja skalarna. W praktyce przewaga kwantyzacji wektorowej jest jeszcze większa, gdyż błąd kwantyzacji dla KLT z kwantyzatorami skalarnymi został obliczony przy założeniu optymalnego rozdziału bitów między poszczególne kwantyzatory. Ze wzoru (4.17) lub (4.28) otrzymuje się rozdzielczości (w bitach na próbkę), które dla kwantyzatorów skalarnych muszą być zaokrąglone do wartości całkowitych dodatnich. Oczywiście powiększa to błąd kwantyzacji.



Rysunek 4.5: Wykorzystanie maskowania w koderze MPEG (RSM- stosunek poziomu sygnału do poziomu maskowania)

4.3 Kodowanie w dziedzinie częstotliwości

4.3.1 Wykorzystanie zestawu filtrów o zmiennej rozdzielczości

Zalety kodowania w dziedzinie częstotliwości

Transformata optymalna (KLT i inne, rozważane w p.4.2.5) zapewnia największy zysk (najmniejszy błąd kwantowania), jednak w praktyce wykorzystuje się głównie transformaty przenoszące sygnał w dziedzinę częstotliwości (DFT, DCT, transformaty "z nakładaniem" [130]). Główną zaletą kodowania w dziedzinie częstotliwości jest możliwość wykorzystania efektów psychoakustycznych, polegających na maskowaniu szumu kwantyzacji sygnałem akustycznym [209], [107], [106]. Psychoakustyczny model maskowania został wykorzystany m.in. w koderach objętych normą MPEG [157].

Zasadę maskowania wyjaśniono na rys.4.5. Znając widmo sygnału fonicznego oblicza się, w funkcji częstotliwości tzw. próg maskowania, tzn. maksymalny poziom zakłóceń, które nie są słyszalne w obecności sygnału fonicznego. Jeżeli poziom szumu kwantyzacji nie przekracza progu maskowania, to szum kwantyzacji jest niesłyszalny. Oznacza to, że stosunek poziomu sygnału do szumu kwantyzacji nie powinien być mniejszy od stosunku RSM (fr: *rapport signal-masque*, ang: *signal to mask ratio*). W MPEG osiąga się to kodowaniem sygnału w 32 podpasmach i odpowiednim rozdziałem bitów między poszczególne podpasma (jeżeli $RSM < 0$ dB, to kwantowanie w danym podpasmie jest zbędne ⁴).

⁴Aby to podkreślić, na rys.4.5 wykreślono RSM [dB] jedynie w tych podpasmach, w których jest on dodatni

W koderach operujących w dziedzinie częstotliwości odchodzi się nie tylko od optymalnej (w sensie minimalizacji mocy błędu kwantowania) transformaty, ale także od optymalnego rozdziału bitów. Optymalny rozdział bitów wymaga spełnienia równania (4.16) lub (4.27), co oznacza jednakową moc błędu kwantowania wszystkich wektorów wydzielonych z przetworzonego sygnału. W przypadku transformat przenoszących sygnał w dziedzinę częstotliwości, wymienione wektory odpowiadają poszczególnym podzakresom częstotliwości. Oznacza to, że optymalny rozdział bitów gwarantuje stałą gęstość mocy szumu kwantyzacji w funkcji częstotliwości. Z kolei rozdział bitów w oparciu o model psychoakustyczny oznacza wyprofilowanie gęstości mocy szumu kwantyzacji na kształt progu maskowania (krzywej maskowania). Jest to niekorzystne pod względem energetycznym (większa moc szumu kwantowania), jednak korzystne pod względem percepcyjnym (szum kwantowania jest mniej słyszalny).

Takie kształtowanie gęstości mocy szumu kwantyzacji może być zrealizowane nie tylko przez modyfikację algorytmu rozdziału bitów. Można również wykorzystać filtry percepcyjne: $W(z)$ po stronie nadawczej i $W^{-1}(z)$ po stronie odbiorczej. Jeśli transmitancja tego ostatniego odpowiada krzywej maskowania, to bity można rozdzielać mając na uwadze korzyść energetyczną (minimalizując moc błędu kwantowania).

Koder sygnału telefonicznego oparty na zestawie filtrów o zmiennej rozdzielczości

W p.4.2 opisano zestaw filtrów o strukturze hierarchicznej i zmiennej rozdzielczości (rys.4.2). W pracach prowadzonych z udziałem autora niniejszej rozprawy [144], [53] zastosowano taki zestaw filtrów do kodowania sygnału telefonicznego. Omawiany zestaw filtrów zapewnia większą rozdzielczość (filtracja wąskopasmowa) w zakresie niższych częstotliwości, a mniejszą rozdzielczość (szersze pasmo filtrów) w zakresie wyższych częstotliwości. Odpowiada to właściwościom ucha ludzkiego (skala w barkach [209], [107]). Jednocześnie w zakresie wyższych częstotliwości możliwe jest uzyskanie większej rozdzielczości w dziedzinie czasu, dzięki większej częstotliwości próbkowania. Umożliwia to dokładniejsze kodowanie szybkich transjentów (głoski plosyjne, zawierające znaczną część energii w zakresie wyższych częstotliwości).

Zestaw filtrów przetwarza 16 kolejnych próbek sygnału wejściowego na 16 próbek sygnałów subpasmowych, według schematu z rys.4.3. W symulowanym koderze filtracji podaje się kolejne bloki liczące $16N$ próbek sygnału wejściowego ($N = 8$), otrzymując 16 8-wymiarowych wektorów sygnałów subpasmowych (po jednym wektorze dla pasm 1-4, po dwa dla pasm 5-6, po cztery dla pasm 7-8).

Każdy z tych 16 wektorów może być kwantowany z wykorzystaniem wielostopniowego kwantyzatora wektorowego typu kształt - wzmocnienie (SGVQ). Liczbę etapów kwantowania ograniczono do 3, na każdy etap przypada 8 bitów. Założono, że przepływność binarna będzie mała, około 8-9.6 kbit/s. Stąd podstawowa rozdzielczość systemu kwantowania $b = 1$ bit na próbkę (pewną liczbę bitów należy zarezerwować na informacje dodatkowe, np. dotyczące rozdziału bitów między 16 kwantowanych wektorów). Rozdzielczość $b = 1$ oznacza, że do zakodowania bloku liczącego $16N = 128$ próbek potrzeba 128 bitów, czyli $128/8=16$ elementarnych kwantyzatorów wektorowych 8-bitowych, z których każdy obsługuje pojedynczy etap w wielostopniowym SGVQ.

Przetestowano kilka sposobów rozdzielania 128 bitów między 16 wektorów. W każdym przypadku bity rozdziela się ósemkami (każda ósemka odpowiada jednemu elementarnemu kwantyzatorowi wektorowemu). Dla uproszczenia zastosowano kryterium energetyczne - bity przydziela się wektorowi o największej normie. Ponieważ wektory kształtu w elementarnych SGVQ są znormalizowane, zatem norma kwantowanego wektora jest proporcjonalna do wybranego wzmocnienia. Po przydzieleniu elementarnego 8-bitowego SGVQ dokonuje się modyfikacji normy tak, aby odpowiadała ona normie błędu kwantowania w tym SGVQ. W pierwszym przybliżeniu dzieli się normę (a ściślej - wykorzystany współczynnik wzmocnienia) przez dwa. Ma to następujące uzasadnienie: 8-bitowy SGVQ operujący na wektorze 8-wymiarowym działa z rozdzielczością 1 bit/próbkę - a więc zmniejsza amplitudę błędu kwantowania co najmniej dwukrotnie (reguła wywodząca się z kwantyzacji skalarnej). Następnie wyszukuje się kolejny wektor o największej normie (największym współczynniku wzmocnienia) itd.

Należy zwrócić uwagę na fakt, że rozdzielenie 16 elementarnych kwantyzatorów między 16 wektorów może być zrealizowane na $\binom{31}{16}$, czyli 300540195 sposobów. Jeśli ograniczy się liczbę przydzielanych kwantyzatorów na wektor (tzn. liczbę etapów w wielostopniowym SGVQ) do 3, to i tak liczba możliwych rozdziałów wyniesie 79067375. Wybrany rozdział kwantyzatorów stanowi informację dodatkową, którą trzeba byłoby przesłać w słowie 27-bitowym. Zauważmy jednak, że algorytm rozdzielania kwantyzatorów korzysta ze współczynników wzmocnienia SGVQ. Jeśli wiadomo, jakie z 16 wektorów są w ogóle kwantowane (otrzymują choć jeden kwantyzator), to przydzielanie następnych kwantyzatorów może być zrealizowane bez przesyłania informacji dodatkowej. Ostatecznie przesyła się (w 16 bitach) numery kwantowanych wektorów, następnie wyniki pierwszego etapu kwantowania (w tym współczynniki wzmocnienia SGVQ). Na podstawie tych współczynników wzmocnienia podejmuje się decyzję, jakim wektorom należy przydzielić kolejne kwantyzatory.

W innym wariantcie symulowanego koderza bity przydzielano 8 podpasmom częstotliwości. W tej sytuacji należało wprowadzić kwantyzatory wektorowe operujące na wektorach 8, 16, 24 i 32 wymiarowych. Np. przydzielenie jedynie 8 bitów piątemu podpasmu oznacza kodowanie 16-wymiarowego wektora w pojedynczym elementarnym 8-bitowym SGVQ.

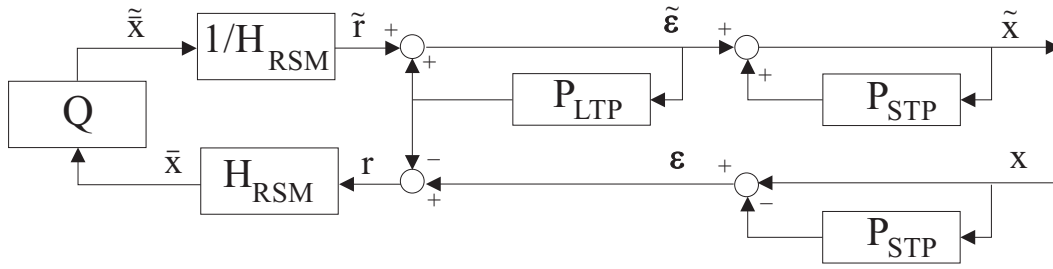
Kolejną modyfikacją było wprowadzenie filtracji słownika (metodą "wstecz", jak w LDCELP) w SGVQ operujących w czterech pierwszych podpasmach.

W wyniku symulacji uzyskiwano lepsze wartości SNR niż w koderze CELP o zbliżonej przepływności, jednak szum kwantowania był bardziej dokuczliwy. Po tych doświadczeniach prowadzono dalsze badania z większą przepływnością binarną i z sygnałem szerokopasmowym (16 kHz).

4.3.2 Predykcyjne kodery transformaty

Kodery MPEG umożliwiają uzyskanie sygnałów fonicznych o dobrej jakości dla przepływności większych od 64 kbit/s. Ich główną zaletą jest możliwość wykorzystania efektów maskowania szumu kwantyzacji sygnałem użytecznym.

Kodery MPEG mają jednak swoje wady - nie można w nich wykorzystać korelacji krótko-



Rysunek 4.6: Koder predykcyjny z kwantowaniem transformaty: P_{STP} - predyktor krótkookresowy, P_{LTP} - predyktor długookresowy, H_{RSM} - filtr o transmitancji aproksymującej RSM, Q - obliczanie transformaty, kwantyzacja i obliczanie transformaty odwrotnej

i długookresowych sygnału mowy i wielu sygnałów muzycznych. Korelacje te są z powodzeniem wykorzystywane w koderach predykcyjnych CELP, stosowanych do kompresji sygnałów telefonicznych. Pojawiają się próby zastosowania koderów CELP do kompresji szerokopasmowych sygnałów akustycznych. W pracy [151] opisano koder sygnału o pasmie 16 kHz i szybkości transmisji 64 kbit/s, działający w oparciu o algorytmy zbliżone do LDCELP (*Low Delay Code Excited Linear Prediction* - norma ITU-T G.728 dla kodowania sygnału telefonicznego z przepływnością 16 kbit/s [19]). Wadą koderów CELP jest to, że kwantują one sygnał w całym pasmie, nie dając możliwości rozdzielania bitów między podpasma częstotliwości.

W predykcyjnych koderach transformaty [117, 30] wykorzystuje się zarówno rozdział bitów w dziedzinie częstotliwości, jak i korelacje sygnału w dziedzinie czasu. Na rys.4.6 pokazano schemat takiego kodera - górna część rysunku stanowi schemat dekodera, który jest integralną częścią kodera (tzw. dekodek lokalny). Wektor \bar{N} kolejnych próbek sygnału akustycznego \mathbf{x} podlega stopniowej dekorelacji w procesie predykcji krótkookresowej (gdzie otrzymuje się sygnał resztkowy ϵ) i długookresowej (gdzie otrzymuje się sygnał resztkowy \mathbf{r}). Zdekorelowany w znacznym stopniu sygnał resztkowy \mathbf{r} ma widmo o obwiedni znacznie "spłaszczonej" w porównaniu z widmem sygnału akustycznego \mathbf{x} . Zrekonstruowany sygnał akustyczny $\tilde{\mathbf{x}}$ otrzymuje właściwy skład widmowy dzięki kaskadowo połączonym filtrom z predyktorami długo- i krótkookresowym w pętli sprzężenia zwrotnego. Transmitancja kaskady tych filtrów

$$\frac{1}{A(z)} = \frac{1}{1 - P_{LTP}(z)} \frac{1}{1 - P_{STP}(z)} \quad (4.69)$$

jest dobrym modelem widma sygnału akustycznego.

Właściwe ukształtowanie widma szumu kwantyzacji zapewnia filtr $H_{RSM}(z)$ o transmitancji modelującej RSM i $1/H_{RSM}(z)$, którego transmitancja modeluje stosunek progu maskowania do poziomu sygnału. Kwantowaniu podlega tzw. sygnał percepcyjny $\tilde{\mathbf{x}} - \bar{\mathbf{x}}$ - odbywa się ono w taki sposób, aby szum kwantyzacji $\tilde{\mathbf{x}} - \bar{\mathbf{x}}$ miał możliwie płaską obwiednię widma. Po przejściu przez filtry $1/H_{RSM}(z)$ i $1/A(z)$ szum kwantyzacji na wyjściu dekodera (tzn. sygnał $\tilde{\mathbf{x}} - \mathbf{x}$) uzyskuje gęstość mocy o kształcie odpowiadającym progowi maskowania. Jest to najkorzystniejsze, z punktu widzenia percepcji, ukształtowanie widma szumu kwantyzacji.

Niestety w koderze predykcyjnym nie udaje się w tak idealny sposób ukształtować widma szumu kwantyzacji. Główną przyczyną jest trudność w realizacji filtrów $H_{RSM}(z)$ i $1/H_{RSM}(z)$. Filtr $H_{RSM}(z)$ zastępuje się zwykle filtrem $1/A(z/\gamma)$, gdzie $\gamma \approx 0.8$ jest współczynnikiem tłumienia rezonansów filtru $1/A(z)$, modelującego widmo sygnału akustycznego. Jest to bardzo niedoskonała aproksymacja RSM. Ponadto w klasycznym koderze CELP trudno jest uzyskać płaską obwiednię widma sygnału $\tilde{\mathbf{x}} - \bar{\mathbf{x}}$.⁵

W predykcyjnym koderze transformaty jest jednak dodatkowa możliwość kształtowania widma szumu kwantyzacji. Jest nią adaptacyjny rozdział bitów między współczynniki transformaty, czyli podpasma częstotliwości. W pracy [117] problem ten rozwiązano w następujący sposób.

- Oblicza się dyskretną transformatę Fouriera (DFT) wektora próbek sygnału percepcyjnego: $\bar{\mathbf{y}} = \bar{\mathbf{T}} \bar{\mathbf{x}}$;
- Dokonuje się predykcji widma amplitudy $|\bar{\mathbf{y}}|$, wykorzystując zakodowane widmo amplitudy poprzedniego wektora sygnału percepcyjnego: $\alpha|\tilde{\mathbf{y}}(-\mathbf{1})|$, gdzie α jest współczynnikiem predykcji;
- Koduje się różnicę $|\bar{\mathbf{y}}| - \alpha|\tilde{\mathbf{y}}(-\mathbf{1})|$, metodą kwantyzacji wektorowej. Po dodaniu skwantowanej różnicy do $\alpha|\tilde{\mathbf{y}}(-\mathbf{1})|$ otrzymuje się zakodowane widmo amplitudy $|\tilde{\mathbf{y}}|$;
- Posługując się zakodowanym widmem amplitudy $|\tilde{\mathbf{y}}|$ oblicza się rozdział bitów dla kodowania fazy. Współczynnikom widma o większych amplitudach przydziela się największą liczbę bitów (na każde 6 dB o 1 bit więcej). Rozdziału bitów dokonuje się aż do wyczerpania się zasobów, określonych przez przepływność binarną. Jeżeli danemu współczynnikowi nie przydzielono ani jednego bitu, to wówczas zeruje się również jego amplitudę. Taki rozdział bitów umożliwia uzyskanie w miarę płaskiego widma szumu kwantyzacji. Ważne jest to, że nieistotne z percepcyjnego punktu widzenia fragmenty widma nie są kodowane, a uzyskane bity są wykorzystane do kodowania bardziej istotnych fragmentów widma;
- Po skwantowaniu fazy odtwarza się całe skwantowane widmo $\tilde{\mathbf{y}}$ i oblicza się odwrotną transformatę Fouriera: $\tilde{\mathbf{x}} = \bar{\mathbf{T}}^{-1} \tilde{\mathbf{y}}$.

Badania symulacyjne wykazały, że opisany predykcyjny koder transformaty daje dobre rezultaty kodowania szerokopasmowego sygnału mowy (przepływność 32 kbit/s dla sygnału o pasmie 16 kHz). Jakość sygnałów muzycznych zależy silnie od materiału dźwiękowego: rodzaju instrumentów muzycznych, partii solowych bądź orkiestrowych itp. Dla muzyki o pasmie 16 kHz wymagana jest przepływność 64 kbit/s. Jakość jest wówczas porównywalna z odpowiednim wariantem MPEG o takiej samej przepływności (oba kodery dają niewielkie zniekształcenia, jednak o różnym brzmieniu).

⁵W koderze CELP kwantuje się wektorowo sygnał resztkowy \mathbf{r} , zastępując go sygnałem $\tilde{\mathbf{r}}$ w taki sposób, aby szum kwantyzacji $\tilde{\mathbf{x}} - \bar{\mathbf{x}}$, tzn. sygnał $\tilde{\mathbf{r}} - \mathbf{r}$ po przejściu przez filtr $1/A(z/\gamma)$, był jak najmniejszy.

Wykorzystanie predyktora długookresowego $P_{LTP}(z)$ jednocześnie z kodowaniem transformaty, jest rozwiązaniem dyskusyjnym. Kodowanie mowy w pasmie 16 kHz wymaga spełnienia sprzecznych ze sobą postulatów. Z jednej strony, wymiar przetwarzanych wektorów nie powinien być większy niż okres tonu krtaniowego, aby umożliwić predyktorowi długookresowemu odwołanie się do poprzedniego okresu tonu krtaniowego. Dla sygnału spróbkowanego z częstotliwością 32 kHz i częstotliwości tonu krtaniowego 250 Hz (głos kobiecy), okres tonu krtaniowego obejmuje 128 próbek. Z drugiej strony, uzyskanie dobrej rozdzielczości w dziedzinie częstotliwości wymaga transformowania dłuższych bloków, np. 512 próbek.

Zauważono, że w procesie kodowania większości sygnałów muzycznych efektywność predykcji długookresowej jest niewielka (nie dotyczy to niektórych instrumentów solowych). W pewnym stopniu predykcję długookresową zastępuje predykcja widma amplitudy. Potrzeba większej rozdzielczości (długie okna czasowe) jest bardziej widoczna w procesie kodowania muzyki niż mowy. Z tego względu proponowano zastosowanie krótkich okien i predykcji długookresowej dla mowy, a długich okien bez predykcji długookresowej dla muzyki [117, 30].

Dla uproszczenia (uniknięcia rozpoznawania muzyki i mowy), w toku dalszych prac usunięto predyktor długookresowy, otrzymując koder opisany w kolejnym punkcie (p.4.3.3).

4.3.3 Koder transformaty o małym opóźnieniu

Ogólne cechy zaprojektowanego kodera

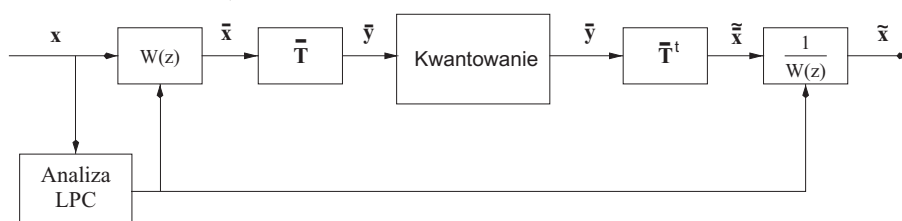
W ciągu ostatnich 20 lat pojawiło się wiele standardów kompresji sygnałów dźwiękowych, zarówno telefonicznych (normy ITU-T) jak i szerokopasmowych (normy ISO) [73, 153, 152, 154]. Dla aplikacji multimedialnych dokonano scalenia różnych algorytmów kompresji opracowując normę ISO MPEG-4 [156]. Obejmuje ona szereg koderów mowy i muzyki z przepływnościami od 2 do 24 kbit/s. Kodery te nie pokrywają wszystkich zastosowań w technice multimedialnej, np. wysokiej jakości audiokonferencji, ze względu na zbyt duże opóźnienie algorytmiczne i związany z tym problem echa. Nie są również przystosowane do współpracy z sieciami pakietowymi, gdzie może nastąpić utrata części przesyłanych pakietów.

Wad tych nie ma koder opisany w dwóch kolejnych podrozdziałach. Przystosowany on jest do kompresji szerokopasmowego sygnału fonicznego (mowy i muzyki w pasmie 16 kHz) przy niewielkim opóźnieniu algorytmicznym (rzędu 25 ms - dla porównania koder MPEG-AAC [155] wtrąca opóźnienie około 64 ms przy szybkości próbkowania 32 kHz). Szybkość transmisji (dla kanału monofonicznego) wynosi 64, 56, 48, 40 i 32 kbit/s. Nadajnik wysyła strumień binarny ze stałą szybkością 64 kbit/s, natomiast odbiornik może wykorzystywać tylko część tego strumienia (wysyłane pakiety o niższym priorytecie w transmisji nie dochodzą do odbiornika lub dochodzą z opóźnieniem przekraczającym dopuszczalny limit). Ograniczenie pojemności kanału transmisyjnego nawet do 50% nie powoduje znaczącego spadku jakości odtworzonego sygnału fonicznego.

Poniżej przedstawiono strukturę kodera: jest to koder transformaty, w którym kształtowanie widma szumu kwantyzacji odbywa się w odpowiednim filtrze "percepcyjnym" (podobnie jak w koderach CELP), natomiast rozdział zasobów binarnych odbywa się na zasadzie minimalizacji energii szumu kwantyzacji. Zastosowano zmodyfikowaną transformatę kosinu-

soidalną MDCT, łącząc kolejne współczynniki w pary tworzące liczby zespolone. Otrzymane w ten sposób widmo amplitudy jest kwantowane wektorowo w podpasmach częstotliwości, natomiast widmo fazy z korekcją amplitudy jest kwantowane z wykorzystaniem kwantyzatorów wektorowych dwuwymiarowych. Koder o stałej przepływności binarnej 64 kbit/s [103] został opisany w niniejszym punkcie (p.4.3.3). Koder o zmiennej przepływności binarnej, skalowalnej od 64 do 32 kbit/s skokami co 8 kbit/s [58, 146, 147] został opisany w p.4.3.4. W szczególności opisano słownik kwantyzatora wektorowego o strukturze hierarchicznej, niezbędny do dekodowania sygnału akustycznego w warunkach ograniczonej przepływności binarnej.

Ogólny schemat predykcyjnego koder transformaty pokazano na rys.4.7. W części wejściowej i wyjściowej przypomina on znane kodery typu CELP: sygnał wejściowy \mathbf{x} jest przepuszczany przez tzw. filtr percepcyjny nadawczy $W(z)$, otrzymany w wyniku analizy predykcyjnej. W efekcie powstaje sygnał percepcyjny $\bar{\mathbf{x}}$, który podlega kwantyzacji wektorowej w dziedzinie transformaty. Szum kwantyzacji ma w zasadzie płaskie widmo, natomiast zadaniem odbiorczego filtru percepcyjnego $1/W(z)$ jest takie jego ukształtowanie, aby był on jak najmniej dokuczliwy dla ucha (najlepiej niesłyszalny). Para macierzy transformacyjnych

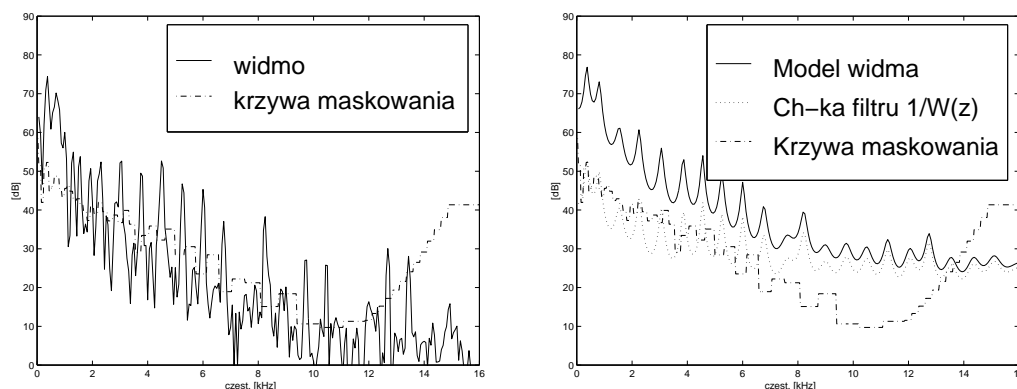


Rysunek 4.7: Ogólna struktura koder transformaty o zmniejszonym opóźnieniu: ($\bar{\mathbf{T}}$ - macierz transformacji prostej, $\bar{\mathbf{T}}^t$ - macierz transformacji odwrotnej)

$\bar{\mathbf{T}}$ i $\bar{\mathbf{T}}^t$ zapewnia przejście do dziedziny częstotliwości i powrót do dziedziny czasu. Poszczególne kodery transformaty różnią się głównie sposobem kwantyzacji widma $\bar{\mathbf{y}}$. W kolejnych podpunktach będą omówione podstawowe bloki koder i dekodera.

Filtracja percepcyjna

Filtr percepcyjny jest obliczany w oparciu o określony model maskowania szumu kwantyzacji sygnałem akustycznym. Na rys.4.8 pokazano (po lewej stronie) widmo sygnału skrzypiec i odpowiednią krzywą maskowania obliczoną według normy MPEG-1 [154]. Według przyjętego modelu percepcji, zakłócenia o widmie poniżej krzywej maskowania nie są słyszalne. Jednocześnie krzywa maskowania określa pożądany skład widmowy zakłóceń (w tym szumu kwantyzacji). Jeśli przyjąć hipotezę o płaskim widmie szumu kwantyzacji, to charakterystyka częstotliwościowa odbiorczego filtru percepcyjnego $1/W(z)$ powinna odpowiadać krzywej maskowania. Zagadnieniu projektowania filtrów percepcyjnych poświęcono wiele publikacji [158], [25], [24], [151], jednak w praktyce wystarczający jest filtr wyznaczany w oparciu o



Rysunek 4.8: Widmo i krzywa maskowania dla sygnału skrzypiec (po lewej); model predykcyjny widma sygnału, transmitancja filtru percepcyjnego $1/W(z)$ i krzywa maskowania (po prawej)

analizę predykcyjną, o następującej transmitancji:

$$W(z) = (1 - \mu z^{-1}) \frac{A(z/\gamma_1)}{A(z/\gamma_2)} \quad (4.70)$$

gdzie $A(z)$ jest predykcyjnym filtrem inwersyjnym rzędu p .

W opisanym koderze filtr $p=50$ -go rzędu jest obliczany co 16 ms (512 próbek). Wartości pozostałych parametrów wynoszą: $\mu = 0.5$, $\gamma_1 = 0.994$, $\gamma_2 = 0.8$. Parametry filtru (50 współczynników odbicia) są dzielone na 15 wektorów o różnym wymiarze (patrz tab.4.1) i kwantowane wektorowo w 6 bitach (co daje 90 bitów na 16 ms ramkę).

Tabela 4.1: Kwantyzacja współczynników filtru percepcyjnego: wymiary wektorów

1	1	2	2	2	3	3	3	4	4	4	5	5	5	6
---	---	---	---	---	---	---	---	---	---	---	---	---	---	---

Otrzymany filtr percepcyjny ma charakterystykę odbiegającą od krzywej maskowania (zwłaszcza w zakresie wyższych częstotliwości, gdzie uwidocznia się spadek wrażliwości ucha - patrz prawa strona rys.4.8), ale w praktyce spełnia on swoją funkcję (bardziej dokładne odwzorowanie krzywej maskowania nie daje pożądanego efektu w postaci wyraźnej poprawy jakości sygnału).

Transformacja czas - częstotliwość

Wybór transformaty nie jest sprawą łatwą ze względu na szereg trudnych do pogodzenia wymagań:

- idealna rekonstrukcja, tj. odwracalność przekształceń $\bar{\mathbf{T}}$ i $\bar{\mathbf{T}}^t$ - rys.4.7,
- dobra rozdzielczość w dziedzinie częstotliwości,

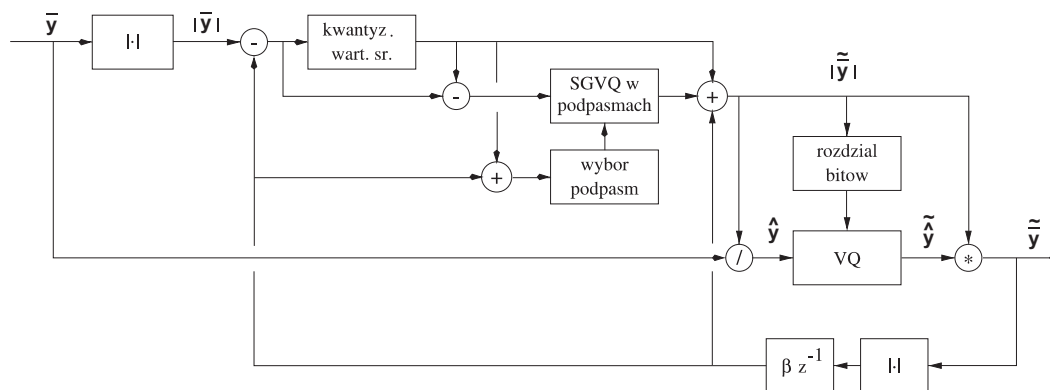
- c) maksymalna decymacja, tj. taka sama liczba kwantowanych współczynników po stronie częstotliwości i po stronie czasu,
- d) maksymalne nakładanie (zachodzenie na siebie) okien czasowych, w celu wyeliminowania zjawisk brzegowych ("terkotanie" z częstotliwością ramkowania i pre-echo),
- e) szybkie algorytmy obliczeniowe,
- f) małe opóźnienie algorytmiczne (dla zastosowań w bezpośredniej łączności do ok. 25 ms).

W p.4.2 opisano transformaty "z nakładaniem" o zmniejszonym opóźnieniu - patrz też [147, 58]. Transformaty te zapewniają idealną rekonstrukcję sygnału, maksymalną decymację, oraz dobrą rozdzielczość przy stosunkowo małym zachodzeniu na siebie kolejnych okien czasowych. Wybrano następujące wartości parametrów: transformata MDCT opisana macierzą liczącą $m = 512$ wierszy i $2m = 1024$ kolumny, efektywne nałożenie się okien $r = 256$ próbek. Opóźnienie algorytmiczne wynosi $m + r = 768$ próbek, co stanowi 24 ms przy częstotliwości próbkowania 32 kHz.

Współczynniki transformaty MDCT łączone są w pary (z wyłączeniem pierwszego), tworzące część rzeczywistą i urojoną widma zespolonego (z 512 współczynników MDCT otrzymuje się $256+1$ współczynników zespolonych). Zabieg ten ma na celu wykorzystanie systemu kwantyzatorów przeznaczonych do kodowania zespolonych współczynników DFT.

Kwantyzacja widma amplitudy

Widmo amplitudy jest kodowane według schematu pokazanego na rys.4.9. Po wyznaczeniu



Rysunek 4.9: Kwantowanie widma amplitudy i fazy

widma \bar{y} (257 wartości zespolonych, w tym pierwsza i ostatnia o zerowej części urojonej), oblicza się wartości bezwzględne poszczególnych składowych, otrzymując wektor $|\bar{y}|$. Podobnie jak w koderze TCX [117] oraz w predycyjnym koderze transformaty, opisanym w poprzednim punkcie, dokonuje się predykcji widma amplitudy z wykorzystaniem skwantowanego widma amplitudy z poprzedniej ramki. Współczynnik predykcji jest oznaczony na schemacie literą β . Kwantuje się go w 6 bitach. Dalsza obróbka jest wykonywana w 12 podzakresach częstotliwości, których długość (liczba współczynników w danym podzakresie) podana jest

Tabela 4.2: Podział skali częstotliwości (257 punktów) na M=12 podzakresów

8	8	8	10	11	13	14	18	20	26	32	89
---	---	---	----	----	----	----	----	----	----	----	----

w tab.4.2. Długości podzakresów odpowiadają w przybliżeniu dwukrotnej szerokości tzw. pasm krytycznych [209], [107].

Proces kwantyzacji rozpoczyna się od wyznaczenia wartości średnich widma amplitudy w podpasmach i ich kwantyzacji w 4 bitach. Następnie selekcjonuje się $M'=8$ podpasm o największej energii, w których przeprowadza się kwantyzację wektorową typu kształt- wzmocnienie (SGVQ). Cały system kwantyzatorów widma amplitudy wymaga 130 bitów na ramkę liczącą 512 próbek sygnału wejściowego (16 ms).

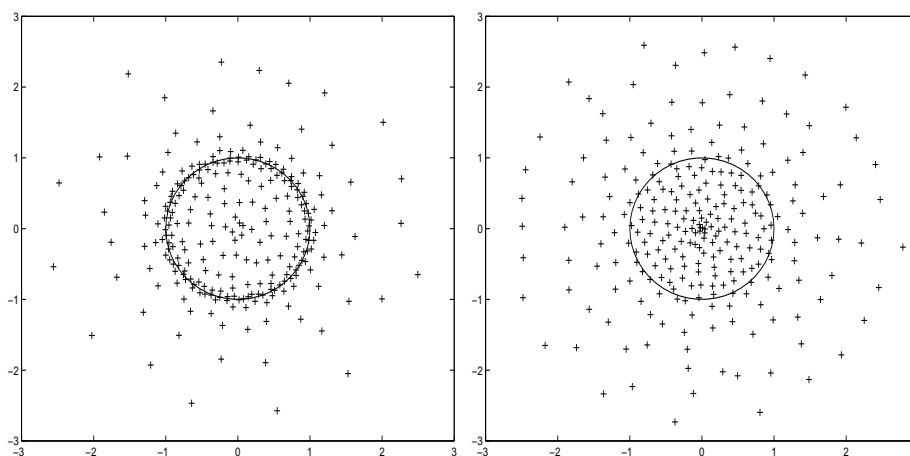
Kwantyzacja widma fazy

Widmem fazy \hat{y} nazywać będziemy wynik dzielenia zespolonych składowych wektora \bar{y} przez ich skwantowane wartości bezwzględne \tilde{y} (czyli skwantowane w poprzednim etapie widmo amplitudy). Tak zdefiniowane widmo zawiera współczynniki o różnych wartościach bezwzględnych, tym bardziej odbiegających od jedności, im mniej dokładne było kwantowanie widma amplitudy. Każdy z zespolonych współczynników jest kwantowany z wykorzystaniem kwantyzatora wektorowego 2-wymiarowego. Liczba bitów przydzielona każdemu kwantyzatorowi jest wyznaczana adaptacyjnie w odpowiednim algorytmie rozdziału bitów - rys.4.9.

Zasada rozdziału bitów jest prosta: dąży się do osiągnięcia minimum mocy błędu kwantyzacji, co prowadzi jednocześnie do "spłaszczenia" gęstości mocy szumu kwantyzacji. W praktyce bity przydziela się z ogólnej puli pojedynczo, wybierając współczynnik charakteryzujący się największą amplitudą błędu. Przydzielenie pojedynczego bitu zmniejsza amplitudę błędu o 3-6 dB⁶, co uwzględnia się modyfikując tę wartość po przydzieleniu bitu. Amplitudy błędu inicjalizuje się z wykorzystaniem skwantowanego widma amplitudy - w tej sytuacji nie trzeba przesyłać dodatkowej informacji o rozdziale bitów, gdyż można go odtworzyć w odbiorniku na podstawie skwantowanego widma amplitudy.

Kwantowane współczynniki zawierają informację o fazie oraz korekcji amplitudy. Ze względu na fakt, że widmo amplitudy jest kwantowane dokładniej w zakresie niskich, niż wysokich częstotliwości, właściwości kwantowanych współczynników zależą od częstotliwości. Współczynniki opisujące widmo w zakresie niskich częstotliwości mają wartości bezwzględne bardziej skupione wokół jedności, niż współczynniki opisujące widmo w zakresie wyższych częstotliwości. Uwzględniono to projektując odpowiednie słowniki kwantyzatorów wektorowych: zaprojektowano 12 kompletów słowników, operujących w podzakresach częstotliwości podanych w tab.4.2. Na rys.4.10 podano przykłady dwóch słowników, z których jeden operuje w zakresie niskich, a drugi w zakresie wysokich częstotliwości.

⁶3 dB, czyli $\sqrt{2}$ raza, gdy korekcja fazy jest znaczna i kwantyzator może być rozpatrywany jako dwuwymiarowy; 6 dB, czyli 2 razy, gdy brak jest korekcji fazy i kwantyzator jest de facto jednowymiarowy



Rysunek 4.10: Słowniki liczące po 128 wektorów dla niskich częstotliwości (lewa strona) i dla wysokich częstotliwości (prawa strona)

4.3.4 Koder o zmiennej przepływności binarnej

Koncepcja kodera o skalowalnej przepływności binarnej

Celem stosowania tego typu kodowania jest zapewnienie łączności audiokonferencyjnej przy wykorzystaniu kanałów transmisyjnych o różnej jakości. Nadajnik generuje strumień binarny o przepływności 64 kbit/s, natomiast do odbiorców może docierać cały strumień binarny, lub tylko jego wyselekcjonowana część (np. połowa). Podobny problem powstaje w przypadku transmisji pakietowej: połowa wysyłanych pakietów o najwyższym priorytecie jest niezbędna do odtworzenia sygnału po stronie odbiorczej, dalsze pakiety o niższym priorytecie tylko poprawiają jakość sygnału akustycznego. W zaprojektowanym koderze niezbędna do odtworzenia sygnału przepływność binarna wynosi 32 kbit/s, zwiększanie tej przepływności o kolejne 8 kbit/s stopniowo poprawia jakość sygnału, osiągając maksimum przy przepływności 64 kbit/s.

Skalowalność przepływności binarnej uzyskano w ten sposób, że wstępne etapy kwantowania, tj. kwantowanie współczynników filtra percepcyjnego (w 90 bitach) i kwantowanie widma amplitudy (w 130 bitach) nie zależą od przepływności binarnej. Zmienność przepływności binarnej pojawia się dopiero na etapie kwantowania widma fazy. Liczba bitów wykorzystywanych do kwantowania tego widma zależy od przepływności binarnej i jest podana w tab.4.3.

Jak już wspomniano, widmo amplitudy jest kwantowane niezależnie od przepływności binarnej. Oznacza to, że w predykcji widma amplitudy należy wykorzystywać widmo z poprzedniej ramki zakodowane z najniższą dokładnością (tj. z przepływnością 32 kbit/s, gdyż predykcja musi przebiegać w odbiorcach w identyczny sposób jak w nadajniku). Dalszą konsekwencją jest konieczność obliczania rozdziału bitów i kwantowania widma fazy w nadajniku: dla przepływności 32 kbit/s (w celu wykonania predykcji widma amplitudy) i dla przepływności 64 kbit/s (w celu skonstruowania pełnego strumienia binarnego). Generowany

Tabela 4.3: Liczba bitów dla kwantowania widma fazy

przepływność (kbit/s)	32	40	48	56	64
liczba bitów w ramce	512	640	768	896	1024
z tego dla widma fazy	292	420	548	676	804
procent	57	66	71	75	79

strumień binarny pojedynczej ramki jest podzielony na następujące części:

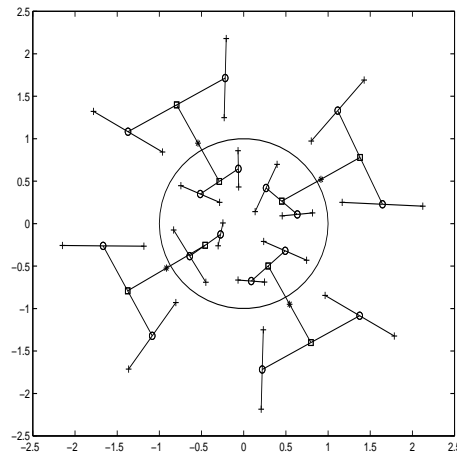
- (1)- 220 bitów opisujących filtr percepcyjny i widmo amplitudy,
- (2)- 292 bity opisujące widmo fazy dla przepływności 32 kbit/s,
- (3)- 128 bitów widma fazy dla przepływności 40 kbit/s (do rekonstrukcji sygnału potrzebne są również wszystkie poprzednie grupy bitów),
- (4)- 128 dodatkowych bitów widma fazy dla przepływności 48 kbit/s,
- (5)- 128 dodatkowych bitów widma fazy dla przepływności 56 kbit/s,
- (6)- 128 dodatkowych bitów widma fazy dla przepływności 64 kbit/s.

Z kolei w odbiorniku konieczna jest rekonstrukcja widma z przepływnością 32 kbit/s (dla wykonania predykcji widma amplitudy) i z przepływnością docelową, z którą aktualnie pracuje odbiornik. W oparciu o skwantowane widmo amplitudy (odpowiadające najniższej przepływności binarnej) jest możliwe obliczenie rozdziału bitów między kwantyzatory fazy dla każdej przepływności binarnej. Bity opisujące pojedynczy współczynnik widma fazy (tj. indeks wektora w słowniku) mogą być rozproszone w 5 częściach strumienia binarnego: najbardziej znaczące należą do części (2), a najmniej znaczące do części (6). Oczywiście niektóre współczynniki mogą nie być w ogóle kwantowane, inne mogą być kwantowane jedynie przy większych przepływnościach binarnych (odpowiednie bity mogą należeć np. jedynie do grupy (6)). W oparciu o informację zawartą w dwóch pierwszych grupach jest możliwe zlokalizowanie w strumieniu binarnym pełnych lub "obciętych" (tj. pozbawionych mniej znaczących bitów) indeksów wektorów opisujących widmo fazy.

Struktura słowników kwantyzatorów wektorowych

Słowniki kwantyzatorów wektorowych wykorzystywanych do kodowania widma fazy powinny mieć odpowiednią strukturę hierarchiczną. Oznacza to, że skrócenie słowa kodowego identyfikującego dany wektor słownika nie powinno prowadzić do znacznego zwiększenia błędu kwantyzacji. Przykład takich słowników o strukturze hierarchicznej podano na rys.4.11. Pokazano tam cztery słowniki liczące odpowiednio 4, 8, 16 i 32 wektory. Każdy wektor należący do słownika niższego szczebla znajduje się w pobliżu dwóch wektorów należących do słownika wyższego szczebla. Utrata jednego bitu w słowie kodowym oznacza przejście do słownika niższego szczebla, co w niezbyt wielkim stopniu zwiększa błąd kwantowania.

Projektowanie słowników hierarchicznych rozpoczyna się od słownika największego, liczącego 1024 wektory. Wykorzystuje się tu klasyczny algorytm Lloyda, przetwarzający zbiór



Rysunek 4.11: Słowniki o strukturze hierarchicznej zawierające 4 (gwiazdka), 8 (kwadrat), 16 (okrąg) i 32 (plus) wektorów

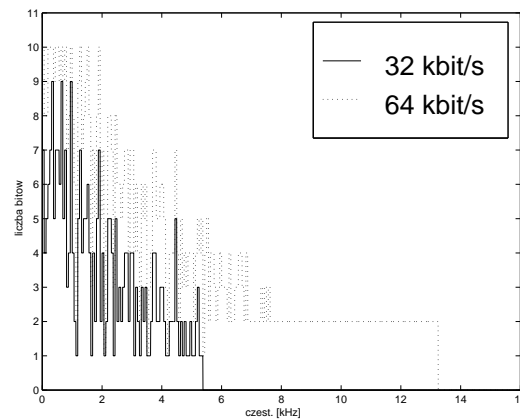
wektorów treningowych, otrzymany w procesie kwantowania sygnałów testowych [75]. Następnie, wykorzystując otrzymany słownik jako zbiór wektorów testowych, dzieli się go na dwie równe części o dużym skupieniu (z wykorzystaniem odpowiednio zmodyfikowanego algorytmu Lloyda). Wartości średnie wektorów należących do każdej części tworzą najmniejszy słownik "jednobitowy". Każdą z otrzymanych części dzieli się na połowy, otrzymując słownik czterowektorowy ("dwubitowy") itd.

Wyniki symulacji

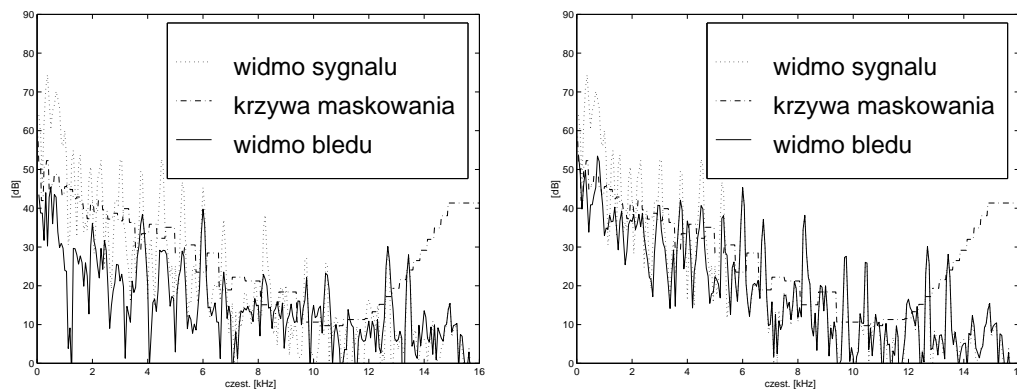
Do testowania kodera wykorzystano szereg fraz muzycznych (instrumenty solowe, orkiestra) i głosowych (mowa, śpiew). Jakość sygnału była bardzo dobra przy szybkości transmisji 64 kbit/s (w wyciszonym pomieszczeniu trudno było odróżnić sygnał zdekodowany od oryginału). Zmniejszenie przepływności prowadziło jednak do stopniowego pogorszenia jakości. Przy najmniejszej przepływności jakość sygnału silnie zależała od frazy.

Na rys.4.12 pokazano rozdział bitów dla pewnej ramki sygnału muzycznego przy przepływności 64 i 32 kbit/s. Przy najmniejszej przepływności binarnej następuje w sposób automatyczny ograniczenie pasma, ponieważ nie wystarcza bitów do zakodowania mniej energetycznych składników widma, a takie znajdują się z reguły w zakresie wyższych częstotliwości.

Na rys.4.13 pokazano widmo błędu kwantowania mierzone na wyjściu dekodera pracującego z przepływnością 64 i 32 kbit/s i porównano go z krzywą maskowania. Przy przepływności 64 kbit/s widmo sygnału błędu znajduje się prawie na każdej częstotliwości pod krzywą maskowania, czego nie można powiedzieć o widmie sygnału błędu przy przepływności 32 kbit/s.



Rysunek 4.12: Przykładowy rozdział bitów dla przepływności 64 i 32 kbit/s



Rysunek 4.13: Widmo błędu kwantyzacji przy przepływności 64 kbit/s (lewa strona) i 32 kbit/s (prawa strona) w porównaniu z krzywą maskowania

Odporność na przekłamania w kanale

Wersja kodera opisana w poprzednich punktach nie jest odporna na przekłamania w kanale transmisyjnym. Przyczyna jest następująca: rozdział bitów między kwantyzatory fazy jest obliczany na podstawie skwantowanego widma amplitudy. Przekłamanie bitów opisujących widmo amplitudy może doprowadzić do obliczenia błędnego rozdziału bitów dla widma fazy i, co za tym idzie, do błędnej interpretacji strumienia binarnego. Efekt ten może rozciągać się na kilka kolejnych ramek ze względu na mechanizm predykcji widma amplitudy. Aby temu zapobiec, zeruje się współczynnik predykcji widma amplitudy w sytuacji, gdy predykcja ta jest mało skuteczna (np. w obszarach transjentowych).

Bardzo skutecznym zabiegiem jest obliczanie rozdziału bitów w nadajniku i transmisja informacji o tym rozdziale do odbiornika. Rozdział bitów jest wyznaczany w obrębie podpasm o szerokości podanej w tab.4.2 (współczynniki transformaty należące do danego podzakresu częstotliwości otrzymują tę samą liczbę bitów). W ten sposób udaje się zakodować informację

o rozdziale bitów w 39-bitowym słowie kodowym. Kodowany jest tylko rozdział bitów dla największej przepływności binarnej. Rozdział bitów dla mniejszych przepływności otrzymuje się drogą usuwania najmniej znaczącego bitu z kolejnych słów kodowych, poczynając od najwyższych częstotliwości (faworyzując w ten sposób niskie częstotliwości, co jest korzystne ze względów percepcyjnych).

4.3.5 Podsumowanie

Opisano koder szerokopasmowego sygnału mowy i muzyki (spróbkowanego z częstotliwością 32 kHz) dla zastosowań multimedialnych, w szczególności wysokiej jakości audio- i wideokonferencji. Koder charakteryzuje się małym opóźnieniem (ok. 25 ms) i przepływnością skalowalną w zakresie od 64 kbit/s do 32 kbit/s z gradacją 8 kbit/s.

Nowym elementem w strukturze kodera jest wykorzystanie zmodyfikowanej transformaty kosinusoidalnej o zmniejszonym opóźnieniu i słowników kwantyzatorów wektorowych o strukturze hierarchicznej. Koder ma strukturę "elastyczną" - może być łatwo modyfikowany, np. dostosowany do innych przepływności binarnych. Liczba pośrednich przepływności binarnych może być dowolnie duża.

Przeprowadzono kompletną symulację kodera i dekodera z uwzględnieniem przekłamań w torze transmisyjnym. Wersja programu kodera napisana w języku C działa na stacji roboczej SUN/ULTRA 5 z szybkością 1.5 raza większą od czasu rzeczywistego, natomiast dekodek działa w czasie rzeczywistym (programowanie w assemblerze, a w szczególności wykorzystanie procesora sygnałowego rozwiązałyby problem złożoności obliczeniowej). Nieformalne odsłuchy fraz muzycznych i słownych wykazują przy szybkości transmisji 64 kbit/s jakość zbliżoną do kodera MPEG-1 Layer2, jednak przy opóźnieniu algorytmicznym zredukowanym do połowy. Przy wykorzystaniu 50% strumienia binarnego (przepływność 32 kbit/s) jakość jest nieco gorsza od jakości oferowanej przez koder G.722, który pracuje jednak z przepływnością 64 kbit/s.

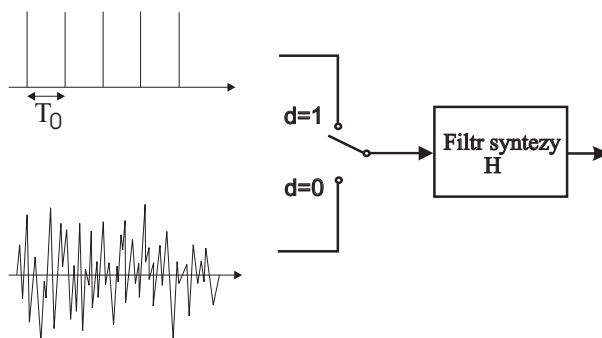
Rozdział 5

Parametryczne kodowanie mowy

5.1 Kodery parametryczne i kodery fali akustycznej

Omówione dotychczas metody kodowania mowy (ADPCM, VQ, CELP, kodery transformaty) miały jedną wspólną cechę - jest nią dążenie do jak najwierniejszego odtworzenia fali akustycznej (tj. przebiegu czasowego) sygnału. Kryterium zgodności sygnału odtworzonego z oryginalnym niekoniecznie musi być odległość euklidesowa - można definiując tę odległość uwzględnić także czynniki percepcyjne. Podobnie samo kodowanie nie musi odbywać się w dziedzinie czasu (można np. kodować w dziedzinie transformaty) - zawsze jednak ma się na uwadze dokładność odtworzenia fali akustycznej.

W przeciwieństwie do kodowania przebiegu czasowego w kodowaniu parametrycznym nie dąży się do minimalizacji odległości między przebiegiem odtworzonym i oryginalnym. Przepływności binarne (< 3 kbit/s) są tu zbyt małe, aby udało się dostatecznie dokładnie odtworzyć przebieg czasowy. Chodzi raczej o wywołanie zbliżonego wrażenia słuchowego. Kodery parametryczne mowy nazywane są wokoderami (ang. *vocoders* - słowo pochodzi od *voice coders*). Idea takiego wokodera, wywodząca się z pracy Dudley'a [48] została przedstawiona na rys.5.1. Mowa generowana jest przez przepuszczanie przez filtr syntezy odpo-



Rysunek 5.1: Zasada działania wokodera: T_0 - okres tonu krtaniowego, d - klasyfikator dźwięczności ($d=0$ dla głoski bezdźwięcznej, $d=1$ dla głoski dźwięcznej)

wiedniego sygnału z generatora pobudzenia. Jeśli mowa jest bezdźwięczna, jest to generator szumu. Jeśli mowa jest dźwięczna, jest to generator impulsów, wytwarzający impulsy "krtaniowe" z okresem powtarzania równym okresowi drgania strun głosowych (T_0 - okres tonu krtaniowego). Filtr syntezy zapewnia odpowiedni skład widmowy sygnału (jego rezonanse wyznaczają częstotliwości formantowe). W początkowych implementacjach wokodera filtr syntezy był realizowany z wykorzystaniem zestawu filtrów wąskopasmowych (wokoder kanałowy). Obecnie filtr syntezy jest najczęściej realizowany jako filtr predykcyjny o transmitancji $H(z) = 1/A(z)$. Mówimy wtedy o wokoderze predykcyjnym [198].

Oczywiście parametry pobudzenia i filtru syntezy muszą być mierzone po stronie nadawczej i przesyłane (co 10-30 ms) do odbiornika. Klasyfikacja sygnału (V- dźwięczny, ang. *voiced*, UV- bezdźwięczny, ang. *unvoiced*) i pomiar okresu tonu krtaniowego odbywa się w tzw. ekstraktorze tonu krtaniowego po stronie nadawczej.

W p.5.2 opisano wybrane metody klasyfikacji sygnałów, nadające się do zastosowania w

rozszerzonym ekstraktorze tonu krtaniowego. Ekstraktor taki powinien rozpoznawać większą liczbę klas sygnałów (nie tylko rozróżniać mowę dźwięczną od bezdźwięcznej) w celu poprawy jakości sygnału odtwarzanego po stronie odbiorczej. Z tego względu skupiono się na algorytmach progowych i dyskryminatorach liniowych. W toku badań porównawczych z wykorzystaniem rzeczywistych sygnałów okazało się, że dobre wyniki daje dyskryminator liniowy działający według algorytmu Fisher'a [207]. Przeprowadzono badania mające na celu wyłonienie parametrów sygnału, mających największe znaczenie dla prawidłowej dyskryminacji klas mowa/cisza, mowa dźwięczna/bezdźwięczna, mowa bezdźwięczna stacjonarna/plozyjna [50].

W p.5.3 przedstawiono algorytmy działania wokodera predykcyjnego o szybkości transmisji 2.4 kbit/s. Podstawową cechą wokodera jest zmodyfikowany generator sygnałów pobudzających filtr predykcyjny. Dla każdej z czterech klas sygnałów (mowa bezdźwięczna o charakterze stacjonarnym, mowa bezdźwięczna o charakterze plozyjnym, mowa słabo dźwięczna, mowa silnie dźwięczna) generowane są inne sygnały pobudzające. Do klasyfikacji sygnału mowy wykorzystano dyskryminatory liniowe. Opisano algorytmy ekstrakcji tonu krtaniowego, wyznaczania współczynników filtru predykcyjnego, generowania sygnałów pobudzających filtr predykcyjny, oraz obróbki końcowej sygnału. Opisano też metodę kwantowania transmitowanych parametrów. Przedstawiono realizację praktyczną na procesorze sygnałowym i wyniki badań jakościowych mowy.

W p.5.4 przedstawiono algorytmy działania wokodera predykcyjnego o szybkości transmisji 1.2 kbit/s. Opisano metodę interpolacji i kwantowania transmitowanych parametrów wokodera, oraz omówiono wyniki testowania wokodera.

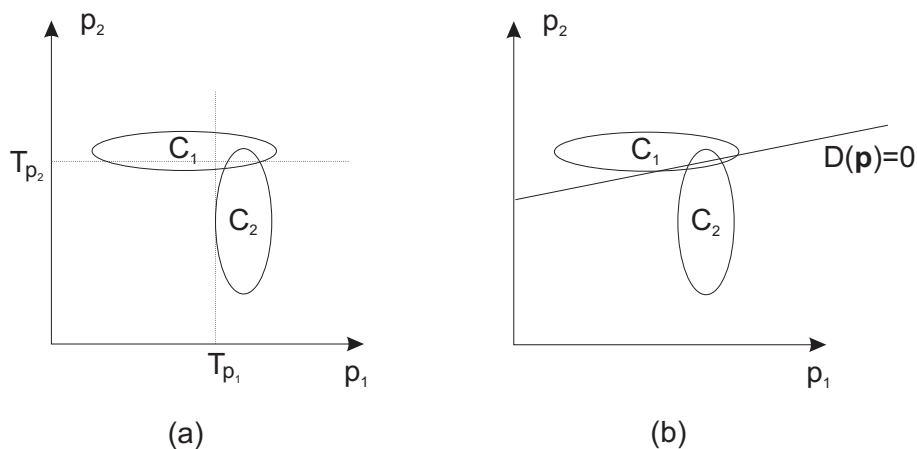
Klasyfikacja sygnałów jest stosowana także w niektórych koderach fali akustycznej. W wielu systemach telefonicznych (np. GSM) stosuje się dyskryminatory cisza/mowa, a w systemach o zmiennej przepływności opartych na CDMA (*Code Division Multiple Access*) - rozpoznaje się kilka klas sygnałów [42, 160]. W niektórych koderach o stałej przepływności klasyfikacja jest stosowana w celu poprawy jakości mowy [125, 202]. W p.5.5 opisano sposób wykorzystania ekstraktora tonu krtaniowego do poprawy jakości mowy w koderze CELP 4.8 kbit/s.

5.2 Metody dyskryminacji liniowej i ich zastosowanie do ekstrakcji tonu krtaniowego

5.2.1 Dyskryminatory

Ze względu na wielką różnorodność algorytmów decyzyjnych, klasyfikatorów, dyskryminatorów itp. zajmiemy się wyłącznie metodami wykorzystywanymi w zagadnieniu ekstrakcji tonu krtaniowego.

W wokoderach kanałowych i predykcyjnych przez długi okres czasu były stosowane dyskryminatory typu progowego. Najprostszy dyskryminator tego typu rozróżnia dwie klasy C_1 i C_2 , wykorzystuje jeden parametr sygnału p i wartość progową T_p . Jeżeli $p > T_p$, to sygnał należy do klasy C_2 , w przeciwnym wypadku należy do klasy C_1 . Np., $p = r_{max}$



Rysunek 5.2: Dyskryminator progowy (a) i liniowy (b)

może być wartością maksymalną znormalizowanej autokorelacji w określonym przedziale argumentu tej funkcji, $T_p = 0.3$, C_1 to klasa sygnałów mowy bezdźwięcznej, a C_2 - klasa sygnałów mowy dźwięcznej. W ogólnym przypadku mamy do czynienia z wektorem parametrów $\mathbf{p} = [p_1, p_2, \dots, p_N]^t$. W oparciu o wartości parametrów i wartości progowe możemy budować funkcje decyzyjne, np. typu "jeżeli $p_1 > T_{p_1}$ i $p_2 < T_{p_2}$, to sygnał należy do klasy C_2 , w przeciwnym przypadku należy do klasy C_1 ". Sytuację tę zilustrowano na rys.5.2a. Dyskryminatory progowe mogą być bardzo złożone (np. dyskryminator mowa dźwięczna/bezdźwięczna opisany w następnym podpunkcie), jednak nie zapewniają one optymalnego rozdziału klas, w sensie minimalizacji liczby błędnych decyzji. Dobrze ilustruje to rys.5.2a: w przypadku dwuwymiarowym linie podziału przebiegają zawsze równoległe do osi głównych, co jest poważnym ograniczeniem.

Ograniczenia tego są pozbawione dyskryminatory liniowe. Funkcją dyskryminującą jest liniowa kombinacja parametrów:

$$\hat{\mathbf{a}}^t \mathbf{p} = \sum_{i=1}^N \hat{\alpha}_i p_i \quad (5.1)$$

Jeżeli $\hat{\mathbf{a}}^t \mathbf{p} > T_p$, to sygnał należy do klasy C_2 , w przeciwnym przypadku należy do C_1 . Sytuację tę zilustrowano na rys.5.2b. Funkcja $D(\mathbf{p})$ jest zdefiniowana jako $\hat{\mathbf{a}}^t \mathbf{p} - T_p$.

W celu określenia współczynników funkcji dyskryminującej można wykorzystać algorytm Fisher'a [207]. Wymaga on zbudowania sekwencji treningowej $\{\mathbf{p}^j\}$. Następnie dokonuje się określenia wartości średnich wektora parametrów w obrębie dwóch klas:

$$\begin{aligned} \mathbf{m}_1 &= \frac{1}{n_1} \sum_{\mathbf{p}^j \in C_1} \mathbf{p}^j \\ \mathbf{m}_2 &= \frac{1}{n_2} \sum_{\mathbf{p}^j \in C_2} \mathbf{p}^j \end{aligned} \quad (5.2)$$

oraz macierzy kowariancji:

$$\begin{aligned}\bar{\mathbf{C}}_1 &= \frac{1}{n_1} \sum_{\mathbf{p}^j \in C_1} (\mathbf{p}^j - \mathbf{m}_1) (\mathbf{p}^j - \mathbf{m}_1)^t \\ \bar{\mathbf{C}}_2 &= \frac{1}{n_2} \sum_{\mathbf{p}^j \in C_2} (\mathbf{p}^j - \mathbf{m}_2) (\mathbf{p}^j - \mathbf{m}_2)^t\end{aligned}\quad (5.3)$$

gdzie n_1 i n_2 oznaczają liczbę wektorów sygnału treningowego należących odpowiednio do C_1 i C_2 .

Mając daną transformację $\hat{\alpha}^t \mathbf{p}$, określa się separację klas jako wartość bezwzględną wyrażenia

$$v_1 = \hat{\alpha}^t (\mathbf{m}_1 - \mathbf{m}_2) \quad (5.4)$$

natomiast wyrażenie

$$v_0^2 = \frac{1}{2} \hat{\alpha}^t (\bar{\mathbf{C}}_1 + \bar{\mathbf{C}}_2) \hat{\alpha} = \hat{\alpha}^t \bar{\mathbf{C}} \hat{\alpha} \quad (5.5)$$

jest miarą rozrzutu wartości parametrów "w kierunku" $\hat{\alpha}$.

Zadanie polega na znalezieniu takiego wektora współczynników $\hat{\alpha}$, który minimalizuje v_0^2 przy stałym v_1 . Minimalizując funkcję Lagrange'a ze współczynnikiem λ :

$$v = v_0^2 - \lambda v_1 = \hat{\alpha}^t [\bar{\mathbf{C}} \hat{\alpha} - \lambda(\mathbf{m}_2 - \mathbf{m}_1)] \quad (5.6)$$

otrzymuje się

$$\hat{\alpha} = \frac{\lambda}{2} \bar{\mathbf{C}}^{-1} (\mathbf{m}_2 - \mathbf{m}_1) \quad (5.7)$$

Współczynnik λ może być skalowany wraz z progiem, można zatem wybrać dowolną wartość, np. $\lambda = 2$. Wówczas optymalny wektor współczynników wyraża się wzorem:

$$\hat{\alpha} = \bar{\mathbf{C}}^{-1} (\mathbf{m}_2 - \mathbf{m}_1) \quad (5.8)$$

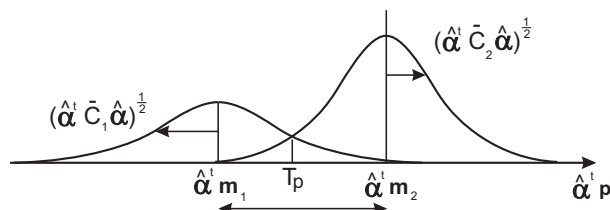
Próg T_p dla funkcji dyskryminującej $\hat{\alpha}^t \mathbf{p}$ wybiera się eksperymentalnie między $\hat{\alpha}^t \mathbf{m}_1$ i $\hat{\alpha}^t \mathbf{m}_2$ - patrz rys.5.3. Jeżeli wariancje wartości funkcji dyskryminującej w obrębie obu klas są zbliżone, to można skorzystać ze wzoru [207]:

$$T_p = \frac{(\hat{\alpha}^t \bar{\mathbf{C}}_2 \hat{\alpha})^{0.5} \hat{\alpha}^t \mathbf{m}_1 + (\hat{\alpha}^t \bar{\mathbf{C}}_1 \hat{\alpha})^{0.5} \hat{\alpha}^t \mathbf{m}_2}{(\hat{\alpha}^t \bar{\mathbf{C}}_1 \hat{\alpha})^{0.5} + (\hat{\alpha}^t \bar{\mathbf{C}}_2 \hat{\alpha})^{0.5}} \quad (5.9)$$

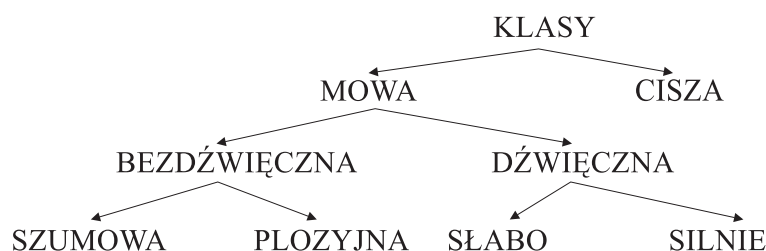
5.2.2 Rozszerzony ekstraktor tonu krtaniowego

Rozpoznawane klasy sygnałów

W kodowaniu mowy wykorzystuje się różnego rodzaju klasyfikację sygnałów. W algorytmach VAD (*Voice Activity Detector*- [70, 188]) oraz np. w GSM rozpoznaje się dwie klasy: ciszę (szum otoczenia) i mowę. W klasycznych wokoderach predykcyjnych rozróżnia się mowę



Rysunek 5.3: Wybór progu dla dyskryminatora liniowego



Rysunek 5.4: Schemat klasyfikacji sygnału mowy

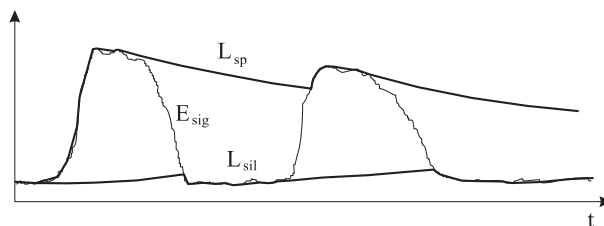
dźwięczną i bezdźwięczną (cisza jest klasyfikowana jako mowa bezdźwięczna) [198, 11]. W wokoderach nowej generacji [134, 54] wyróżnia się większą liczbę klas, np. mowa bezdźwięczna, plozyjna, słabo dźwięczna, silnie dźwięczna. W koderach CELP rozróżnia się niekiedy mowę dźwięczną, bezdźwięczną i fazę początkową głoski [202].

W rozszerzonym ekstraktorze tonu krtaniowego, zastosowanym w wokoderach opisanych w p.5.3 i w p.5.4 rozpoznaje się następujące klasy sygnałów (opisanych identyfikatorem dźwięczności d):

- cisza (szum otoczenia) i mowa bezdźwięczna o charakterze stacjonarnym, np. "sz": $d = 0$;
- mowa bezdźwięczna o charakterze plozyjnym (wybuchowym), np. "p": $d = 1$;
- mowa słabo dźwięczna, np. "ż": $d = 2$;
- mowa silnie dźwięczna, np. "a": $d = 3$.

Dla każdej z tych klas generuje się inny sygnał pobudzający filtr syntezy (p.5.3). Schemat klasyfikacji podano na rys.5.4. Ma on strukturę drzewa. Każdy z widocznych w strukturze 4 elementarnych klasyfikatorów rozpoznaje 2 klasy sygnałów.

Parametry sygnału wykorzystywane w klasyfikacji



Rysunek 5.5: Moc sygnału wejściowego E_{sig} , poziom mowy L_{sp} i poziom ciszy L_{sil}

W wokoderach opisanych w p.5.2 i w p.5.3 klasyfikacja sygnału wejściowego jest dokonywana co pół 30 ms ramki wykorzystywanej do analizy predykcyjnej, czyli co 15 ms. Zakłada się, że częstotliwość próbkowania wynosi 8 kHz, a wartości próbek sygnału nie przekraczają zakresu $\langle -1, +1 \rangle$.

W algorytmie wykorzystuje się następujące wartości progowe, adaptowane do przetwarzanego sygnału: poziom mowy L_{sp} i poziom ciszy L_{sil} . Oblicza się je w oparciu o moc sygnału E_{sig} w obrębie ramki ekstraktora tonu krtaniowego:

$$E_{sig}(i) = \frac{1}{\bar{N}} \bar{\mathbf{x}}^t(i) \bar{\mathbf{x}}(i) \quad (5.10)$$

gdzie \bar{N} – długość ramki ekstraktora, i – numer ramki, $\bar{\mathbf{x}}(i)$ – \bar{N} -wymiarowy wektor próbek.

Poziom mowy L_{sp} jest maksymalną wartością mocy, obarczoną pewną inercją. Jest on obliczany w następujący sposób:

$$L_{sp}(i) = \begin{cases} E_{sig}(i) & \text{gdy } E_{sig}(i) > L_{sp}(i-1) \\ 0.995L_{sp}(i-1) & \text{gdy } E_{sig}(i) \leq L_{sp}(i-1) \end{cases} \quad (5.11)$$

Poziom ciszy (szumu otoczenia) obliczany jest w następujący sposób:

$$L_{sil}(i) = \begin{cases} E_{sig}(i) & \text{gdy } E_{sig}(i) < L_{sil}(i-1) \\ 1.02L_{sil}(i-1) & \text{gdy } E_{sig}(i) \geq L_{sil}(i-1) \end{cases} \quad (5.12)$$

Poziom ciszy jest ograniczony od dołu pewną wartością $L_{sil_{min}}$. Zapobiega to przybieraniu przez L_{sil} wartości zerowej. Przykładowy przebieg parametrów E_{sig} , L_{sil} i L_{sp} w funkcji czasu pokazano na rys.5.5.

Próg dzwięczności T_{sp} określa się jako minimalny poziom mowy dzwięcznej [134]. Jest on wyznaczany na podstawie zmierzonego poziomu mowy i poziomu ciszy:

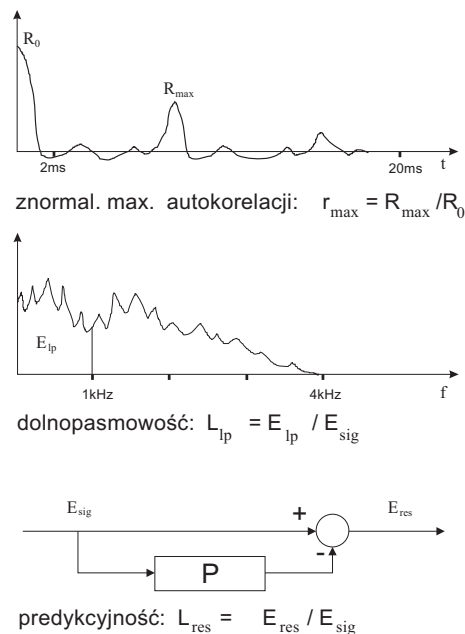
$$T_{sp}(i) = \max \{0.001L_{sp}(i), 4L_{sil}(i)\} \quad (5.13)$$

Ponadto wykorzystywano następujące parametry:

- energia (współczynnik autokorelacji R_0) sygnału błędu predykcyjnego przepuszczonego przez filtr dolnopasmowy o częstotliwości granicznej 1 kHz;

- maksimum autokorelacji R_{max} ww. sygnału w zakresie zmienności okresu tonu krtaniowego;
- moc sygnału wejściowego odfiltrowanego dolnopasmowo (0-1 kHz) E_{lp} i górnopasmowo (3-4 kHz) E_{hp} ;
- moc sygnału błędu predykcji E_{res} ;
- stosunek wartości maksymalnej do wartości skutecznej w sygnale błędu predykcji r_{max}/rms .

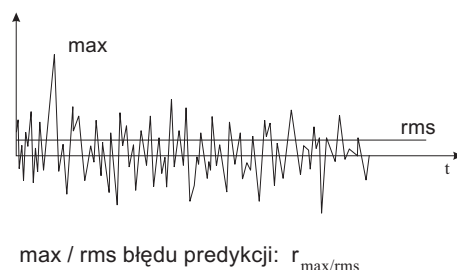
Liczba wyznaczanych współczynników autokorelacji odpowiada maksymalnej wartości okresu tonu krtaniowego T_0 , którą tu określono na 20 ms, co odpowiada 160 próbkom. Współczynniki autokorelacji R_0, \dots, R_{160} obliczane są w poszerzonym oknie, obejmującym co najmniej 2 okresy tonu krtaniowego. W opisanych w niniejszej pracy wokoderach za długość tego poszerzonego okna przyjęto 1.5 ramki analizy predykcyjnej, czyli 45 ms, co odpowiada 360 próbkom. Wartości maksymalnej R_{max} poszukuje się w zakresie 2 ms – 20 ms (bada się współczynniki autokorelacji w przedziale $< 16, 160 >$). Odpowiada to zakresowi zmian okresu tonu krtaniowego z wyłączeniem przypadków skrajnych.



Rysunek 5.6: Najważniejsze parametry wykorzystywane w klasyfikacji mowa dźwięczna - mowa bezdźwięczna

W procesie klasyfikacji sygnału bardziej przydatne są wartości znormalizowane ww. parametrów: znormalizowane maksimum autokorelacji $r_{max} = R_{max} / R_0$, współczynniki charakteryzujące dolnopasmowość i górnopasmowość sygnału $L_{lp} = E_{lp} / E_{sig}$, $L_{hp} = E_{hp} / E_{sig}$,

współczynnik charakteryzujący predykcyjność $L_{res} = E_{res}/E_{sig}$. Te z wymienionych parametrów, które okazały się najbardziej przydatne w procesie klasyfikacji, pokazano na rys.5.6 i rys.5.7.



Rysunek 5.7: Stosunek wartości maksymalnej do wartości skutecznej sygnału błędów predykcji - wykorzystywany do rozpoznawania pobudzenia pozytywnego

Dyskryminator cisza/mowa

Dyskryminator cisza/mowa, zastosowany w rozszerzonym ekstraktorze tonu krtaniowego, jest układem prostym, opartym na kryteriach energetycznych. Energia sygnału może być wyrażona w sposób bezwzględny (jako E_{sig}) lub względny (jako E_{sig}/L_{sil}).

Do testowania algorytmów klasyfikacji wykorzystano 10 krótkich fraz mowy, wypowiedzianych przez różnych mówców w różnych warunkach. Wzorcowe wyniki uzyskiwano metodą "ręczną", osobno dla każdej ramki sygnału.

Wykorzystując bezwzględną miarę mocy sygnału, otrzymano następujący dyskryminator progowy: ramka sygnału jest klasyfikowana jako mowa, gdy $E_{sig} > 1.3E - 6$. Dla wspomnianych 10 fraz otrzymano 7% błędów. Wykorzystując względną miarę mocy sygnału, uzyskano następujący dyskryminator progowy: ramka sygnału jest klasyfikowana jako mowa, gdy $E_{sig}/L_{sil} > 3.4$. Dla wspomnianych 10 fraz otrzymano 3.5% błędów. Dalszą niewielką poprawę uzyskano biorąc pod uwagę obie miary mocy sygnału i stosując dyskryminator liniowy Fisher'a:

$$4.83\sqrt{E_{sig}} + 0.0274\sqrt{E_{sig}/L_{sil}} > 0.056 \quad (5.14)$$

Dyskryminator Fisher'a (w formie nierówności dla kombinacji liniowej parametrów, w tym wypadku $\sqrt{E_{sig}}$ i $\sqrt{E_{sig}/L_{sil}}$) zapewnia najlepszą liniową separację dwóch klas. Pierwiastki kwadratowe dodano, aby zmniejszyć wariancję parametrów, zwłaszcza w klasie mowy. Otrzymano 3.1% błędów dla 10 fraz testowych, które były wykorzystywane do obliczenia dyskryminatora. Dla innych fraz otrzymywano od 0% do 12% błędów. Błędy te występują głównie na początku lub na końcu wyrazów, gdzie trudno o jednoznaczną klasyfikację nawet metodą "ręczną".

Dyskryminator mowa dźwięczna/bezdźwięczna

Dyskryminator progowy

Dyskryminatory progowe, nieraz bardzo rozbudowane, stanowiły przez długi czas niezbędny element wokoderów o małej przepływności binarnej. Do lepszych rozwiązań tego typu należy algorytm zastosowany w wokoderze predykcyjnym [54], opracowany przez autora niniejszej rozprawy na podstawie sugestii opublikowanych przez T.P.Barnwella (patrz np. [134]).

W algorytmie dyskryminacji mowa dźwięczna/bezdźwięczna wykorzystano omówione poprzednio parametry: E_{sig} , R_0 , R_{max} , E_{lp} , E_{hp} , E_{res} , $r_{max/rms}$. Ponadto do pomiaru okresu tonu krtaniowego stosowane są następujące parametry:

- i_{max} - położenie maksimum autokorelacji, w zakresie zmienności okresu tonu krtaniowego, wyznaczone dla odfiltrowanego dolnopasmowo sygnału błędu predykcji. Wartość i_{max} wyraża się w wielokrotności okresu próbkowania. Poszukuje się jej w zakresie 2-20 ms, co (przy częstotliwości próbkowania 8 kHz) oznacza, że $16 \geq i_{max} \geq 160$;
- \bar{i}_{max} - ważona wartość średnia okresu tonu krtaniowego (wyrażona jako wielokrotność okresu próbkowania). Wartość \bar{i}_{max} jest wykorzystana dla korekcji nietypowych, najczęściej błędnych wartości i_{max} . Dla mowy bezdźwięcznej $\bar{i}_{max} \rightarrow 100$. Wartość \bar{i}_{max} oblicza się w następujący sposób:

$$\bar{i}_{max}(i) = \begin{cases} 0.5 [i_{max}(i-1) + \bar{i}_{max}(i-1)], & \text{jeżeli poprzednia ramka była dźwięczna,} \\ 0.95\bar{i}_{max}(i-1) + 5, & \text{jeżeli poprzednia ramka była bezdźwięczna.} \end{cases} \quad (5.15)$$

Wartości $R_{max}(i)$ i $i_{max}(i)$ podlegają korekcji, w celu ograniczenia zjawiska podwajania (lub zwielokrotniania) okresu tonu krtaniowego. Na początku $R_{max}(i)$ mnoży się przez $\frac{4N}{4N-i_{max}(i)}$, gdzie $N = 160$, aby zmniejszyć efekt ważenia autokorelacji wynikający z wykorzystania okna prostokątnego o długości $2N$ do jej obliczania. Następnie $i_{max}(i)$ podlega modyfikacji z wykorzystaniem $i_{max}(i-1)$ i $\bar{i}_{max}(i)$. Jeżeli $i_{max}(i)$ przybiera wartość minimalną równą 16 (maksimum autokorelacji leży na brzegu przedziału $< 16, 160 >$), to wówczas nie jest ona brana pod uwagę (podstawia się $i_{max}(i) = 0$ oraz $R_{max}(i) = 0$). Jeżeli $i_{max}(i) \in < 1.8i_{max}(i-1), 2.2i_{max}(i-1) >$, to oznacza to podwajanie okresu tonu krtaniowego i $i_{max}(i)$ dzieli się przez 2. Wartości nietypowe, większe od $2.2i_{max}(i-1)$ są zastępowane przez $i_{max}(i-1)$, jeżeli $i_{max}(i-1) \neq 0$. Wartości większe od $2.2\bar{i}_{max}(i)$ są ignorowane (zastępowane zerami). Wartości mniejsze niż $0.4i_{max}(i-1)$ są zastępowane przez $i_{max}(i-1)$. Należy zauważyć, że opisany algorytm korekcji nie wprowadza dodatkowego opóźnienia.

Na algorytm dyskryminacji składają się następujące kroki:

1. Jeżeli $E_{sig}(i) < T_{sp}(i)$, to ramka i jest uznawana za bezdźwięczną ($d = 0$ lub 1);
2. Jeżeli $R_{max}(i) < 0.25R_0(i)$, to ramka i jest uznawana za bezdźwięczną ($d = 0$ lub 1);
3. Jeżeli $E_{hp}(i) > 0.25E_{sig}(i)$, to ramka i jest uznawana za bezdźwięczną ($d = 0$ lub 1);
4. Jeżeli $R_{max}(i) < 0.36R_0(i)$ i $E_{lp}(i) < 0.06E_{sig}(i)$ oraz $E_{res}(i) > 0.05E_{sig}(i)$, to ramka i jest uznawana za bezdźwięczną ($d = 0$ lub 1);

5. Dla ramek uznanych za bezdźwięczne przeprowadza się test na plozyjność:

- jeżeli $r_{max}/r_{rms} > 4$, to ramka jest uznawana za plozyjną ($d = 1$),
- w pozostałych przypadkach ma ona charakter stacjonarny ($d = 0$).

Na tym kończy się test dla ramek bezdźwięcznych. Dla pozostałych ramek wykonuje się test:

6. Jeżeli $R_{max}(i) > 0.55R_0(i)$ i $E_{lp}(i) > 0.01E_{sig}(i)$ oraz $E_{res}(i) < 0.2E_{sig}(i)$, to ramka jest silnie dźwięczna ($d = 3$);

7. w pozostałych przypadkach jest ona słabo dźwięczna ($d = 2$).

Uproszczoną wersję opisanego algorytmu zaimplementowano w wokoderze predykcyjnym opisanym w [54].

Wartości progowe zależą od charakterystyki częstotliwościowej mikrofonu, parametrów toru akustycznego i rodzaju występujących zakłóceń (szum otoczenia). Zostały one wybrane eksperymentalnie. Testowanie przeprowadzono w oparciu o wspomniane 10 fraz mowy. Nie brano pod uwagę błędów w rozróżnianiu mowy słabo i silnie dźwięcznej. Błędy dyskryminacji dźwięczna/ bezdźwięczna (V/UV) wynosiły 10.8%.

Dyskryminatory liniowe

W oparciu o takie same parametry, jakie były wykorzystane w dyskryminatorze progowym, skonstruowano dyskryminator liniowy według algorytmu Fisher'a [207]. Zamiast wartości E_{sig} i T_{sp} wykorzystano stosunek $L_v = \sqrt{E_{sig}/T_{sp}}$. Ramka mowy jest określona jako dźwięczna, gdy wartość funkcji dyskryminującej

$$0.0507L_v + 17.62r_{max} + 2.16L_{lp} - 0.767L_{hp} - 5.59L_{res} \quad (5.16)$$

przekracza próg 6.8. Testowanie na 10 frazach wykorzystanych do obliczania funkcji dyskryminującej i progu daje w wyniku 5.6% błędów (Tab.5.1). Znaczenie poszczególnych parametrów (ich "moc dyskryminująca") nie jest jednakowe. Aby zbadać ten problem skonstruowano szereg dyskryminatorów Fisher'a wykorzystujących różne podzbiory parametrów. Wyniki przedstawiono w Tab.5.1. Najważniejszym parametrem okazała się znormalizowana wartość autokorelacji r_{max} , obliczana w zakresie 2-20 ms z dolnopasmowo odfiltrowanego błędu predykcji. Parametry L_v i L_{hp} mają najmniejsze znaczenie i mogą być usunięte. Ostatecznie wyznaczono dyskryminator wykorzystujący parametry r_{max} , L_{lp} i L_{res} i ramka mowy jest określana jako dźwięczna, jeżeli funkcja

$$18.34r_{max} + 2.54L_{lp} - 5.68L_{res} \quad (5.17)$$

przekracza próg 7.3.

Zaproponowany dyskryminator V/UV daje 5.6% błędów w procesie przetwarzania 10 testowych fraz. Dla innych fraz, nie wchodzących do zbioru testowego, otrzymywano od 0%

Tabela 5.1: Porównanie dyskryminatorów V/UV dla różnych podzbiorów parametrów

parametr	1-wykorzystany, 0-pominięty							
L_v	1	1	1	1	1	0	0	0
r_{max}	1	1	1	1	0	1	1	1
L_{lp}	1	1	1	0	1	1	0	1
L_{hp}	1	1	0	1	1	1	0	0
L_{res}	1	0	1	1	1	1	0	1
%błędów	5.6	6	5.6	6.3	14	5.6	8.2	5.6

do 16% błędów (średnio 6.4%). Należy podkreślić, że niemal wszystkie błędy występują w obszarach transjentowych, gdzie nawet klasyfikacja "ręczna" niekiedy zawodzi.

Dla odróżnienia głosek słabo dźwięcznych od silnie dźwięcznych wprowadzono drugą wartość progową dla funkcji dyskryminującej (wzór (5.17)). Ramka mowy jest klasyfikowana jako silnie dźwięczna, gdy funkcja (5.17) przekracza wartość 12.

Dyskryminator stacjonarna/plozyjna dla mowy bezdźwięcznej

Badanie przeprowadza się dla ramek sygnału uprzednio sklasyfikowanych jako mowa bezdźwięczna. W pierwszym etapie skonstruowano dyskryminator liniowy Fisher'a przetwarzający następujące parametry: stosunek $E_{sig}(i)/E_{sig}(i-1)$, oraz L_{lp} , L_{hp} , L_v i $r_{max/rms}$. Opracowany dyskryminator dawał 21% błędów. Nie jest to wartość zadowalająca, należy jednak zwrócić uwagę na fakt, że wpływ tego typu błędów na jakość sygnału wokodera jest znacznie mniejszy niż wpływ błędów w klasyfikacji dźwięczna/ bezdźwięczna. W toku dalszych badań ustalono, że główną rolę w procesie dyskryminacji mowa stacjonarna/plozyjna odgrywa parametr $r_{max/rms}$ - stosunek wartości maksymalnej do wartości skutecznej sygnału błędu predykcji (rys.5.7). Po jego usunięciu ze zbioru przetwarzanych parametrów (i wyznaczeniu nowej funkcji dyskryminującej) liczba błędów wzrosła z 21% do 36%. Z kolei usunięcie wszystkich pozostałych parametrów prowadzi do zwiększenia liczby błędów do 22%.

Ostatecznie proponuje się następujący dyskryminator: ramka jest klasyfikowana jako plozyjna gdy

$$r_{max/rms} > 4.6 \quad (5.18)$$

Stosunkowo duża liczba błędów (22%) nie musi prowadzić do obniżenia jakości wokodera. Wystarczy wspomnieć, że w klasycznym wokoderze w ogóle nie przewiduje się osobnego pobudzenia dla głosek plozyjnych - wszystkie głoski bezdźwięczne są generowane z wykorzystaniem generatora szumu.

5.3 Zmodyfikowany wokoder predykcyjny o przepływności 2.4 kbit/s

5.3.1 Ogólna charakterystyka wokodera

Dla przepływności binarnej 2.4 kbit/s można zastosować kilka wariantów koderów predykcyjnych. Możliwa jest adaptacja kodera CELP, zasadniczo przeznaczonego dla większych przepływności binarnych, do przepływności 2.4 kbit/s. Wstępne badania wykazały, że adaptacja taka nastrecza wiele trudności. Przede wszystkim należy się liczyć z pogorszeniem jakości mowy w porównaniu z koderami CELP o większych przepływnościach binarnych, np. 1016-CELP o przepływności 4.8 kbit/s [12]. Pogorszeniu jakości można częściowo przeciwdziałać, stosując kwantowanie parametrów w wielu ramkach jednocześnie z wykorzystaniem interpolacji. Takie rozwiązanie wiąże się ze znacznym wzrostem opóźnienia, a także złożoności obliczeniowej.

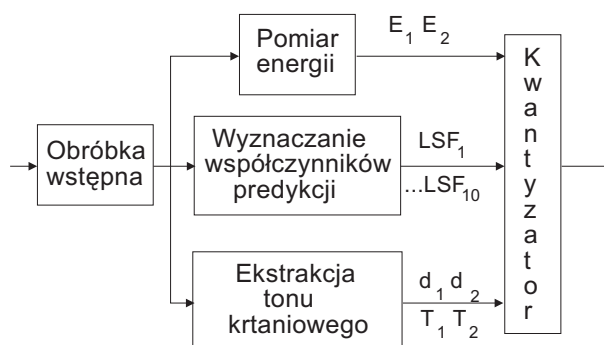
Innym rozwiązaniem jest zastosowanie klasycznego wokodera predykcyjnego, np. opartego na standardzie amerykańskim LPC10 [198]. Zaletą takiego rozwiązania jest prostota, wadą - przeciętna jakość sygnału mowy. Standard LPC10 został opracowany w początku lat 80-tych i był modyfikowany (LPC10e), a w końcu został zastąpiony nowym standardem MELP (*Mixed Excitation Linear Prediction*) [191]. Dzięki odpowiedniej modyfikacji sygnału pobudzającego filtr predykcyjny udało się uzyskać jakość mowy bardzo niewiele ustępującą jakości osiąganą w koderze 1016 CELP o dwa razy większej przepływności binarnej.

Opisany w niniejszej pracy wokoder predykcyjny 2.4 kbit/s jest częściowo oparty na wcześniejszej wersji kodera MELP [135], która została zmodyfikowana i opisana w pracy [54]. Obecna wersja zawiera dalsze istotne modyfikacje. Poprawiono ekstraktor tonu krtaniowego, opierając jego działanie na dyskryminatorze liniowym Fishera [50]. W ekstraktorze rozpoznawane są 4 klasy sygnałów (cisza z mową bezdźwięczną o charakterze stacjonarnym, bezdźwięczne głoski płozyjne, mowa słabo dźwięczna, mowa silnie dźwięczna). Ekstraktor jest uruchamiany 2 razy na ramkę (długość ramki wynosi 30 ms, czyli 240 próbek). Poprawiono również algorytm predykcji - obecnie oblicza się i kwantuje współczynniki tzw. widma prążkowego (LSF - *Line Spectrum Frequencies*). Omawiany wokoder cechuje opóźnienie algorytmiczne 1.5 ramki, czyli 45 ms (jest to opóźnienie związane z przetwarzaniem sygnału w nadajniku - nie uwzględnia czasu transmisji skwantowanych parametrów).

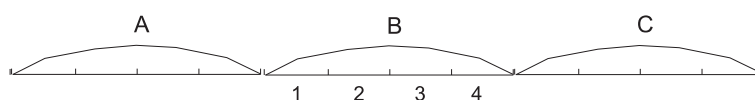
Zmodyfikowany wokoder jest częścią kodera o przepływnościach binarnych 2.4/4.8/8 kbit/s, opisanego w [55, 66] i w p.3.4.7 niniejszej pracy.

5.3.2 Nadajnik wokodera predykcyjnego 2.4 kbit/s

Schemat nadajnika wokodera pokazano na rys.5.8. Zawiera on bloki pomiaru parametrów sygnału mowy. Przed pomiarem sygnał poddawany jest obróbce wstępnej, mającej na celu odfiltrowanie składowej stałej i przydźwięku sieci, korekcję charakterystyki częstotliwościowej mikrofonu, oraz automatyczną regulację poziomu sygnału. Zagadnienia te mają charakter



Rysunek 5.8: Schemat blokowy nadajnika wokodera



Rysunek 5.9: Rozmieszczenie kolejnych ramek analizy predykcyjnej (oznaczonych dużymi literami) względem wektorów pobudzenia (oznaczonych cyframi)

ściśle techniczny i zostały opisane w [65]. W każdej ramce dokonuje się obliczenia parametrów predykcyjnych, tzw. współczynników widma prążkowego ($LSF_1, LSF_2 \dots, LSF_{10}$). Dwa razy częściej wykonywany jest pomiar energii (E_1, E_2) oraz ekstrakcja tonu krtaniowego według algorytmu opisanego w p.5.2 (wynik klasyfikacji d_1, d_2 , okres tonu krtaniowego T_1, T_2).

Podział sygnału na ramki analizy predykcyjnej jest zawsze pewnym kompromisem. Używanie dobrej rozdzielczości w dziedzinie częstotliwości wymaga stosowania długich ramek, rzędu 50-100 ms, natomiast struktura czasowa sygnału mowy narzuca znacznie krótsze ramki, około 10 ms (taki jest czas trwania krótkich głosek). W projektowanym koderze przyjęto za podstawową długość ramki 30 ms (240 próbek). W licznych rozwiązaniach wokoderów stosowana jest interpolacja parametrów filtrów predykcyjnych. W omawianym wokoderze wykonuje się ją co 7.5 ms. Implikuje to podział sygnału pobudzającego w ramce na 4 wektory o długości 60 próbek. Rozmieszczenie ramek i wektorów na skali czasu pokazano na rys.5.9. Wektorowi nr 1 odpowiada filtr predykcyjny, będący interpolacją filtrów A i B ($\frac{1}{3}A$ i $\frac{2}{3}B$). Wektorowi nr 2 i 3 odpowiada filtr predykcyjny B. Z kolei wektorowi nr 4 odpowiada filtr, będący interpolacją $\frac{2}{3}A$ i $\frac{1}{3}C$.

Osobnym zagadnieniem jest wybór parametrów opisujących transmitancję filtru predykcyjnego i wykorzystywanych w procesie interpolacji. Badano współczynniki odbicia (k_i) i współczynniki widma prążkowego (LSF). Te ostatnie dawały zawsze lepsze rezultaty (bardziej płynne zmiany charakterystyki filtru spowodowane interpolacją współczynników). Współczynniki LSF oblicza się w następujący sposób. Ze współczynnikami predykcji a_1, a_2, \dots, a_{10} związany jest filtr inwersyjny o transmitancji:

$$A(z) = \sum_{i=0}^{10} a_i z^{-i}, \quad a_0 = 1 \quad (5.19)$$

i odpowiedzi impulsowej $a_0, a_1, \dots, a_{10}, 0, 0, \dots$. Dodając i odejmując od $A(z)$ transmitancję filtru o odpowiedzi impulsowej

$$0, a_{10}, a_9, \dots, a_0, 0, 0, \dots$$

otrzymuje się odpowiednio filtry $\tilde{P}(z)$ i $\tilde{Q}(z)$. Można wykazać, że wszystkie zera transmitancji $\tilde{P}(z)$ i $\tilde{Q}(z)$ leżą na okręgu jednostkowym i to na przemian: $z = 1$ jest miejscem zerowym $\tilde{Q}(z)$, $z = \exp(j\omega_1)$ miejscem zerowym $\tilde{P}(z)$, $z = \exp(j\omega_2)$, $\omega_2 > \omega_1$ miejscem zerowym $\tilde{Q}(z)$, itd., aż do $z = -1$, które jest miejscem zerowym $\tilde{P}(z)$. Ze względu na symetrię, zera leżące w górnej części płaszczyzny zespolonej są wystarczające do odtworzenia transmitancji obu filtrów. Parametry $\omega_1, \omega_2, \dots, \omega_{10}$ tworzą współczynniki widma prążkowego $LSF_1, LSF_2, \dots, LSF_{10}$.

Miejsca zerowe transmitancji są wyznaczane metodą próbkowania transmitancji $\tilde{P}(z)$ i $\tilde{Q}(z)$ na okręgu jednostkowym (256 punktów rozłożonych od 0 do π). Po skwantowaniu są one transmitowane do odbiornika, gdzie odtwarza się współczynniki wielomianów $\tilde{P}(z)$ i $\tilde{Q}(z)$ metodą mnożenia dwumianów. W końcu odtwarza się współczynniki predykcji a_1, a_2, \dots, a_{10} korzystając z zależności

$$A(z) = \frac{1}{2} [\tilde{P}(z) + \tilde{Q}(z)] \quad (5.20)$$

W wokoderze 2.4 kbit/s zastosowano schemat kodowania parametrów w ramce, przedstawiony w tab.5.2. Po dodaniu 1 bitu na ramkę przepływność osiąga wartość nominalną.

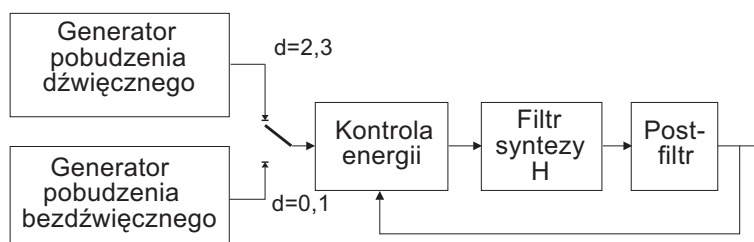
Tabela 5.2: Schemat kodowania parametrów w 30 ms ramce - przepływność 2.367 kbit/s

parametr	liczba bitów
LSF_1, \dots, LSF_{10}	4,5,5,4,4,4,4,3,3
klasyfikator d	2+2
T_0	7+7
energia	6+6
synchronizacja	1
razem	71

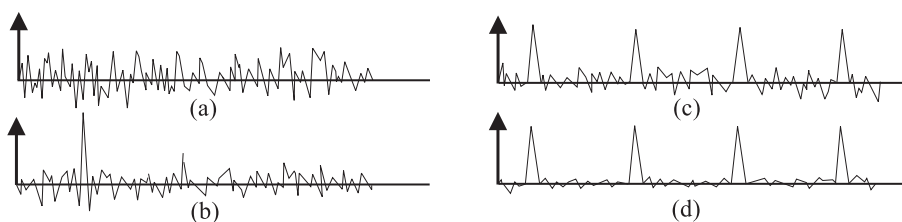
5.3.3 Odbiornik wokodera predykcyjnego

Schemat blokowy odbiornika wokodera pokazano na rys.5.10. Wykorzystuje się w nim osobne generatory pobudzenia dla mowy dźwięcznej i bezdźwięcznej. Dla ciszy i mowy bezdźwięcznej o charakterze stacjonarnym pobudzenie ma charakter szumowy (rys.5.11a). Dla mowy bezdźwięcznej o charakterze plozyjnym do szumu dodawane są impulsy aperiodyczne (rys.5.11b).

Dla mowy dźwięcznej pobudzenie ma charakter mieszany: składa się z odfiltrowanych dolnopasmowo impulsów o okresie powtarzania równym okresowi tonu krtaniowego oraz z



Rysunek 5.10: Schemat blokowy odbiornika wokodera



Rysunek 5.11: Sygnały pobudzające: (a) - cisza i mowa bezdźwięczna o charakterze stacjonarnym, (b) - mowa bezdźwięczna o charakterze plosyjnym, (c) - mowa słabo dźwięczna, (d) - mowa silnie dźwięczna

odfiltrowanego górnopasmowo szumu. Zawartość obu składników w sygnale pobudzającym jest regulowana współczynnikiem wzmocnienia szumu g . Nieskorelowany szum jest przepuszczany przez filtr o transmitancji

$$G_{hp}(z) = \frac{g}{1 + g^2} (1 - z^{-1}) \quad (5.21)$$

Jednocześnie periodyczny składnik impulsowy jest przepuszczany przez filtr

$$G_{lp}(z) = \frac{1}{1 + g^2} (1 + g^2 z^{-1}) \quad (5.22)$$

Po dodaniu do siebie obu odfiltrowanych składników otrzymuje się sygnał o płaskiej obwiedni gęstości mocy. Wynika to z faktu, że dla każdego wzmocnienia g i dla każdej częstotliwości (tzn. dla każdego $z = \exp(j\omega)$) spełniony jest warunek $|G_{hp}|^2 + |G_{lp}|^2 = 1$. Wzmocnienie szumu g zależy od stopnia dźwięczności ($d = 2$ - rys.5.11c lub $d = 3$ - rys.5.11d) oraz od mocy sygnału (ciche sygnały są bardziej zaszumione).

Ściślej biorąc, wzmocnienie to zależy od stosunku mocy sygnału w danej półramce do poziomu mowy dźwięcznej L_v . Poziom mowy dźwięcznej jest wyznaczany 2 razy w każdej ramce na podstawie transmitowanej do odbiornika mocy sygnału E_{sig} . Jeżeli dana półramka jest sklasyfikowana jako dźwięczna ($d = 2$ lub $d = 3$), to poziom mowy dźwięcznej określa się ze wzoru:

$$L_v(i) = \begin{cases} E_{sig}(i) & \text{gdy } E_{sig}(i) > L_v(i-1) \\ 0.98L_v(i-1) & \text{gdy } E_{sig}(i) \leq L_v(i-1) \end{cases} \quad (5.23)$$

Jeżeli dana półramka jest sklasyfikowana jako bezdźwięczna, to

$$L_v(i) = 0.98L_v(i-1) \quad (5.24)$$

Po wyznaczeniu L_v oblicza się stosunek E_{sig}/L_v dla mowy dźwięcznej i wyraża w decybelach:

$$E_v = 10 \log_{10} \frac{E_{sig}}{L_v} \quad (5.25)$$

Wzmocnienie szumu zmienia się od $g = 1$ dla cichego sygnału, do $g = 0.2$ dla sygnału głośnego:

$$g = \begin{cases} -\frac{1}{15} E_v - 0.2, & \text{gdy } -18 < E_v < -6 \\ 1, & \text{gdy } E_v \leq -18 \\ 0.2, & \text{gdy } E_v \geq -6 \end{cases} \quad (5.26)$$

Jeżeli mowa jest słabo dźwięczna ($d = 2$), to dodawane jest niewielkie (10%) przesunięcie (ang. *jitter*) położenia impulsów. Powoduje ono lepsze odtwarzanie głosek "ż,ź" itp.

Sygnał pobudzający jest podawany do filtru syntezy o transmitancji:

$$H(z) = \frac{const}{A(z)} = \frac{const}{\sum_{i=0}^p a_i z^{-i}} \quad (5.27)$$

W celu zwiększenia stałości parametrów filtru (w procesie przetwarzania sygnałów stacjonarnych) i zmniejszenia jego wzmocnienia wprowadza się niewielkie tłumienie rezonansów formantowych, czyli poszerzenie ich pasma. Filtr syntezy z tłumieniem ma transmitancję

$$H(z) = \frac{const}{A(z/\gamma')} = \frac{const}{\sum_{i=0}^p a_i (\gamma')^i z^{-i}} \quad (5.28)$$

gdzie $\gamma' \approx 0.995$.

Postfiltr (rys.5.10) ma za zadanie zwiększenie energii sygnału w zakresach formantowych, aby lepiej zamaskować szum kwantyzacji w zakresach międzyformantowych. Jest to rodzaj filtru percepcyjnego o transmitancji $W(z)$, który w pewnym stopniu zniekształca sygnał użyteczny (po stronie nadawczej nie jest stosowana filtracja odwrotna z wykorzystaniem filtru o transmitancji $1/W(z)$). Transmitancja typowego postfiltru jest następująca (patrz wzór (4.70)¹):

$$W(z) = (1 - \mu z^{-1}) \frac{A(z/\gamma_1)}{A(z/\gamma_2)}, \quad \gamma_1 < \gamma_2 < 1, \quad \mu < 1$$

W wyniku badań symulacyjnych przyjęto $\gamma_1 = 0.55$, $\gamma_2 = 0.8$, $\mu = 0.3$. Postfiltr wprowadza najczęściej wzmocnienie mocy sygnału - konieczna jest odpowiednia korekcja wzmocnienia.

Przedstawiony wokoder o przepływności 2.4 kbit/s stanowi część kodera o przełączanej przepływności binarnej, opisanego w p.3.4.7. Do implementacji algorytmów wokodera wybrano zmiennoprzecinkowy procesor sygnałowy TMS320C31 firmy Texas Instruments. Jego

¹Charakterystyka częstotliwościowa postfiltru, oczywiście w zakresie częstotliwości właściwym dla sygnału telefonicznego, jest zbliżona do charakterystyki filtru $1/W(z)$ z rys.4.8 - ze względu na inne wartości współczynników γ_1 i γ_2

możliwości obliczeniowe (rzędu 20 Mflops) całkowicie wystarczają do realizacji tego zadania. Najbardziej złożoną obliczeniowo część programu stanowi algorytm wyznaczania autokorelacji, uruchamiany 2 razy na ramkę wraz z ekstraktorem tonu krtaniowego.

Ważnym problemem jest redukcja zniekształceń wywołanych przekłamaniami w kanale. Szczególnie ma to znaczenie w procesie dekodowania współczynników widma LSF. Prawidłowo obliczone i zakodowane współczynniki są liczbami rosnącymi. Naruszenie tego warunku (np. skutek przekłamań w kanale) powoduje utratę stabilności filtru syntezy i dokuczliwe zniekształcenia. W związku z tym, po zdekodowaniu współczynników LSF bada się ich monotoniczność. Gdy nie tworzą one funkcji ściśle rosnącej, to podstawia się współczynniki z poprzedniej ramki (lub jeszcze wcześniejszej, jeśli współczynniki z poprzedniej ramki również były "uszkodzone").

Wyniki pomiaru wyrazistości logatomowej podano w tab.3.10. Mają one znaczenie jedynie porównawcze - w trakcie przeprowadzania odsłuchów nie dotrzymano wszystkich warunków określonych w normie. Osiągnięta 50% wyrazistość jest w tej sytuacji zaniżona (podobnie jak inne wyniki w tab.3.10). Przeprowadzono wiele nieformalnych testów, polegających na nawiązywaniu łączności z wykorzystaniem wokodera. Jakość mowy i przydatność wokodera była oceniana pozytywnie.

5.4 Wokoder 1.2 kbit/s

5.4.1 Wokoder radiowy

Opisany w tym punkcie wokoder ma zmniejszoną przepływność binarną: 1.2 kbit/s, ze względu na współpracę z modemem krótkofalowym. Wraz z opisanym w poprzednim punkcie wokoderem 2.4 kbit/s oraz modemem jest elementem wokodera radiowego - urządzenia do przesyłania mowy w postaci cyfrowej lub cyfrowych danych przez kanał krótkofalowy [65, 210].

Technika transmisji cyfrowej przez łącza krótkofalowe różni się zasadniczo od technik transmisji cyfrowej poprzez inne łącza radiowe. Zakres fal krótkich przysparza wiele kłopotów związanych z dyspersyjnym charakterem tego środowiska oraz dużą liczbą występujących tam zakłóceń addytywnych. Transmisję cyfrową przez tradycyjne analogowe radiostacje KF wykonuje się za pomocą specjalnych modemów radiowych. Metody transmisji stosowane w modemach radiowych dla pasma KF cechuje duża odporność zarówno na zjawiska występujące w kanale radiowym, jak i na zniekształcenia wnoszone przez radiostacje (np. duży błąd częstotliwości, dochodzący nawet do ± 100 Hz).

W opisanym w [210] modemie radiowym KF wykorzystano równoległą modulację QPSK o 39 nośnych. Każda z nośnych jest modulowana 4-wartościową modulacją fazy. Odstęp jednostkowy modulacji wynosi 30 ms. Maksymalna szybkość transmisji wynosi 2.6 kbit/s. Modem został zrealizowany (przez dr P. Zwierkę) w oparciu o procesor sygnałowy TMS320C31,

pracujący z zegarem 40 MHz.

Wokoder (zrealizowany przez autora niniejszej rozprawy) przekształca sygnał mowy na strumień binarny o przepływności 1.2 lub 2.4 kbit/s, który w procesie kodowania nadmiarowego jest uzupełniany do przepływności 2.6 kbit/s. Podobnie moduł transmisji danych umożliwia przesyłanie danych z efektywną przepływnością 1.2 kbit/s. Zabezpieczenie przesyłanych danych przed błędami jest wykonywane w oparciu o kodowanie nadmiarowe kodem splotowym i przeplot czasowy.

Koder 1.2 kbit/s oparty jest na interpolacji i jednoczesnym kwantowaniu parametrów sygnału w 4 kolejnych, 30-milisekundowych ramkach. Oznacza to zwiększenie opóźnienia algorytmicznego (jest to opóźnienie związane z przetwarzaniem sygnału w nadajniku, nie uwzględnia czasu transmisji skwantowanych parametrów) - do 4.5 ramki czyli 135 ms. Opóźnienie w transmisji jest równe czasowi trwania 4 ramek i wynosi 120 ms. Zwiększone opóźnienie jest ceną, którą trzeba zapłacić za zmniejszenie przepływności binarnej bez nadmiernego pogorszenia jakości mowy. Jednoczesne kodowanie kilku kolejnych ramek (od 3 do 8) jest rozwiązaniem często stosowanym w wokoderach o przepływnościach niższych od 2 kbit/s [150].

5.4.2 Interpolacja i kwantyzacja transmitowanych parametrów

Tabela 5.3: Schemat kodowania parametrów - przepływność 1.2 kbit/s

parametr	liczba bitów na 4 ramki (120 ms)
LSF_1, \dots, LSF_{10}	$2 \cdot 34$
strategia interpolacji LSF	3
klasyfikator d	$8 \cdot 2$
wartość średnia T_0	7
3 przyrosty T_0	$3 \cdot 4$
energia	$6 + 7 \cdot 4$
synchronizacja	4
razem	144

W wokoderze 1.2 kbit/s zastosowano następujący schemat kodowania parametrów w 4 kolejnych ramkach (Tabl.5.3):

- Parametry predycyjne LSF_1, \dots, LSF_{10} są kwantowane odpowiednio w 3,4,4,4,4,3,3,3,3,3 bitach - razem 34 bity. Kwantowane są 2 ramki na 4 - parametry w pozostałych ramkach są interpolowane. Wybór ramek przeznaczonych do kwantowania odbywa się na zasadzie minimalizacji ważonego błędu rekonstrukcji parametrów predycyjnych w 4 ramkach. Analizowane są następujące strategie (1 oznacza kwantowaną ramkę - 0 oznacza interpolowaną ramkę): 1100, 1010, 1001, 0110, 0101, 0011. Sposób przeprowadzania interpolacji zależy od klasyfikacji ramki - nie interpoluje się ramek pozytywnych ($d = 1$);

- Klasyfikator sygnału d , przyjmujący wartości 0,1,2,3, kwantowany jest w 2 bitach 2 razy na ramkę (tzn. 8 razy na 4 ramki);
- Okres tonu krtaniowego T_0 kwantowany jest w następujący sposób: Wartość średnia w 4 ramkach kodowana jest w 7 bitach, 3 przyrosty kodowane są w 4 bitach;
- Energia sygnału jest kwantowana w 6 bitach w pierwszej półramce odniesienia. Energia w pozostałych 7 półramkach jest kwantowana w 4 bitach.

Po uwzględnieniu synchronizacji (4 bity na 4 ramki) otrzymuje się 144 bity na 4 ramki, co odpowiada szybkości transmisji 1.2 kbit/s.

5.4.3 Realizacja i testowanie wokodera

Odbiornik wokodera 1.2 kbit/s nie różni się zasadniczo od odbiornika wokodera 2.4 kbit/s.

Do testowania wokodera wykorzystano szereg fraz sygnału mowy, o czasie trwania od kilku do kilkudziesięciu sekund, wypowiedzianych przez różnych mówców w różnych warunkach. Wykorzystano również testowe frazy amerykańskie, obejmujące różne głosy męskie i żeńskie. Wstępne wyniki są pozytywne, jednak korzystne byłoby zorganizowanie sformalizowanych odsłuchów.

Przetestowano również modem krótkofalowy. Dla lepszych jakościowo kanałów radiowych wykorzystuje się przepływność 2.4 kbit/s, w zasadzie bez żadnego zabezpieczenia przed błędami. Badania dowiodły, że zastosowany wokoder może pracować bez zbyteńnego pogorszenia zrozumiałości przy stopie błędów BER równej nawet 10^{-2} . Dla kanałów radiowych o gorszych parametrach urządzenie oferuje opcję kodowania nadmiarowego kanału kodem splotowym o $R = 1/2$ wraz z dekodерem Viterbiego oraz wokoderem o przepływności 1.2 kbit/s.

Prototyp urządzenia zawierającego opisywany modem KF został poddany eksperymentom laboratoryjnym i praktycznym (przez firmę TEL-KA i Wojsk. Inst. Łączności) zarówno przy pracy na fali przyziemnej, jak i odbitej przy odległościach kilkuset kilometrów. Stwierdzono, że do poprawnej pracy urządzenia w trybie mowy wystarczy stosunek $S/N = 6\text{dB}$ (przy wokoderze 1.2 kbit/s i wydłużonej preambule synchronizacyjnej). Modem toleruje (bez utraty synchronizacji) nawet kilkunastosekundowe zaniki sygnału. Również błąd częstotliwości sięgający 50 Hz nie ma wpływu na pracę modemu.

Przeprowadzone badania miały charakter nieformalny i porównawczy - wykazały jednak dobrą współpracę wokodera predykcyjnego z modemem radiowym i przydatność całego zestawu w radiokomunikacji krótkofalowej.

5.5 Zastosowanie ekstraktora tonu krtaniowego w kodерze CELP

W toku prac mających na celu realizację koderów CELP 4.8 i 8 kbit/s [66] przeprowadzono badania symulacyjne wariantów o zmniejszonej złożoności obliczeniowej. W tym celu do

koderów CELP, opisanych w p.3.4.7, wprowadzono istotne modyfikacje. Przede wszystkim dodano uproszczony ekstraktor tonu krtaniowego, opierając jego działanie na dyskryminatorze liniowym Fishera [50]. Ekstraktor ten jest uruchamiany 2 razy na ramkę (ramka obejmuje 30 ms, czyli 240 próbek sygnału). W ekstraktorze tonu krtaniowego określa się dźwięczność sygnału d i estymuje się okres tonu krtaniowego T_0 . Dźwięczność jest określana jednobitowo i transmitowana do odbiornika. Informacji o okresie tonu krtaniowego nie przesyła się, wykorzystuje się ją jednak do przyspieszenia przeszukiwania słownika adaptacyjnego. Parametry te są przydatne do konstruowania sygnału pobudzającego filtr predykcyjny (d) i do przyspieszenia algorytmu wyznaczania predyktora długookresowego (T_0).

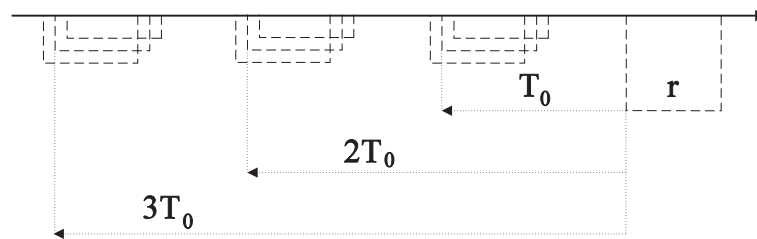
W zaprojektowanym koderze (odbiornik działa według schematu z rys.1.1) stosuje się słownik stochastyczny o specjalnej strukturze, umożliwiającej jego szybką filtrację i przeszukiwanie - rys.3.14. Jest to struktura "rzadka" (wiele składowych zerowych) i "nakładkowa" (poszczególne wektory są "wycinane" z długiej sekwencji próbek z przesunięciem o 1 lub 2 próbki). Słownik adaptacyjny ma strukturę "nakładkową" (poszczególne wektory stanowią wycinki sekwencji próbek pobudzenia wygenerowanego w bezpośredniej przeszłości), jednak nie jest "rzadki". Z tego względu przeszukiwanie słownika adaptacyjnego jest procesem bardziej złożonym obliczeniowo. Słownik ten ma istotne znaczenie w procesie kodowania mowy dźwięcznej (predykcja długookresowa wykorzystuje korelacje próbek odległych o okres tonu krtaniowego lub jego wielokrotność). Dla mowy bezdźwięcznej nie ma on istotnego znaczenia, dlatego w celu zmniejszenia złożoności obliczeniowej tworzy się go w sposób identyczny jak słownik stochastyczny (rys.3.14), wypełniając "rzadką" sekwencją przypadkową.

Z kolei dla mowy dźwięcznej stosowane jest tzw. selektywne przeszukiwanie słownika adaptacyjnego. Proponowany algorytm selektywnego przeszukiwania słownika adaptacyjnego opiera się na fakcie, że w procesie kodowania mowy dźwięcznej wybierane są wektory opóźnione względem tworzonego wektora pobudzenia o okres tonu krtaniowego T_0 lub jego wielokrotność. Należy zatem dokonać wstępnej oceny okresu tonu krtaniowego i wybrać odpowiedni podzbiór wektorów słownika adaptacyjnego.

Zaproponowana metoda będzie korzystna, jeżeli dysponuje się szybkim algorytmem pomiaru okresu tonu krtaniowego. Metoda autokorelacyjna wykorzystywana w wokoderze predykcyjnym jest nazbyt złożona obliczeniowo (samo obliczanie autokorelacji wymaga około 1 mln operacji na sekundę). W implementowanym koderze CELP zastosowano metodę korelacji wzajemnej, którą można opisać w następujący sposób:

1. Znalezienie największej pod względem modułu próbki sygnału mowy w buforze obejmującym 300 próbek (niemal całą bieżącą ramkę x_0, x_1, \dots, x_{229} i 70 próbek poprzedniej ramki x_{-70}, \dots, x_{-1}). Brane są pod uwagę jedynie maksima właściwe (sąsiednie próbki muszą być mniejsze). Określa się położenie maksimum globalnego i_{max} ;
2. Obliczenie wartości współczynników korelacji wzajemnej między sygnałem mowy a 10-próbkowym otoczeniem próbki o indeksie i_{max} :

$$R_n = \sum_{i=i_{max}-10}^{i_{max}+10} x_i x_{i-n} \quad (5.29)$$



Rysunek 5.12: Selektywne przeszukiwanie słownika adaptacyjnego

Korelację oblicza się dla przesunięć odpowiadających zakresowi zmian okresu tonu krtaniowego $n = \pm 20, \pm 21, \dots, \pm 150$, czyli od 2.5 ms do 18.75 ms. Jeżeli $i_{max} - n$ leży poza granicami 300-próbkowego bufora, to współczynnika korelacji nie oblicza się;

3. Znalezienie maksimum właściwego korelacji wzajemnej, którego położenie odpowiada okresowi tonu krtaniowego T_0 lub jego wielokrotności;
4. Eliminacja wielokrotności T_0 przez sprawdzenie wartości korelacji wzajemnej dla przesunięć 2, 3, ..., 7-krotnie mniejszych od położenia maksimum globalnego (jednak nie mniejszych niż dolna granica równa 20) i porównanie z wartością maksimum globalnego znalezione w poprzednim punkcie.

Algorytm ten jest znacznie mniej złożony obliczeniowo niż algorytm autokorelacyjny - obliczanie korelacji wzajemnej dla każdej ramki sygnału wymaga ok. 0.2 mln operacji na sekundę.

Po oszacowaniu okresu tonu krtaniowego T_0 dokonuje się wyboru podzbioru wektorów ze słownika adaptacyjnego. Przetestowano 2 algorytmy:

1. Przeszukiwanie wektorów odpowiadających opóźnieniom równym $T_0 \pm 5$ próbek, w tym wektorów odpowiadających opóźnieniom ułamkowym (co $\frac{1}{2}$ próbki);
2. Przeszukiwanie wektorów odpowiadających opóźnieniom równym $T_0 \pm 5, 2T_0 \pm 5, 3T_0 \pm 5, 4T_0 \pm 5$ próbek (nie wykraczając poza górną granicę równą 150).

Lepsze wyniki dał ten drugi algorytm (rys.5.12). W najgorszym przypadku przeszukuje się podzbiór 44 wektorów (zamiast 130). Przeprowadzając badania symulacyjne nie zaobserwowano pogorszenia jakości sygnału w wyniku ograniczenia liczby testowanych wektorów. Wyniki symulacji kodera CELP z ekstraktorem tonu krtaniowego były pozytywne, jednak okazało się, że na procesorze sygnałowym TMS320C31 z zegarem 40 MHz możliwa jest realizacja kodera w wersji podstawowej (opisanej w p.3.4.7). Ostatecznie zdecydowano się na realizację wersji podstawowej.

Dodatek A

Oznaczenia

- $x(n)$ - n -ta (bieżąca) próbka sygnału akustycznego
 $\hat{x}(n)$ - próbka sygnału predykcji
 $\epsilon(n) = x(n) - \hat{x}(n)$ - próbka sygnału różnicowego (błędu predykcji)
 $\tilde{\epsilon}(n)$ - skwantowany sygnał różnicowy
 $e(n)$ - błąd kwantyzacji
 $\tilde{x}(n)$ - próbka sygnału wyjściowego dekodera (sygnału zrekonstruowanego)
 $\sigma_x^2, \sigma_\epsilon^2, \sigma_e^2$ - moce sygnałów $x(n), \epsilon(n), e(n)$
 $[\sigma_\epsilon^p]^2$ - moc sygnału błędu predykcji $\epsilon(n)$ dla predyktora rzędu p
 $G_p = \frac{\sigma_x^2}{\sigma_\epsilon^2}$ - zysk predykcji (dla predyktora rzędu p)
 $SNR_q = \frac{\sigma_\epsilon^2}{\sigma_e^2}$ - stosunek mocy sygnału wejściowego kwantyzatora do szumu kwantyzacji
 $SNR = \frac{\sigma_x^2}{\sigma_e^2}$ - stosunek mocy sygnału akustycznego do szumu kwantyzacji
 $\mathbf{a} = \mathbf{a}^p = [a_1, \dots, a_p]^t$ - wektor współczynników predykcji
 t - transpozycja
 $\mathbf{a}(n)$ - wektor współczynników predykcji w chwili n
 p - rząd predyktora
 $\tilde{\mathbf{x}}(n-1) = [\tilde{x}(n-1), \dots, \tilde{x}(n-p)]^t$ - wektor poprzednich próbek sygnału zrekonstruowanego
 $\Delta \mathbf{a}(n)$ - poprawka wektora współczynników predykcji w chwili n
 L - liczba poziomów kwantyzacji kwantyzatora skalarnego lub liczba wektorów słownika kwantyzatora wektorowego
 b - rozdzielczość (liczba bitów na próbkę) kwantyzatora
 y_i - i -ty poziom kwantyzacji
 x_i - granica przedziałów kwantyzacji
 $p(\epsilon)$ - gęstość prawdopodobieństwa dla próbek sygnału $\epsilon(n)$
 $\Delta, \Delta(n)$ - odstęp sąsiednich poziomów kwantyzacji w kwantyzatorze równomiernym
 Δ_{opt} - odstęp sąsiednich poziomów kwantyzacji w optymalnym kwantyzatorze równomiernym

- $I(n)$ - numer poziomu kwantyzacji wykorzystany w chwili n
 $M(1), \dots, M(L/2)$ - współczynniki adaptacji kwantyzatora
 $\bar{y}_i, \bar{x}_i, \bar{\Delta}_{opt}$ - wartości y_i, x_i, Δ_{opt} dla sygnału o jednostkowej mocy
 $\hat{\sigma}_\epsilon(n)$ - estymata wartości skutecznej sygnału $\epsilon(n)$
 $P(i)$ - prawdopodobieństwo wystąpienia i -tego poziomu kwantyzacji
 α - szybkość adaptacji kwantyzatora
 k_1^f, \dots, k_p^f - współczynniki predykcji filtru kratowego (tzw. współczynniki odbicia w przód)
 k_1^b, \dots, k_p^b - współczynniki predykcji filtru kratowego (tzw. współczynniki odbicia wstecz)
 β - szybkość adaptacji predyktora w metodzie stochastycznego gradientu
 L_β - szybkość adaptacji predyktora w metodzie stochastycznego gradientu z normalizacją
 $\bar{\alpha}$ - stała estymatora mocy sygnału $x(n)$
 μ - wielkość tłumienia w metodzie stochastycznego gradientu z tłumieniem
 $e_f(n|m), e_b(n|m)$ - sygnały w m -tym ogniwie predyktora kratowego
 $R_i^{fm} = E[e_f(n|m)e_f(n+i|m)]$ - i -ty współczynnik autokorelacji sygnału $e_f(n|m)$
 $R_i^{bm} = E[e_b(n|m)e_b(n+i|m)]$ - i -ty współczynnik autokorelacji sygnału $e_b(n|m)$
 $R_i^{b_fm} = E[e_b(n|m)e_f(n+i|m)]$ - i -ty współczynnik korelacji wzajemnej sygnałów $e_b(n|m)$ i $e_f(n|m)$
 M - długość okna czasowego, także liczba ogniw predyktora o zmiennych opóźnieniach
 $\mathcal{E}_f(n|p)$ - energia sygnału $\tilde{\epsilon}(i)$ w obrębie okna, jednocześnie kwadrat normy wektora $\mathbf{E}_f(n|p)$
 λ - współczynnik charakteryzujący okno o kształcie wykładniczym
 $\mathbf{X}(n)$ - wektor skwantowanego sygnału akustycznego
 $\hat{\mathbf{X}}(n)$ - wektor sygnału predykcji
 $\mathbf{S}(n|p)$ - macierz próbek sygnału akustycznego
 $\mathbf{X}(n|1), \dots, \mathbf{X}(n|p)$ - kolumny macierzy $\mathbf{S}(n|p)$
 $\mathbf{X}_{ort}(n|1), \dots, \mathbf{X}_{ort}(n|p)$ - baza ortogonalna
 $\mathbf{E}_b(n|m) = \mathbf{X}_{ort}(n|m)$ - wektory bazy ortogonalnej dla predyktora o strukturze kratowej
 $\mathbf{E}_f(n|m)$ - wektor błędu predykcji dla predyktora rzędu m
 $K_{m+1}(n)$ - iloczyn skalarny wektorów $\mathbf{E}_b(n|m+1)$ i $\mathbf{E}_f(n|m)$
 $\mathcal{E}_b(n-1|m)$ - kwadrat normy wektora $\mathbf{E}_b(n|m+1)$
 $\gamma(n|m)$ - współczynnik charakteryzujący szybkość zmian sygnału
 R_i - i -ty współczynnik autokorelacji sygnału
 H - entropia
 $H(\epsilon)$ - entropia różniczkowa dla gęstości prawdopodobieństwa $p(\epsilon)$
 E - operator uśredniania statystycznego
 $\mathbf{R}^p(\mathbf{x})$ - macierz korelacji sygnału $x(n)$, stopnia p ; jeżeli sygnał ma zerową wartość średnią, $\mathbf{R}^p(\mathbf{x})$ jest również macierzą kowariancji
 $\rho^p(\mathbf{x})$ - znormalizowana macierz korelacji sygnału $x(n)$, stopnia p

- ρ_{ij} - znormalizowany współczynnik korelacji $R_{|i-j|}/R_0$
 \det - wyznacznik
 $P(z)$ - transmitancja predyktora
 $P_{STP}(z)$ - transmitancja predyktora krótkookresowego
 $P_{LTP}(z)$ - transmitancja predyktora długookresowego
 $A(z)$ - transmitancja filtra inwersyjnego
 $1/A(z)$ - transmitancja filtra predykcyjnego modelującego sygnał akustyczny
 $X(z) = \mathcal{Z}[x(n)]$ - transformata \mathcal{Z} sygnału akustycznego
 $E(z) = \mathcal{Z}[e(n)]$ - transformata \mathcal{Z} błędu kwantyzacji
 $\mathcal{E}(z) = \mathcal{Z}[\epsilon(n)]$ - transformata \mathcal{Z} sygnału różnicowego (błędu predykcji)
 $\tilde{\mathcal{E}}(z) = \mathcal{Z}[\tilde{\epsilon}(n)]$ - transformata \mathcal{Z} skwantowanego sygnału różnicowego
 $\tilde{X}(z) = \mathcal{Z}[\tilde{x}(n)]$ - transformata \mathcal{Z} skwantowanego sygnału akustycznego
 N - wymiar wektorów przetwarzanych w kwantyzatorze wektorowym i koderze CELP
 $\mathbf{F} = \{\mathbf{f}_1, \dots, \mathbf{f}_L\}$ - słownik kwantyzatora wektorowego
 \mathbf{x} - wektor utworzony z N kolejnych próbek sygnału akustycznego ($\mathbf{x}(i)$ oznacza i -ty kolejny wektor)
 $\{\mathbf{x}(i)\}$ - zbiór wektorów, tworzący frazę treningową
 $\tilde{\mathbf{x}}$ - wektor N próbek sygnału wyjściowego dekodera (sygnału zrekonstruowanego)
 $\tilde{\mathbf{x}}(i)$ - i -ty kolejny wektor jak wyżej
 $P(\mathbf{f}_j)$ - komórka Voronoi'a, której centroidem jest wektor \mathbf{f}_j
 D - energia błędu kwantyzacji dla frazy treningowej
 $p(\mathbf{x})$ - gęstość prawdopodobieństwa dla wektorów próbek sygnału \mathbf{x}
 $\bar{c}(N), c(N)$ - stałe zależne od wymiaru wektora
 $X(f) = X(z)$ dla $z = e^{j2\pi fT}$ - transformata Fouriera sygnału akustycznego
 T - okres próbkowania
 $f_s = 1/T$ - częstotliwość próbkowania
 $y(n)$ - sygnał akustyczny po przetworzeniu (np. preemfazie, transformacji)
 $\tilde{y}(n), \mathbf{y}, \tilde{\mathbf{y}}, Y(z), Y(f)$ - określone dla sygnału $y(n)$ podobnie jak odpowiednie wielkości dla sygnału $x(n)$
 $W(f), W^{-1}(f)$ - transmitancja filtra preemfazy i deemfazy
 w_i, \bar{w}_i - próbki odpowiedzi impulsowej filtra preemfazy i deemfazy
 $\eta/2$ - gęstość mocy (dwustronna) szumu nieskorelowanego
 \mathbf{W} - macierz filtra preemfazy, stopnia $N \times N$
 $\mathbf{\Omega}$ - macierz filtra preemfazy, stopnia $N \times (\bar{N} + N)$
 γ - współczynnik charakteryzujący głębokość preemfazy filtra $W(z) = \frac{A(z)}{A(z/\gamma)}$
 $\mathbf{F}^1, \dots, \mathbf{F}^K$ - słowniki elementarne, których iloczyn kartezyjski tworzy słownik \mathbf{F}
 $\mathbf{F}^k = \{\mathbf{f}_1^k, \dots, \mathbf{f}_{L_k}^k\}$ - k -ty słownik elementarny zawierający L_k wektorów (indeks k może być pominięty, np. w jednostopniowym SGVQ, gdzie słownik kształtów zawiera wektory $\mathbf{f}_1, \dots, \mathbf{f}_{L_s}$)
 $\bar{\mathbf{F}}^k = \{\bar{\mathbf{f}}_j^k\}$ - k -ty słownik elementarny (lub kompletny słownik \mathbf{F}), po zortogonalizowaniu względem wektorów wybranych w poprzednich etapach modelowania

- $g_{m_k}^k$ - współczynnik wzmocnienia w k -tym etapie modelowania, w wielostopniowym VQ lub koderze CELP (indeks k może być pominięty, np. w jednostopniowym SGVQ, gdzie słownik wzmocnień zawiera skalary g_m , $m = 1, \dots, L_g$)
 $\bar{g}_{m_k}^k$ - współczynnik wzmocnienia w k -tym etapie modelowania, skojarzony z wektorem należącym do zortogonalizowanego słownika kształtów $\bar{\mathbf{F}}^k$
 $g^k = gs^k$ - współczynnik wzmocnienia jak wyżej, gdy współczynniki wzmocnienia różnią się jedynie znakiem
 $s^k = \pm 1$ - znak współczynnika wzmocnienia g^k
 $r(n)$ - próbka sygnału resztkowego w koderze CELP, odpowiednik sygnału różnicowego $\epsilon(n)$ w koderze ADPCM
 $\tilde{r}(n)$ - odtworzony sygnał resztkowy
 $\mathbf{r}, \tilde{\mathbf{r}}$ - wektory (N -wymiarowe) utworzone z próbek sygnału resztkowego i odtworzonego sygnału resztkowego
 $H(z) = 1/A(z)$ - filtr predykcyjny (wyjściowy)
 $H'(z) = 1/A(z/\gamma)$ - filtr percepcyjny
 \mathbf{H}' - macierz filtru percepcyjnego
 $D(i)$ - energia błędu kwantyzacji dla i -go wektora sygnału
 $\tilde{\mathbf{y}}_0$ - odpowiedź filtru percepcyjnego na zerowe pobudzenie
 $\mathbf{y}_0 = \mathbf{y} - \tilde{\mathbf{y}}_0$ - zmodyfikowany wektor sygnału percepcyjnego (sygnał docelowy w pierwszym etapie modelowania w CELP)
 \mathbf{y}_k - zmodyfikowany wektor sygnału percepcyjnego (sygnał docelowy w $k + 1$ etapie modelowania w wielostopniowym VQ lub CELP)
 $\mathbf{C}^1, \dots, \mathbf{C}^K$ - słowniki sygnałów pobudzających, których iloczyn kartezyjski tworzy słownik "mieszany" \mathbf{C}
 $\mathbf{C}^k = \{\mathbf{c}_1^k, \dots, \mathbf{c}_{L_k}^k\}$ - k -ty słownik elementarny sygnałów pobudzających, zawierający L_k wektorów (indeks k może być pominięty, np. w jednostopniowym CELP, gdzie słownik kształtów sygnałów pobudzających zawiera wektory $\mathbf{c}_1, \dots, \mathbf{c}_{L_s}$)
 $\mathbf{R}_j^N(\mathbf{y})$ - macierz korelacji wektorów z j -tej komórki Voronoi'a
 $\mathbf{f}_j^{(k)}$ - centroid j -tej komórki, obliczony w k -tej iteracji
 $g_m^{(j)}$, $m = 1, \dots, L_g$ - m -ty element słownika wzmocnień, przyporządkowanego wektorowi kształtu \mathbf{f}_j
 \mathbf{g} - wektor, którego składowymi są wzmocnienia odpowiadające kolejnym wektorom słownika \mathbf{F}
 $\tilde{\mathbf{g}}$ - wektor zawierający skwantowane wzmocnienia
 $\mathbf{P}(\mathbf{f})$ - operator ortogonalizacji względem \mathbf{f}
 $r(k, j) = \frac{\mathbf{f}_j^t \mathbf{f}_{j_k}}{\mathbf{f}_{j_k}^t \mathbf{f}_{j_k}}$ - współczynnik rzutowania wektora \mathbf{f}_j na \mathbf{f}_{j_k}
 $\bar{r}(k, j) = \frac{(\bar{\mathbf{f}}_{j_k}^k)^t \bar{\mathbf{f}}_j^k}{\|\bar{\mathbf{f}}_{j_k}^k\|^2}$ - współczynnik rzutowania wektora $\bar{\mathbf{f}}_j^k$ na $\bar{\mathbf{f}}_{j_k}^k$
 $\bar{\mathbf{R}}^k$ - macierz ortogonalizacji Grama- Schmidta, w k -tym kroku
 $\Psi = (\mathbf{H}')^t \mathbf{y}_0$ - odfiltrowany "wstecznie" wektor docelowy \mathbf{y}_0
 Φ - macierz, równa $(\mathbf{H}')^t \mathbf{H}'$

- \mathbf{f}^k - wektor definiujący jednowymiarową przestrzeń, otrzymany w k -tej iteracji
 \mathbf{c}^k - nieodfiltrowana wersja wektora \mathbf{f}^k
 \mathbf{T} , $\bar{\mathbf{T}}$ - macierze transformaty unitarnej, stopnia $N \times N$ i $\bar{N} \times \bar{N}$, $\bar{N} \geq N$
 $\bar{\mathbf{x}}$ - \bar{N} -wymiarowy wektor przed transformacją
 $\bar{\mathbf{y}} = [\bar{\mathbf{y}}_1^t, \bar{\mathbf{y}}_2^t, \dots, \bar{\mathbf{y}}_M^t]^t$ wyżej wymieniony wektor po transformacji, podzielony na M wektorów składowych
 b_1, \dots, b_M - rozdzielczości kwantyzatorów wektorowych operujących w dziedzinie transformaty
 N_1, \dots, N_M - długości wektorów $\bar{\mathbf{y}}_1, \dots, \bar{\mathbf{y}}_M$
 L_1, \dots, L_M - liczba wektorów w słownikach kwantyzatorów wektorowych (PCVQ) operujących w dziedzinie transformaty
 $G(\bar{\mathbf{T}})$ - zysk dla macierzy transformującej $\bar{\mathbf{T}}$
 $\bar{\mathbf{T}}(M)$ - macierz opisująca transformację M kolejnych, $N = \bar{N}/M$ - wymiarowych wektorów, z wykorzystaniem tej samej macierzy transformującej \mathbf{T}
 $\bar{\mathbf{P}}(M)$ macierz opisująca ekstrakcję M wektorów polifazowych o wymiarze $N = \bar{N}/M$
 $\mathbf{T}(m, l)$ - macierz transformacji "z nakładaniem", stopnia $m \times ml$
 $\bar{\mathbf{T}}_m$ - macierz transformacji z nakładaniem, dla ciągu m - wymiarowych wektorów
 $\bar{\mathbf{P}}(k, m)$ - macierz ekstrakcji k -tej składowej z ciągu m - wymiarowych wektorów
 $\sigma_1^2 \dots \sigma_{\bar{N}}^2$ - wariancje składowych wektora $\bar{\mathbf{y}}$, otrzymanego z wykorzystaniem KLT
 $\tilde{\bar{\mathbf{y}}}$ - skwantowana wersja wektora $\bar{\mathbf{y}}$
 $|\bar{\mathbf{y}}|$ - wektor o składowych równych wartościom bezwzględnym składowych wektora $\bar{\mathbf{y}}$
 $|\tilde{\bar{\mathbf{y}}}|$ - skwantowana wersja wektora $|\bar{\mathbf{y}}|$
 $\hat{\mathbf{y}}$ - wektor o składowych będących wynikiem dzielenia składowych wektora $\bar{\mathbf{y}}$ przez składowe wektora $|\tilde{\bar{\mathbf{y}}}|$ (zwany umownie widmem fazy)
 $\tilde{\hat{\mathbf{y}}}$ - skwantowana wersja wektora $\hat{\mathbf{y}}$
 $\mathbf{p} = [p_1, p_2, \dots, p_N]^t$ - wektor parametrów sygnału, wykorzystywany w procesie klasyfikacji
 $\hat{\alpha} = [\hat{\alpha}_1, \hat{\alpha}_2, \dots, \hat{\alpha}_N]^t$ - współczynniki dyskryminatora liniowego
 C_1, C_2 - klasy sygnałów (np. mowa bezdźwięczna i dźwięczna)
 $\mathbf{m}_1, \mathbf{m}_2$ - wartości średnie wektora parametrów \mathbf{p} , odpowiednio w klasie C_1 i C_2
 $\bar{\mathbf{C}}_1, \bar{\mathbf{C}}_2$ - macierze kowariancji wektora parametrów \mathbf{p} , odpowiednio w klasie C_1 i C_2
 T_p - próg decyzyjny
 d - klasyfikator dźwięczności sygnału mowy
 T_0 - okres tonu krtaniowego
 $E_{sig}(i)$ - moc sygnału wejściowego w obrębie i -tej ramki liczącej \bar{N} próbek
 $L_{sp}(i)$ - poziom mowy, obliczony w i -tej ramce
 $L_{sil}(i)$ - poziom ciszy (szumu otoczenia), obliczony w i -tej ramce
 $T_{sp}(i)$ - próg dźwięczności (minimalny poziom mowy dźwięcznej) w i -tej ramce
 E_{lp}, E_{hp} - moc sygnału wejściowego odfiltrowanego dolnopasmowo (0-1kHz) i górnopasmowo (3-4kHz)

- $L_{lp} = E_{lp}/E_{sig}$, $L_{hp} = E_{hp}/E_{sig}$ - znormalizowane wartości ww. parametrów
- E_{res} - moc sygnału błędu predykcji
- $L_{res} = E_{res}/E_{sig}$ - współczynnik charakteryzujący predykcyjność sygnału (znormalizowana wartość E_{res})
- $r_{max/rms}$ - stosunek wartości maksymalnej do wartości skutecznej w sygnale błędu predykcji
- R_{max} - maksimum autokorelacji odfiltrowanego dolnopasmowo sygnału błędu predykcji, mierzone w zakresie zmienności okresu tonu krtaniowego
- $r_{max} = R_{max}/R_0$ - znormalizowane maksimum autokorelacji
- i_{max} - położenie ww. maksimum autokorelacji
- \bar{i}_{max} - ważona wartość średnia okresu tonu krtaniowego
- $LSF_1, LSF_2 \dots, LSF_p$ - współczynniki widma prążkowego (LSF lub LSP)
- $L_v(i)$ - poziom mowy dźwięcznej, obliczony w i -tej ramce sygnału

Dodatek B

Ważniejsze skróty

- ACELP - *Algebraic Code Excited Linear Prediction* - koder CELP o pobudzeniu algebraicznym
- AC-3 - standard kompresji dla dodatkowych kanałów audio w HDTV
- a/c - przetwornik analogowo- cyfrowy
- ACR - *Absolute Category Rating* - bezwzględna skala jakości w teście słuchowym
- ADPCM - *Adaptive Differential Pulse Code Modulation* - adaptacyjna różnicowa modulacja kodowo- impulsowa
- AR - *autoregressive* - autoregresyjny
- ARMA - *autoregressive moving average* - autoregresyjny z ruchomą średnią
- ATCELP - *Adaptive Transform CELP* - połączenie kodera CELP z koderem transformaty
- ATM - *Asynchronous Transfer Mode* - asynchroniczna technika transferu
- ATRAC - *Adaptive Transform Acoustic Coding* - standard kompresji opracowany dla Mini-Dysku
- BER - *Bit Error Rate* - elementowa stopa błędów
- c/a - przetwornik cyfrowo- analogowy
- CCR - *Comparison Category Rating* - skala porównawcza w teście słuchowym
- CDMA - *Code Division Multiple Access* - system wielokrotny z podziałem kodowym
- CELP - *Code Excited Linear Prediction* - koder predykcyjny (pobudzany sygnałami z książki kodowej)
- CGVQ - *Contour- Gain Vector Quantization* - kwantyzacja wektorowa typu kontur- wzmocnienie
- CS-ACELP - *Conjugate Structure Algebraic Code Excited Linear Prediction* - koder CELP o pobudzeniu algebraicznym w strukturze sprzężonej
- DAB - *Digital Audio Broadcasting* - radiodyfuzja cyfrowa
- DCC - *Digital Compact Cassette* - miniaturowa kasetka z zapisem cyfrowym
- DCR - *Degradation Category Rating* - skala pogorszenia jakości w teście słuchowym
- DCT - *Discrete Cosine Transform* - dyskretna transformata kosinusoidalna

- DFT - *Discrete Fourier Transform* - dyskretna transformara Fouriera
- DM - *Delta Modulation* - modulacja delta
- DPCM - *Differential Pulse Code Modulation* - różnicowa modulacja kodowo-impulsowa
- ELT - *Extended Lapped Transform* - "rozciągnięta" transformata "z nakładaniem" (należy do tzw. *lapped transforms* - transformat "z nakładaniem")
- FPLMTS - *Future Public Land Mobile Telecommunications System*
- GSM - *Global System of Mobile Communication* (pierwotnie: fr. *Groupe Spécial Mobile*) - globalny system komunikacji ruchomej
- GSM- FR - *Full Rate GSM coder* - koder GSM o pełnej przepływności binarnej
- GSM- HR - *Half Rate GSM coder* - koder GSM o "połówkowej" przepływności binarnej
- GSM- EFR - *Enhanced Full Rate GSM coder* - rozszerzony koder GSM o pełnej przepływności binarnej
- GSM- AMR - *Adaptive Multi Rate GSM coder* - koder GSM o zmiennej przepływności binarnej
- HMM - *Hidden Markov Models* - ukryte modele Markowa
- ISO - *Int. Standards Organisation* - Międzynar. Organizacja Standardów
- ITU-T - *International Telecommunication Union - Telecommunications Standardization Sector* (dawniej CCITT)
- KF - fale krótkie
- KLT - *Karhunen- Loeve Transform* - transformata Karhunen- Loevego
- LBG - *Linde Buzo Gray algorithm* - algorytm projektowania słowników kwantyzatorów wektorowych (skrót pochodzi od nazwisk twórców)
- LDCELP - *Low Delay Code Excited Linear Prediction* - koder CELP o zmniejszonym opóźnieniu
- LP - *Linear Prediction* - predykcja liniowa
- LPC - *Linear Predictive Coder* - koder predykcyjny
- LSF - *Line Spectrum Frequencies* lub LSP - *Line Spectrum Pairs* - współczynniki widma prążkowego
- LTP - *Long- Term Predictor* - predyktor długookresowy
- LS - *least squares* - (metoda) najmniejszych kwadratów
- MA - *moving average* - ruchoma wartość średnia
- MBE- *Multiband Excitation* - pobudzenie wielopasmowe
- MDCT - *Modified Discrete Cosine Transform* - zmodyfikowana DCT (należy do tzw. *lapped transforms* - transformat "z nakładaniem")
- MELP - *Mixed Excitation Linear Prediction* - wokoder predykcyjny o pobudzeniu mieszanym
- Mflops - mln operacji zmiennoprzecinkowych na sekundę (miara złożoności obliczeniowej)
- MLT - *Modulated Lapped Transform* - transformata modulowana (należy do tzw. *lapped transforms* - transformat "z nakładaniem")
- MNRU - *Modulated Noise Reference Unit* - układ odniesienia dla oceny jakości

- sygnału metodą porównawczą
- MP - *multipulse* - wieloimpulsowy
- MP-LPC - *Multipulse Linear Predictive Coder* - koder predykcyjny CELP o pobudzeniu impulsowym
- MPEG - *Moving Pictures Expert Group*
- MPEG-1 Audio - standard ISO/IEC IS 11172
- MPEG-2 AAC (*Advanced Audio Coding*) - standard ISO/IEC IS 13818-7
- MUSICAM - standard kompresji dla europejskiej radiodifuzji cyfrowej
- NICAM - *Nearly Instantaneous Companding Audio Multiplex* - standard kompresji audio wykorzystywany w BBC
- PASC - *Precision Adaptive Subband Coding* - standard kompresji dla miniaturowej kasy cyfrowej (DCC)
- PCM - *Pulse Code Modulation* - modulacja kodowo-impulsowa
- PCS - *Personal Communications Services*
- PCVQ - *Product Code Vector Quantizer* - kwantyzator wektorowy z dekompozycją słownika
- PSI-CELP - koder CELP japońskiej telefonii komórkowej
- PWI - *Prototype Waveform Interpolation* - koder predykcyjny z interpolacją prototypu sygnału pobudzającego
- QCELP - *Qualcomm CELP* - koder CELP firmy Qualcomm (o zmiennej przepływności binarnej)
- QMF - *Quadratic Mirror Filters* - filtry analizy i syntezy charakteryzujące się idealną rekonstrukcją sygnału
- QPSK - *Quadrature Phase Shift Keying* - 4-wartościowa modulacja fazy
- RDF - *Rate Distortion Function* - funkcja wiążąca przepływność binarną ze zniekształceniami kodowanego sygnału
- RPE-LPC - *Regular Pulse Excitation Linear Predictive Coder* - koder predykcyjny o impulsowym pobudzeniu regularnym
- RSM - fr: *rapport signal- masque* (ang: SMR - *Signal to Mask Ratio*) - stosunek mocy sygnału do progu maskowania
- SELP - *Stochastically Excited Linear Prediction* - koder CELP o pobudzeniu stochastycznym
- SEV - *Self-Excited Vocoder* - koder CELP pobudzany wyłącznie wektorami ze słownika adaptacyjnego
- SGVQ - *Shape- Gain Vector Quantizer* - kwantyzator wektorowy typu kształt-wzmocnienie
- SNR - *Signal to Noise Ratio* - stosunek mocy sygnału do mocy szumu
- STP - *Short- Term Predictor* - predyktor krótkookresowy
- TPC - *Transform Predictive Coder* - koder predykcyjny wykorzystujący kodowanie transformaty
- TCX - *Transform Coded Excitation* - koder predykcyjny z kodowaniem transformaty sygnału pobudzającego
- TDMA - *Time Division Multiple Access* - system wielokrotny z podziałem cza-

sowym

UMTS - *Universal Mobile Telephone System*

UV - *unvoiced* - bezdźwięczny (o głosce)

V - *voiced* - dźwięczny (o głosce)

VAD - *Voice Activity Detector* - dyskryminator "cisza- mowa"

VQ - *Vector Quantizer* - kwantyzator wektorowy

VSELP - *Vector Sum Excited Linear Prediction* - koder CELP pobudzany sumą wektorów

VXC - *Vector Excitation Coding* - kodowanie wektorowe sygnału pobudzającego

Dodatek C

Przegląd standardów kompresji

W ostatnim dziesięcioleciu, ze względu na wielką różnorodność potrzeb, opracowano wiele koderów sygnału mowy i innych sygnałów akustycznych [6]. W telekomunikacji nadal największe znaczenie ma kodowanie sygnału telefonicznego [45]. W miarę rozwoju multimediiów coraz większe znaczenie ma przesyłanie sygnału mowy o poszerzonym pasmie (do 7 kHz a także do 16 kHz) i sygnałów muzycznych.

W niniejszym punkcie zostaną omówione najważniejsze standardy i metody kodowania sygnałów: sygnału telefonicznego, sygnału mowy o poszerzonym pasmie i sygnałów akustycznych o wysokiej jakości (hi-fi).

C.1 Standardy dla sygnałów telefonicznych

Dotychczas opracowano wiele różnych norm kodowania sygnału telefonicznego, jednak większość z nich ma charakter ściśle lokalny, obowiązujący w obrębie konkretnej organizacji, zajmującej się np. telefonią komórkową, łącznością morską czy lotniczą. Poniżej wymieniono niektóre z tych norm:

- PCM - CCITT (ITU-T) G711 (1972 r.) - 64 kbit/s;
- ADPCM - CCITT (ITU-T) G721 (1984 r.) - 32 kbit/s;
- ADPCM - CCITT (ITU-T) G726 (1988 r.) - 40, 32, 24 i 16 kbit/s;
- ADPCM - CCITT (ITU-T) G727 (1989 r.) - jak wyżej;
- LDCELP - CCITT (ITU-T) G728 (1991 r.) - 16 kbit/s;
- CS-ACELP - CCITT (ITU-T) G729 (1996 r.) - 8 kbit/s;
- ACELP - CCITT (ITU-T) G723.1 (1996 r.) - 6.3 i 5.3 kbit/s;
- GSM-FR - Europejska telefonia komórkowa (1992 r.) - 13 kbit/s;
- GSM-HR - Europejska telefonia komórkowa (1994 r.) - 6.5 kbit/s;

- GSM-EFR - Europejska telefonia komórkowa (1996 r.) - 12.2 kbit/s;
- VSELP (IS54) - Amerykańska telefonia komórkowa (1990 r.) - 8 kbit/s;
- QCELP (IS95) - Amerykańska telefonia komórkowa (1993 r.) - zmienna przepływność binarna, średnio 4 kbit/s;
- 1016 CELP - Fed. Stand. USA 1016 (1991 r.) - 4.8 kbit/s;
- LPC10 - Fed. Stand. USA 1015 (1982 r.) - 2.4 kbit/s;
- MELP - Fed. Stand. USA 1017 (1997 r.) - 2.4 kbit/s.

Listę tę można by też uzupełnić np. o kodery INMARSAT do komunikacji ze statkami, a ostatnio również samolotami i pojazdami [83], kodery do transmisji satelitarnej [113] itp.

Wyżej wymienione standardy można podzielić na dwie grupy. Do pierwszej należy zaliczyć normy CCITT (ostatnio ITU-T: *International Telecommunication Union - Telecommunications Standardization Sector*) o zasięgu ogólnosiwiatowym. Dwie pierwsze normy ITU-T (modulacja kodowo-impulsowa PCM [17] i adaptacyjna różnicowa modulacja kodowo-impulsowa ADPCM [18]) cechuje największa uniwersalność. Kodery PCM i transkodery PCM/ADPCM/PCM mogą być umieszczone w sieci komutowanej, w której, oprócz sygnału mowy, pojawiają się sygnały transmisji danych. Kodery te spełniają wszystkie wymagania nałożone przez ITU-T na łącza komutowane: małe opóźnienie, możliwość pracy tandemowej (wielokrotne przetwarzanie a/c i c/a), przenoszenie sygnałów szybkiej transmisji danych. Niestety za tę uniwersalność płaci się dużą przepływnością binarną (64 kbit/s dla PCM i 32 kbit/s dla ADPCM). Opracowano również dwie normy pokrewne dla ADPCM: G726 [15] i G727 [16], dopuszczające przepływności binarne 40, 32, 24 i 16 kbit/s. Niestety obserwuje się przy tym wyraźne pogorszenie jakości mowy, zwłaszcza dla szybkości transmisji 16 kbit/s.

Z tego względu CCITT przyjęło kolejny standard: LDCELP - *Low Delay Code Excited Linear Prediction* o przepływności 16 kbit/s [27, 28, 19]. Zapewnia on bardzo dobrą "telefoniczną" jakość sygnału mowy, jednak nie przenosi sygnałów szybkiej transmisji danych (powyżej 10 kbit/s). Wymaga to konstruowania urządzeń wykrywających sygnały transmisji danych i przesyłających te sygnały z pominięciem LDCELP.

W 1996 r. ITU-T zatwierdziło ostateczną wersję standardu G729 dla przepływności 8 kbit/s [177, 181, 100, 176]. Jest to koder CS-ACELP (*Conjugate Structure Algebraic Code Excited Linear Prediction*), oparty na dwóch propozycjach: kanadyjsko-francuskiej [179] i japońskiej [110]. Dla tej przepływności binarnej ITU-T złagodziło wymagania odnośnie opóźnienia i przenoszenia sygnałów transmisji danych, co ogranicza jego zastosowanie do odcinków końcowych łącza (np. telefon komórkowy).

Przyjęty w 1996 r. standard ACELP stanowi normę ITU-T G723.1 i obejmuje dwie przepływności binarne: 6.3 i 5.3 kbit/s [99]. Przewidziano jego stosowanie głównie w wideotelefonii (standard multimedialny H324). Przepływność binarna może być zmieniana co ramkę (30 ms) - daje to większe możliwości realizacji różnego typu wideotelefonów.

Do drugiej grupy należą normy o zasięgu lokalnym, związane z konkretnym zastosowaniem. W pierwszym rzędzie należy tu wymienić szereg standardów dla telefonii komórkowej.

Normą europejską jest koder GSM- FR [67, 201] o szybkości transmisji 13 kbit/s (po dodaniu kodu nadmiarowego zabezpieczającego przed błędami przepływność wzrasta do 22.8 kbit/s). Koder ten nie zapewnia jakości sygnału mowy akceptowanej przez ITU-T, odzwierciedla on stan techniki z końca lat 80 ub. stulecia. W związku z tym prowadzono badania nad nowym koderem GSM-EFR [88, 116, 195]. Odpowiednia norma ETSI została opublikowana w końcu 1996 roku [69]. Z kolei koder GSM Half Rate (tzw. "połówkowy" [68]) ma przepływność binarną 5.6 kbit/s, jednak po dodaniu kodów nadmiarowych zabezpieczających przed błędami wzrasta ona do 11.4 kbit/s. Oparty on jest na propozycji VSELP [79] firmy Motorola, podobnie jak koder amerykańskiej telefonii komórkowej (norma IS54, przepływność 8 kbit/s [78]).

Wszystkie kodery GSM są wyposażone w dodatkowe algorytmy umożliwiające przerywaną transmisję (DTX- *discontinuous transmission*). Dyskryminator mowa/cisza (VAD - *voice activity detector*) wykrywa ramki nie zawierające mowy i przerywa transmisję. W tym czasie w odbiorniku generowany jest szum o niewielkiej mocy (ang. *comfort noise*), aby uniknąć przykrego wrażenia przerwania łączności przy odbieraniu absolutnej ciszy w słuchawce.

Kodery "połówkowe" zapewniają dwukrotne zwiększenie liczby abonentów, jednak prawdziwą rewolucję (dalsze kilkakrotne zwiększenie liczby abonentów) zapewnią systemy telefonii komórkowej trzeciej generacji, oparte na CDMA (*code division multiple access*) i koderach o zmiennej przepływności binarnej [43, 46, 76, 159, 89]. W USA przyjęto już normę IS94 dla kodera QCELP [42]. Koder ten ma cztery przepływności binarne: 8, 4, 2 i 1 kbit/s, przełączane w funkcji mocy chwilowej sygnału mowy (8 kbit/s dla głośno wypowiedzianych samogłosek, 1 kbit/s dla ciszy międzywyrazowej i międzyzdaniowej) i w funkcji pojemności sieci (liczby zgłoszonych rozmów). Na doświadczeniach amerykańskiej telefonii komórkowej trzeciej generacji będą oparte przyszłe systemy telekomunikacji osobistej. Należy wymienić tu następujące projekty: PCS - *Personal Communications Services* [120], UMTS - *Universal Mobile Telephone System* [20] i FPLMTS - *Future Public Land Mobile Telecommunications System*.

W kierunku zmiennej przepływności binarnej ewoluują również systemy GSM. W ostatnim okresie opracowano algorytmy tzw. kodera AMR [159, 89]. Ważnym zagadnieniem jest transmisja sygnału mowy w sieciach z transmisją pakietów (ATM, Voice over IP). Transmisja pakietów nakłada szczególne wymagania na kodery mowy. Poważnym problemem jest zmienne opóźnienie w transmisji poszczególnych pakietów, prowadzące do utraty pakietów docierających zbyt późno do odbiornika. Odbiornik musi być wyposażony w algorytmy interpolacji utraconych pakietów [4]. Z drugiej strony, istnieje możliwość klasyfikacji bitów z punktu widzenia ich wpływu na jakość mowy i tworzenia pakietów uprzywilejowanych [49]. Innym rozwiązaniem jest tworzenie "ukrytego" kodu o mniejszej przepływności binarnej w kodzie o większej przepływności binarnej (ang. *embedded coding*). W przypadku pogorszenia się warunków transmisji wykorzystuje się jedynie ten "ukryty" kod do odtworzenia mowy o obniżonej jakości [92].

W warunkach, gdy poziom zakłóceń w transmisji jest wysoki, konieczne staje się obniżenie przepływności binarnej do 4.8 lub nawet 2.4 kbit/s. Dotyczy to planowanej, bezpośredniej komunikacji satelitarnej między obiektami ruchomymi, komunikacji podwodnej, specjalnej, a także przesyłania cyfrowego (najczęściej zaszyfrowanego) sygnału mowy w sieciach komuto-

wanych o złej jakości (np. w Polsce poza wielkimi ośrodkami miejskimi). W tych warunkach nie jest możliwe uzyskanie dobrej "telefonicznej" jakości mowy. Do niedawna najbardziej popularnym standardem był amerykański wokoder liniowo-predykcyjny LPC10 [198] o szybkości transmisji 2.4 kbit/s. W połowie lat 90 ub. stulecia administracja amerykańska rozpisała konkurs na opracowanie kodera nowej generacji o tej samej przepływności binarnej [205, 199]. Rozważane były kodery CELP [125], które przy tej przepływności nie pracują zbyt dobrze ze względu na duży szum kwantyzacji, oraz różne modyfikacje klasycznego wokodera predykcyjnego (patrz np. [135, 136]). Ostatecznie konkurs został rozstrzygnięty na korzyść koderu MELP - wokodera predykcyjnego o zmodyfikowanym pobudzeniu [191]. Najbardziej popularnym koderem dla przepływności 4.8 kbit/s jest amerykański koder CELP (Federal Standard 1016) [13, 12]. Znajduje on przede wszystkim zastosowanie do przesyłania zaszyfrowanego sygnału mowy w sieci publicznej i drogą radiową.

Większość koderów sygnału telefonicznego można łatwo zaimplementować w czasie rzeczywistym, na ogólnie dostępnych procesorach sygnałowych (patrz np. [83, 62, 66]).

C.2 Standardy dla sygnałów akustycznych

Dla wielu zastosowań (audio- i telekonferencje, rozgłaszanie komunikatów słownych, wyższej jakości wideotelefony) sygnał o jakości telefonicznej (pasmo do 4 kHz) nie zapewnia wymaganej jakości przekazu. Konieczne staje się poszerzenie pasma przynajmniej do 7 kHz. Z drugiej strony, istnieje potrzeba przesyłania sygnałów fonicznych o pasmie przynajmniej 15 kHz (radiodyfuzja cyfrowa naziemna i satelitarna, dźwięk w systemach multimedialnych itp.). W chwili obecnej najbardziej rozpowszechnione są następujące standardy:

- 2xADPCM - ITU-T G722 (1986 r.) - 64 kbit/s (56 kbit/s, 48 kbit/s);
- CD-Audio - około 1.5 Mbit/s (tryb stereo);
- MPEG-1 Audio - standard ISO/IEC IS 11172 (1991 r.) - od 32 kbit/s do ponad 300 kbit/s;
- MPEG-2 AAC (*Advanced Audio Coding*) - standard ISO/IEC IS 13818-7 (1997 r.);
- MPEG-4 - kodowanie mowy i muzyki w systemach multimedialnych - od 2 do ponad 128 kbit/s (1997 r.);
- NICAM - standard BBC (1993 r.) - 728 kbit/s (tryb stereo);
- AC-3 - standard dla dodatkowych kanałów audio w HDTV;
- MUSICAM - standard dla europejskiej radiodyfuzji cyfrowej;
- PASC - standard dla kaset cyfrowych (DCC);
- ATRAC - standard dla Mini-Dysków.

Standard ITU-T G722 [14] dla sygnałów o pasmie 7 kHz powstał kilkanaście lat temu i jest odzwierciedleniem ówczesnego stanu techniki [137]. Kodowanie odbywa się w dwóch pasmach metodą ADPCM. Przy maksymalnej przepływności binarnej 64 kbit/s (48 kbit/s w dolnym i 16 kbit/s w górnym pasmie) osiąga się jakość mowy [185], jaką obecnie udaje się uzyskać przy przepływnościach 32 kbit/s [164, 158], a nawet 20 kbit/s [10] i 16 kbit/s [71, 171]. Z drugiej strony, przy szybkości 64 kbit/s można zakodować sygnały akustyczne o pasmie 16 kHz i jakości zbliżonej do hi-fi [157, 117, 30]. Wobec powyższego znaczenie normy G722 będzie malało (zatwierdzono już normę G722.1, opartą na kwantowaniu sygnału w 14 podpasmach częstotliwości o szerokości 500 Hz).

Do zapisu dźwięku na płytach kompaktowych jest powszechnie stosowany standard kodowania PCM, oparty na próbkowaniu i kwantowaniu równomiernym. Wymaga on jednak przesyłania ponad 700 kbit/s w trybie mono (około 1.5 Mbit/s w trybie stereo), co zapewnia wysoką jakość dźwięku (hi-fi). Aby zmniejszyć przepływność ISO/IEC przyjęła standardy stratnego kodowania sygnałów fonicznych pod ogólną nazwą IS 11172 [157] (potoczna nazwa MPEG-1 Audio). Norma ta stanowi część standardu MPEG kodowania obrazu ruchomego z towarzyszącym dźwiękiem [72, 149, 184]. Jej część akustyczna powstała drogą modyfikacji wcześniej opracowanego kodera "MUSICAM" [41, 40]. Norma ta umożliwia zakodowanie szerokopasmowego sygnału akustycznego (16 kHz) z przepływnościami binarnymi od 32 kbit/s do ponad 300 kbit/s, jednak poniżej 64 kbit/s trudno mówić o zbliżonej jakości do hi-fi [10]. Lepsze wyniki, już przy przepływności 64 kbit/s, osiągnięto w koderze MPEG-2 AAC [155]. Standard ten umożliwia kodowanie z różną przepływnością binarną, jednak podstawowa przepływność wynosi 320 kbit/s dla sygnału o jakości prawie hi-fi w konfiguracji 5.1 obejmującej 5 kanałów: *center*, *left*, *right*, *left surround*, *right surround* oraz kanał LFE (*Low Frequency Enhancement*). Dla potrzeb transmisji mowy i muzyki w sieciach multimedialnych opracowano standard MPEG-4 [86]. Obejmuje on szereg koderów (parametryczne, predykcyjne, operujące w dziedzinie częstotliwości) o przepływnościach od 2 do ponad 128 kbit/s.

W niektórych przypadkach wykorzystuje się proste rozwiązania techniczne, nie zapewniające jednak dużej kompresji. Do takich standardów należy NICAM (*Nearly Instantaneous Companding Audio Multiplex*), wykorzystujący jedynie kwantowanie adaptacyjne [172]. Standard AC-3, opracowany w Dolby Labs, zapewnia dużą kompresję, lecz nie gwarantuje jakości hi-fi [196]. Standard PASC (*Precision Adaptive Subband Coding*) został opracowany dla miniaturowej kasety cyfrowej (DCC) - nie różni się on zasadniczo od warstwy I standardu MPEG [123]. Standard ATRAC (*Adaptive Transform Acoustic Coding*) został opracowany dla Mini-Dysku [200].

Jak wykazała praktyka, przyjęcie pewnej normy kodowania sygnału nie oznacza zakończenia badań w obszarze objętym tą normą. Normy reprezentują określony stan techniki i w miarę postępu naukowego i technologicznego są wypierane przez inne standardy. Dobrym przykładem jest tu amerykański wokoder LPC10 (norma federalna 1015 [198]), który został zastąpiony nowym standardem MELP [191]. Postęp w dziedzinie kodowania sygnału akustycznego o poszerzonym pasmie [115, 171, 158, 164, 126, 127, 117, 151, 30, 29] doprowadził do spadku znaczenia normy G722 (będzie stopniowo wypierana przez G722.1). Jak się wydaje, podobny los spotka wkrótce koder GSM-FR - zostanie on stopniowo wyparty przez "enhanced" GSM, a w końcu przez AMR GSM o zmiennej przepływności.

Bibliografia

- [1] K. Abou-Kassem. *Optymalizacja kwantyzera błędu predykcji*. PhD thesis, Instytut Telekomunikacji Politechniki Warszawskiej, 1990.
- [2] K. Abou-Kassem and P. Dymarski. Kwantyzery adaptacyjne w modulacji DPCM. *Kwartalnik Elektroniki i Telekomunikacji*, vol.37, z.1/2:239–251, 1991.
- [3] J.P. Adoul, P. Mabillean, M. Delprat, and S. Morisette. Fast CELP coding based on algebraic codes. *Proc. Int. Conf. Acoust., Speech, Signal Processing*, pages 1957–1960, 1987.
- [4] A.K. Anandakumar, A.V. McCree, and V. Viswanathan. Efficient, CELP based diversity schemes for VOIP. *Proc. Int. Conf. Acoust., Speech, Signal Processing*, 2000.
- [5] B.S. Atal. High-quality speech at low bit rates: multi-pulse and stochastically excited linear predictive coders. *Proc. Int. Conf. Acoust., Speech, Signal Processing*, pages 1681–1684, 1986.
- [6] B.S. Atal, V. Cuperman, and A. Gersho, editors. *Speech and Audio Coding for Wireless and Network Applications*. Kluwer, 1993.
- [7] B.S. Atal and J.R. Remde. A new model of LPC excitation for producing natural-sounding speech at low bit rates. *Proc. Int. Conf. Acoust., Speech, Signal Processing*, pages 614–617, 1982.
- [8] T. Berger. *Rate-distortion theory : a mathematical basis for data compression*. Prentice-Hall, 1971.
- [9] M. Berouti, H. Garten, P. Kabal, and P. Mermelstein. Efficient computation and encoding of the multi-pulse excitation for LPC. *Proc. Int. Conf. Acoust., Speech, Signal Processing*, pages 10.1.1–10.1.4, 1984.
- [10] K. Brandenburg, M. Dietz, E. Eberlein, and R. Bitto. Extending MPEG-Audio layer III to wideband speech coding. In *Proceedings of IEEE Workshop on Speech Coding : Speech Coding for the Network of the Future*, pages 7–8, October 13-15, 1993.
- [11] J.P. Campbell and T.E. Tremain. Voiced/unvoiced classification of speech with applications to the U.S. government LPC-10E algorithm. *Proc. Int. Conf. Acoust., Speech, Signal Processing*, art.9.11, 1986.
- [12] J.P. Campbell, T.E. Tremain, and V.C. Welch. The DoD 4.8 kbps standard (proposed federal standard 1016). *Advances in Speech Coding edited by B.S. Atal, V. Cuperman and A. Gersho, Kluwer*, 1991.

- [13] J.P. Campbell, V.C. Welch, and T.E. Tremain. An expandable error-protected 4800 bps CELP coder (U.S. federal standard 4800 bps voice coder). *Proc. Int. Conf. Acoust., Speech, Signal Processing*, pages 735–738, 1989.
- [14] CCITT. *CCITT Recommendation G.722: 7 kHz audio coding within 64 kbit/s*.
- [15] CCITT. *CCITT Recommendation G.726: 40-, 32-, 24-, and 16 kbit/s adaptive differential pulse code modulation*.
- [16] CCITT. *CCITT Recommendation G.727: 5-, 4-, 3-, and 2 bits per sample embedded adaptive differential pulse code modulation*.
- [17] CCITT. *CCITT Recommendation G.711: Pulse Code Modulation (PCM) of voice frequencies*, 1984.
- [18] CCITT. *CCITT Recommendation G.721: 32 kbit/s adaptive differential pulse code modulation (ADPCM)*, Aug. 1986.
- [19] CCITT. *CCITT Recommendation G.728: Coding of speech at 16 kbit/s using low delay code excited linear prediction*, Sep. 1992.
- [20] L. Cellario and D. Sereno. Variable rate speech coding for UMTS. In *Proceedings of IEEE Workshop on Speech Coding : Speech Coding for the Network of the Future*, pages 1–2, October 13-15, 1993.
- [21] W.Y. Chan and A. Gersho. High fidelity audio transform coding with vector quantization. *Proc. Int. Conf. Acoust., Speech, Signal Processing*, pages 1109–1112, 1990.
- [22] W.Y. Chan and A. Gersho. Constrained-storage vector quantization in high fidelity audio transform coding. *Proc. Int. Conf. Acoust., Speech, Signal Processing*, pages 3597–3600, 1991.
- [23] P.C. Chang, R.M. Gray, and J. May. Fourier transform vector quantization for speech coding. *IEEE Trans. on Com.*, Com-35, No 10:1059–1068, October 1987.
- [24] W. Chang and C. Wang. A masking-threshold-adapted weighting filter for excitation search. *IEEE Trans. on Speech and Audio Processing*, pages 124–132, March 1996.
- [25] W. Chang and C. Wang. Audio coding using masking-threshold adapted perceptual filter. *Proc. IEEE Workshop on Speech Coding for Telecommunications*, pages 9–10, October 1993.
- [26] J. Chao, H. Perez, and S. Tsuji. A fast adaptive filter algorithm using eigenvalue reciprocals as stepsizes. *IEEE Trans. Acoust., Speech, Signal Processing*, ASSP-38, August 1990.
- [27] J. Chen. High-quality 16 kb/s speech coding with a one-way delay less than 2 ms. *Proc. Int. Conf. Acoust., Speech, Signal Processing*, pages 453–456, 1990.
- [28] J. Chen, M. Melchner, and R. Cox. Real-time implementation and performance of a 16 kb/s low-delay CELP speech coder. *Proc. Int. Conf. Acoust., Speech, Signal Processing*, pages 181–184, 1990.
- [29] J.H. Chen. A high fidelity speech and audio codec with low delay and low complexity. *Proc. Int. Conf. Acoust., Speech, Signal Processing*, 2000.

- [30] J.H. Chen and D. Wang. Transform predictive coding of wideband speech signals. *Proc. Int. Conf. Acoust., Speech, Signal Processing*, 1996.
- [31] A. Chmielewski and P. Dymarski. The CELP coder with both adaptive and fixed part of long term predictor memory. *Proceedings of the YUTEL'90 - Yugoslav Conference on Telecommunications, Ljubljana*, 1990.
- [32] P. Cummiskey, N. Jayant, and J.L. Flanagan. Adaptive quantization in DPCM coding of speech. *B.S.T.J.*, vol.52, No.7, Sept. 1973.
- [33] V. Cuperman. Joint bit allocation and dimensions optimization for vector transform quantization. *IEEE Trans. on Information Theory*, Vol. 39, No. 1:302–305, January 1993.
- [34] V. Cuperman. On adaptive vector transform quantization for speech coding. *IEEE Trans. on Communications*, Vol. 37, No. 3:261–267, March 1989.
- [35] G. Davidson, L. Fielder, and M. Antill. High-quality audio transform coding at 128 kbits/s. *Proc. Int. Conf. Acoust., Speech, Signal Processing*, pages 1117–1120, 1990.
- [36] G. Davidson and A. Gersho. Complexity reduction methods for vector excitation coding. *Proc. Int. Conf. Acoust., Speech, Signal Processing*, pages 3055–3058, 1986.
- [37] G. Davidson and A. Gersho. Multiple-stage vector excitation coding of speech waveforms. *Proc. Int. Conf. Acoust., Speech, Signal Processing*, pages 163–166, 1988.
- [38] G. Davidson, M. Yong, and A. Gersho. Real-time vector excitation coding of speech at 4800 bps. *Proc. Int. Conf. Acoust., Speech, Signal Processing*, pages 2189–2192, 1987.
- [39] A. De and P. Kabal. Auditory distortion measure for speech coder evaluation - discrimination information approach. *Speech Communication*, Vol. 14:205–229, 1994.
- [40] Y.F. Dehery, M. Lever, and J.B. Rault. Une norme de codage sonore de haute qualité pour la diffusion, les télécommunications et les systèmes multimédias. *l'Echo des Recherches*, Vol. 151:17–28, 1993.
- [41] Y.F. Dehery, M. Lever, and P. Urcun. A musicam source codec for digital audio broadcasting and storage. *Proc. Int. Conf. Acoust., Speech, Signal Processing*, pages 3605–3608, 1991.
- [42] A. DeJaco, W. Gardner, P. Jacobs, and C. Lee. QCELP: the North American CDMA digital cellular variable rate speech coding standard. In *Proceedings of IEEE Workshop on Speech Coding : Speech Coding for the Network of the Future*, pages 5–6, October 13-15, 1993.
- [43] M. Delprat, A. Urie, and C. Evci. Speech coding requirements from the perspective of the future mobile systems. In *Proceedings of IEEE Workshop on Speech Coding : Speech Coding for the Network of the Future*, pages 89–90, October 13-15, 1993.
- [44] E.F. Deprettere and P. Kroon. Regular excitation reduction for effective and efficient LP-coding of speech. *Proc. Int. Conf. Acoust., Speech, Signal Processing*, pages 965–968, 1985.
- [45] S. Dimolitsas. Standardizing speech coding technology for network applications. *IEEE Communications Magazine*, Nov. 1993.

- [46] S. Dimolitsas and C. Ravishankar. Future objectives in low-rate speech coding technology standardization. In *Proceedings of IEEE Workshop on Speech Coding : Speech Coding for the Network of the Future*, pages 91–92, October 13–15, 1993.
- [47] A. Drozdek. *Wprowadzenie do kompresji danych*. Wyd. Naukowo- Techniczne, Warszawa, 1999.
- [48] H. Dudley. Remaking speech. *J. Acoust. Soc. Am.*, Vol. 11, No. 2:169–177, Oct. 1939.
- [49] R.E. Van Dyck, J. Patel, E. Simotas, and Q. Lin. A multimedia CELP/subband coder for ATM networks. In *Proceedings of IEEE Workshop on Speech Coding : Speech Coding for the Network of the Future*, pages 79–80, October 13–15, 1993.
- [50] P. Dymarski. Speech coding based on classification. *Proc. of the 4th International Workshop on Systems, Signals and Image Processing IWSSIP'97*, Poznań 1997.
- [51] P. Dymarski and A. Chmielewski. Kwantyzer wektorowy z predykcją długookresową. *Krajowe Sympozjum Telekomunikacji KST'89*, 1989.
- [52] P. Dymarski, A. Chmielewski, and S. Ratre. Kodowanie wektorowe sygnału mowy z niezależnym kwantowaniem kształtu i poziomu sygnału. *Krajowe Sympozjum Telekomunikacji KST'92*, 1992.
- [53] P. Dymarski and S. Kula. Kompresja szerokopasmowego sygnału mowy i sygnałów akustycznych. *Krajowe Sympozjum Telekomunikacji KST'97*, Bydgoszcz 1997.
- [54] P. Dymarski, S. Kula, E. Wirkus, and P. Zwierko. Wokoder predykcyjny o pobudzeniu mieszanym. *Krajowe Sympozjum Telekomunikacji KST'94*, 1994.
- [55] P. Dymarski, S. Kula, and P. Zwierko. Variable bit rate speech coder for military and special applications. *Regional Conference on Military Communication and Information Systems RCMCIS'99*, Zegrze, 6–8 Oct. 1999.
- [56] P. Dymarski and N. Moreau. Algorithms for the CELP coder with ternary excitation. *3rd European Conf. on Speech Commun. and Technology EURO-SPEECH'93*, vol.1:241–244, 1993.
- [57] P. Dymarski and N. Moreau. QR factorization in the CELP coder. In *Speech and Audio Coding for Wireless and Network Applications*, pages 231–238. Kluwer, 1993.
- [58] P. Dymarski and N. Moreau. Koder sygnału akustycznego dla zastosowań audiokonferencyjnych. *VI Sympozjum "Nowości w Technice Audio"*, Warszawa, Oct. 1999.
- [59] P. Dymarski, N. Moreau, and A. Chmielewski. Kwantyzer wektorowy ze słownikiem mieszanym. *Krajowe Sympozjum Telekomunikacji KST'90*, 1990.
- [60] P. Dymarski, N. Moreau, and A. Vigier. Optimal and sub-optimal algorithms for selecting the excitation in linear predictive coders. *Proc. Int. Conf. Acoust., Speech, Signal Processing ICASSP'90*, pages 485–488, 1990.
- [61] P. Dymarski, N. Moreau, and W. Vos. Détermination et codage de l'excitation dans un codeur CELP. *13 Coll. GRETSI, Juan-Les-Pins*, 1991.

- [62] P. Dymarski and P. Zwierko. Problemy implementacji koderów CELP na procesorach sygnałowych. *Krajowe Sympozjum Telekomunikacji KST'96*, Bydgoszcz 1996.
- [63] P. Dymarski and S. Ratre. Kwantowanie wektorowe sygnału mowy z wykorzystaniem predykcji długookresowej i transformaty ortogonalnej. *Krajowe Sympozjum Telekomunikacji KST'93*, 1993.
- [64] P. Dymarski and S. Ratre. Speech coding using product code vector quantization, long-term prediction and orthogonal transformation. *Kwartalnik Elektroniki i Telekomunikacji*, vol.40, z.3:333–356, 1994.
- [65] P. Dymarski and P. Zwierko. Wokoder 1200 bit/s z modelem do transmisji na falach krótkich. *Krajowe Sympozjum Telekomunikacji KST'98*, Bydgoszcz 1998.
- [66] P. Dymarski, P. Zwierko, S. Kula, and E. Wirkus. Wokoder 2.4/4.8/8 kbit/s dla transmisji utajnionej mowy w sieciach publicznych i komórkowych. *Krajowe Sympozjum Telekomunikacji KST'2000, oraz Biuletyn Wojsk. Inst. Łączności*, 2000.
- [67] ETSI. *ETSI Standard GSM 06.10: European digital cellular telecommunications system; GSM full rate speech transcoding*, 1995.
- [68] ETSI. *ETSI Standard GSM 06.20: European digital cellular telecommunications system; GSM half rate speech transcoding*, 1995.
- [69] ETSI. *ETSI Standard GSM 06.60: Digital cellular telecommunications system; enhanced full rate EFR speech transcoding*, 1996.
- [70] D.K. Freeman, G. Cosier, C.B. Southcott, and I. Boyd. The voice activity detector for the pan-european digital cellular mobile telephone service. *Proc. Int. Conf. Acoust., Speech, Signal Processing*, pages 369–372, 1989.
- [71] A. Fuldseth, E. Harborg, F.T. Johansen, and J.E. Knudsen. Wideband speech coding at 16 kbit/s for a videophone application. *Speech Communication*, Vol. 11:139–148, 1992.
- [72] D. Le Gall. MPEG: a video compression standard for multimedia applications. *Communications of the ACM*, 34, No. 4:46–58, April 1991.
- [73] A. Gersho. Advances in speech and audio compression. *Processing of the IEEE*, Vol. 82, No. 6, 1994.
- [74] A. Gersho. Asymptotically optimal block quantization. *IEEE Trans. on Information Theory*, Vol. IT-25:373–380, July 1979.
- [75] A. Gersho and R.M. Gray. *Vector quantization and signal compression*. Kluwer Academic Publishers, 1992.
- [76] A. Gersho and E. Paksy. Variable rate speech coding for cellular networks. In *Speech and Audio Coding for Wireless and Network Applications*, pages 77–84. Kluwer, 1993.
- [77] A. Gersho and Y. Shoham. Hierarchical vector quantization of speech with dynamic codebook allocation. *Proc. Int. Conf. Acoust., Speech, Signal Processing*, pages 10.9.1–10.9.4, 1984.

- [78] I.A. Gerson and M.A. Jasiuk. Vector sum excited linear prediction (VSELP) speech coding at 8 kbps. *Proc. Int. Conf. Acoust., Speech, Signal Processing*, pages 461–464, 1990.
- [79] I.A. Gerson and M.A. Jasiuk. A 5600 bps VSELP speech coder candidate for half rate GSM. In *Proceedings of IEEE Workshop on Speech Coding : Speech Coding for the Network of the Future*, pages 43–44, October 13-15, 1993.
- [80] G.H. Golub and C.F. Van Loan. *Matrix computations*. Johns Hopkins University Press, 1983 (Second Edition 1989).
- [81] J.D. Goodman and A. Gersho. Theory of an adaptive quantizer. *IEEE Trans. on Communications*, COM-22, No.8, Aug. 1974.
- [82] J.D. Goodman and R.M. Wilkinson. A robust adaptive quantizer. *IEEE Trans. on Communications*, COM-23, Nov. 1975.
- [83] R. Górajec, S. Kula, and M. Niesiołkowski. Implementacja w czasie rzeczywistym kodera kanałowego systemu INMARSAT-B. *Krajowe Sympozjum Telekomunikacji KST'93*, 1993.
- [84] W. Granzow, B.S. Atal, K.K. Paliwal, and J. Schroeter. Speech coding at 4 kb/s and lower using single-pulse and stochastic models of LPC excitation. *Proc. Int. Conf. Acoust., Speech, Signal Processing*, pages 217–220, 1991.
- [85] R.M. Gray, A. Buzo, Jr A.H. Gray, and Y. Matsuyama. Distortion measures for speech processing. *IEEE Trans. Acoust., Speech, Signal Processing*, ASSP-28:367–376, Aug. 1980.
- [86] B. Grill. A bit rate scalable perceptual coder for MPEG-4 audio. *Convention AES*, 1997/98.
- [87] A. Le Guyader, P. Combescure, C. Lamblin, M. Mouly, and J. Zurcher. A robust 16 kbit/s vector adaptive predictive coder for mobile communications. *Proc. Int. Conf. Acoust., Speech, Signal Processing*, pages 857–860, 1986.
- [88] H.B. Hansen, J. Haagen, K.B. Mikkelsen, H Nielsen, and K.J. Larsen. Toll quality codec for the GSM system. In *Proceedings of IEEE Workshop on Speech Coding : Speech Coding for Interoperable Global Communications*, pages 59–60, Sept. 20-22, 1995.
- [89] S. Heinen, M. Adrat, O. Steil, P. Vary, and W. Xu. A 6.1 to 13.3 kb/s variable rate CELP codec (VR-CELP) for AMR speech coding. *Proc. Int. Conf. Acoust., Speech, Signal Processing*, 1999.
- [90] M.L. Honig and D.G. Messerschmitt. *Adaptive filters - structures and applications*. Kluwer, 1984.
- [91] M.L. Honig and D.G. Messerschmitt. Comparison of adaptive linear prediction algorithms in ADPCM. *IEEE Trans. on Communications*, Vol. COM-30, No. 7:857–865, July 1982.
- [92] R.D. De Iacovo and D. Sereno. Embedded CELP coding for variable bit rate between 6.4 and 9.6 kbit/s. *Proc. Int. Conf. Acoust., Speech, Signal Processing ICASSP'91*, pages 681–683, 1991.
- [93] M.A. Ireton and C.S. Xydeas. On improving vector excitation coders through the use of spherical lattice codebooks. *Proc. Int. Conf. Acoust., Speech, Signal Processing*, pages 57–60, 1989.

- [94] F. Itakura. Line spectrum representation of linear predictive coefficients of speech signals. *J. Acoust. Soc. Am.*, 57, 1975.
- [95] ITU-T. *ITU-T Recommendation P.800: Methods for subjective determination of transmission quality*, August 1996.
- [96] ITU-T. *ITU-T Recommendation P.861: Objective quality measurement of telephone-band speech codecs*, August 1998.
- [97] ITU-T. *ITU-T Recommendation P.810: Modulated Noise Reference Unit MNRU*, Feb. 1996.
- [98] ITU-T. *ITU-T Recommendation P.830: Subjective performance assessment of telephone-band and wideband digital codecs*, Feb. 1996.
- [99] ITU-T. *ITU-T Recommendation G.723.1: Dual rate speech coder for multimedia communications transmitting at 5.3 and 6.3 kbit/s*, March 1996.
- [100] ITU-T. *ITU-T Recommendation G.729: Coding of speech at 8kbit/s using conjugate structure algebraic code excited linear prediction CS-ACELP*, March 1996.
- [101] N. Jayant. Adaptive quantization with a one-word memory. *B.S.T.J.*, vol.52, No 7, Sept. 1973.
- [102] N. Jayant and P. Noll. *Digital coding of waveforms - principles and applications to speech and video*. Prentice Hall, 1984.
- [103] A. Jbira, N. Moreau, and P. Dymarski. Low delay coding of wideband audio (20Hz-15kHz) at 64kbps. *Proc. Int. Conf. Acoust., Speech, Signal Processing*, Seattle, 12-15 May 1998.
- [104] M. Johnson and T. Taniguchi. Pitch-orthogonal code-excited LPC. *Proc. GLOBECOM*, pages 542-546, 1990.
- [105] M. Johnson and T. Taniguchi. Low-complexity multi-mode VXC using multi-stage optimization and mode selection. *Proc. Int. Conf. Acoust., Speech, Signal Processing*, pages 221-224, 1991.
- [106] J.D. Johnston and K. Brandenburg. Wideband coding - perceptual considerations for speech and music. *Advances in Speech Signal Processing*, pages 109-140, 1992.
- [107] U. Jorasz. *Wykłady z psychoakustyki*. Wyd. Naukowe UAM, Poznań, 1998.
- [108] B.H. Juang and Jr A.H.Gray. Multiple stage vector quantization for speech coding. *Proc. Int. Conf. Acoust., Speech, Signal Processing*, vol.1:597-600, 1982.
- [109] G.S. Kang and L.J. Fransen. Application of line-spectrum pairs to low-bit-rate speech encoders. *Int. Congress on Acoustics, Speech and Signal Proc. ICASSP'85*, pages 244-247, 1985.
- [110] A. Kataoka, T. Moriya, and S. Hayashi. Implementation and performance of an 8 kbit/s conjugate structure CELP speech coder. *Int. Congress on Acoustics, Speech and Signal Proc. ICASSP'94*, vol.2:93-96, 1994.

- [111] W.B. Kleijn, D.J. Krasinski, and R.H. Ketchum. An efficient stochastically excited linear predictive coding algorithm for high quality low bit rate transmission of speech. *Speech Communication*, pages 305–316, 1988.
- [112] W.B. Kleijn, D.J. Krasinski, and R.H. Ketchum. Fast methods for the CELP speech coding algorithm. *IEEE Trans. on Acoust., Speech, and Signal Processing*, 38, No. 8:1330–1342, Aug. 1990.
- [113] A.M. Kondozi and B.G. Evans. A high quality voice coder with integrated echo canceller and voice activity detector for mobile satellite applications. *3rd Int. Mobile Satellite Conf. and Exhibition, Pasadena, USA*, June 1993.
- [114] P. Kroon and B.S. Atal. Pitch predictors with high temporal resolution. *Proc. Int. Conf. Acoust., Speech, Signal Processing*, pages 661–664, 1990.
- [115] C. Laflamme, J.P. Adoul, R. Salami, S. Morissette, and P. Mabilieu. 16 kbps wideband speech coding technique based on algebraic CELP. *Proc. Int. Conf. Acoust., Speech, Signal Processing*, pages 13–16, 1991.
- [116] W. LeBlanc, C.Liu, and V. Viswanathan. An enhanced full rate speech coder for digital cellular application. *Proc. Int. Conf. Acoust., Speech, Signal Processing*, pages 569–572, 1996.
- [117] R. Lefebvre, R. Salami, C. Laflamme, and J.P. Adoul. High quality coding of wideband audio signals using transform coded excitation TCX. *Proc. Int. Conf. Acoust., Speech, Signal Processing*, pages I-193–196, 1994.
- [118] W. Li and Y.Q. Zhang. New insights and results on transform domain VQ of images. *Proc. Int. Conf. Acoust., Speech, Signal Processing*, pages V-609–612, 1993.
- [119] W. Li and Y.Q. Zhang. A study on the optimal attributes of transform domain vector quantization for image and video compression. *IEEE Int. Conf. on Communications*, pages 401–405, 1993.
- [120] D. Lin. A toll quality 8 kbps speech coder for PCS applications. In *Proceedings of IEEE Workshop on Speech Coding : Speech Coding for the Network of the Future*, pages 27–28, October 13-15, 1993.
- [121] Y. Linde, A. Buzo, and R.M. Gray. An algorithm for vector quantizer design. *IEEE Trans. on Communications*, COM-28, January 1980.
- [122] S.P. Lloyd. Least squares quantization in PCM. *IEEE Trans. on Information Theory*, IT-28:129–137, Mar. 1982. Praca opublikowana w 1957r jako raport wewnętrzný Bell Laboratories.
- [123] C.P. Lockhoff. Precision adaptive sub-band coding PASC for the digital compact cassette DCC. *IEEE Trans. on Consumer Electronics*, 38, No. 4:784–789, Nov. 1992.
- [124] T.D. Lookabaugh and R. M. Gray. High-resolution quantization theory and the vector quantizer advantage. *IEEE Trans. on Information Theory*, 35, No. 5:1020–1033, September 1989.

- [125] P. Lupini, H. Hassanein, and V. Cuperman. A 2.4 kb/s CELP speech codec with class dependent structure. *Proc. Int. Conf. Acoust., Speech, Signal Processing ICASSP'93*, vol.2:143–146, 1993.
- [126] Y. Mahieux. High quality audio transform coding at 64 kbit/s. *Annales des Télécommunications*, 47, no 3-4, pages 95–106, 1992.
- [127] Y. Mahieux and J.P. Petit. High-quality audio transform coding at 64 kbps. *IEEE Trans. on Communications*, Vol. 42, No. 11:3010–3019, November 1994.
- [128] K.T. Malone and T.R. Fischer. Contour-gain vector quantization. *IEEE Trans. on Acoust., Speech, and Signal Processing*, ASSP-36, June 1988.
- [129] H.S. Malvar. Efficient signal coding with hierarchical lapped transforms. *Proc. Int. Conf. Acoust., Speech, Signal Processing*, pages 1519–1522, 1990.
- [130] H.S. Malvar. *Signal processing with lapped transforms*. Artech House, 1992.
- [131] H.S. Malvar. Lapped transforms for efficient transform/subband coding. *IEEE Trans. on Acoust., Speech, and Signal Processing*, 38, No. 6, June 1990.
- [132] J.D. Markel and A.H. Gray. *Linear prediction of speech*. New-York:Springer, 1976.
- [133] J.S. Marques, J.M. Tribolet, I.M. Trancoso, and L.B. Almeida. Pitch prediction with fractional delays in CELP coding. *Proc. European Conf. on Speech Communication and Technology*, pages 509–512, 1989.
- [134] A.V. McCree and T.P. Barnwell. A new mixed excitation LPC vocoder. *Proc. Int. Conf. Acoust., Speech, Signal Processing*, 1991.
- [135] A.V. McCree and T.P. Barnwell. A mixed excitation LPC vocoder model for low bit rate speech coding. *IEEE Trans. on Speech and Audio Processing*, vol.3:242–250, July 1995.
- [136] A.V. McCree, K. Truong, E.B. George, T.P. Barnwell, and V. Viswanathan. A 2.4kbit/s MELP coder candidate for the new U.S. federal standard. *Proc. Int. Conf. Acoust., Speech, Signal Processing*, pages 200–203, 1996.
- [137] P. Mermelstein. G.722, a new CCITT standard for digital transmission of wideband audio signals. *IEEE Communication Magazine*, 26, no. 1, January 1988.
- [138] S. Miki, K. Mano, H. Ohmuro, and T. Moriya. Pitch synchronous innovation CELP (PSI-CELP). *3rd European Conf. on Speech Commun. and Technology EURO-SPEECH'93*, vol.1:261–264, 1993.
- [139] N. Moreau. *Techniques de compression des signaux*. Masson, Collection technique et scientifique des télécommunications, 1995.
- [140] N. Moreau and P. Dymarski. Codeur CELP à excitation mixte. *12 Coll. GRETSI, Juan-les-Pins*, 1989.

- [141] N. Moreau and P. Dymarski. Mixed excitation CELP coder. *Proc. European Conf. on Speech Communication and Technology EUROSPEECH'89*, pages 322–325, 1989. Présentowana również 2 nd EURASIP Workshop on Medium to Low Rate Speech Coding, Hersbruck, Sept.1989.
- [142] N. Moreau and P. Dymarski. QR factorization in the CELP coder. *IEEE Workshop on Speech Coding - Vancouver'91*, 1991.
- [143] N. Moreau and P. Dymarski. Successive orthogonalizations in the multistage CELP coder. *Proc. Int. Conf. Acoust., Speech, Signal Processing ICASSP'92*, 1992.
- [144] N. Moreau and P. Dymarski. Introduction d'un banc de filtres hiérarchique dans un codeur prédictif. *14 Coll. GRETSI, Juan-les-Pins*, 1993.
- [145] N. Moreau and P. Dymarski. Selection of excitation vectors for the CELP coders. *IEEE Trans. on Speech and Audio Processing*, vol.2, no.1, part 1:29–41, January 1994.
- [146] N. Moreau and P. Dymarski. Codeur audio (20Hz-15kHz) hiérarchique (32-64kbit/s). *Proc. 5èmes Journées d'Etudes et d'Echanges "Compression et Représentation des Signaux Audio-visuels"*, (CORESA'99), Sophia - Antipolis, Juin 1999.
- [147] N. Moreau and P. Dymarski. Low delay coder (< 25 ms) of wideband audio (20 hz-15 khz) scalable from 64 to 32 kbit/s. *Annales de Télécommunications*, pages 493–506, Sep.-Oct. 2000.
- [148] N. Moreau, P. Dymarski, and A. Jbira. Codeur audio (20Hz-15kHz) à 64kbit/s et à faible retard. *Proc. 4èmes Journées d'Etudes et d'Echanges "Compression et Représentation des Signaux Audiovisuels"*, (CORESA'98), Lannion, 9-10 Juin 1998.
- [149] D.G. Morrison. Video compression standards : where we are and how we got there. *Proc. European Signal Processing Conf.*, pages 1709–1715, September 1994.
- [150] B.M. Mouy and P.E. LaNoue. Voice transmission at a very low bit rate on a noisy channel. *Proc. Int. Conf. Acoust., Speech, Signal Processing ICASSP'92*, pages 149–152, 1992.
- [151] C. Murgia, G. Feng, A. Le Guyader, and C. Quinquis. Codage des signaux de parole 20Hz à 15kHz à très faible retard au débit de 64 kbit/s. *15 Coll. GRETSI, Juan-les-Pins*, pages 345–348, 1995.
- [152] P. Noll. Digital audio coding for visual communications. *Proceedings of the IEEE*, Vol. 83, No. 6:925–943, June 1995.
- [153] P. Noll. Wideband speech and audio coding. *IEEE Communications Magazine*, pages 34–44, Nov. 1993.
- [154] P. Noll. MPEG digital audio coding. *IEEE Signal Processing Magazine*, pages 59–81, Sep. 1997.
- [155] Norm ISO/IEC IS13818-7. *MPEG-2 - Advanced Audio Coding AAC*, April 1997.
- [156] Norm ISO/IEC IS14496-3. *Information technology - Very low bitrate audio-visual coding*, 1998.

- [157] Norme internationale ISO/CEI 11172. *Codage de l'image animée et du son associé pour les supports de stockage numérique jusqu'à environ 1,5 Mbit/s*, 1993.
- [158] E. Ordentlich and Y. Shoham. Low-delay CELP coding of wideband speech at 32 kbps. *Proc. Int. Conf. Acoust., Speech, Signal Processing ICASSP'91*, S1.3, 1991.
- [159] E. Paksoy, J. C. DeMartin, A. McCree, C. G. Gerlach, A. Anandakumar, W. M. Lai, and V. Viswanathan. An adaptive multi-rate speech coder for digital cellular telephony. *Proc. Int. Conf. Acoust., Speech, Signal Processing ICASSP'99*, 1999.
- [160] E. Paksoy, K. Srinivasan, and A. Gersho. Variable rate speech coding with phonetic segmentation. *Proc. Int. Conf. Acoust., Speech, Signal Processing ICASSP'93*, vol.2:155–158, 1993.
- [161] K.K. Paliwal and B.S. Atal. Efficient vector quantization of LPC parameters at 24 bits/frame. *Proc. Int. Conf. Acoust., Speech, Signal Processing*, pages 661–664, 1991.
- [162] Y.C. Pati, R. Rezaifar, and P.S. Krishnaprasad. Orthogonal matching pursuit: recursive function approximation with application to wavelet decomposition. *Proc. 27th Asilomar Conf. Signals, Systems and Computers*, pages 40–44, 1993.
- [163] PKNiM. *PN-90/T-05100: Cyfrowe łańcuchy telefoniczne - Wymagania i metody pomiaru wyrazistości logatomowej*, 1997.
- [164] S. Quackenbush. A 7kHz bandwidth, 32 kbps speech coder for ISDN. *Proc. Int. Conf. Acoust., Speech, Signal Processing ICASSP'91*, art. S1.1, 1991.
- [165] L.R. Rabiner and R.W. Schafer. *Digital processing of speech signal*. Prentice-Hall, 1979.
- [166] R.P. Ramachandran and P. Kabal. Pitch prediction filters in speech coding. *IEEE Trans. on Acoust., Speech, and Signal Processing*, 37, No. 4, April 1989.
- [167] S. Ratre. *Product code vector quantizers and their application to speech coding*. PhD thesis, Instytut Telekomunikacji Politechniki Warszawskiej, 1994.
- [168] R.C. Reininger and J.D. Gibson. Backward adaptive lattice and transversal predictors in ADPCM. *IEEE Trans. on Communications*, Vol. 33, Jan. 1985.
- [169] R.C. Rose and T.P. Barnwell. The self excited vocoder - an alternate approach to toll quality at 4800 bps. *Proc. Int. Conf. Acoust., Speech, Signal Processing*, pages 453–456, 1986.
- [170] R.C. Rose and T.P. Barnwell. Design and performance of an analysis-by-synthesis class of predictive speech coders. *IEEE Trans. on Acoust., Speech, and Signal Processing*, 38, No. 9:1489–1503, Sept. 90.
- [171] G. Roy and P. Kabal. Wideband CELP speech coding at 16 kbits/s. *Proc. Int. Conf. Acoust., Speech, Signal Processing ICASSP'91*, art. S1.5, 1991.
- [172] F. Rumsey. Hearing both sides-stereo sound for TV in the UK. *IEE Review*, vol.36, no.5:173–176, May 1990.

- [173] M.J. Sabin. Fixed-shape adaptive gain vector quantization for speech waveform coding. *Speech Communication*, Vol. 8:177–183, 1989.
- [174] M.J. Sabin and R.M. Gray. Product code vector quantizers for waveform and voice coding. *IEEE Trans. on Acoust., Speech, and Signal Processing*, ASSP-32, No. 3:474–488, June 1984.
- [175] M.J. Sabin and R.M. Gray. Global convergence and empirical consistency of the generalized Lloyd algorithm. *IEEE Trans. on Inform. Theory*, IT-32, No. 2:148–155, March 1986.
- [176] R. Salami, C. Laflamme, J. P. Adoul, A. Kataoka, S. Hayashi, T. Moriya, C. Lamblin, D. Massaloux, S. Proust, P. Kroon, and Y. Shoham. Design and description of CS-ACELP: a toll quality 8kb/s speech coder. *IEEE Trans. on Speech and Audio Processing*, vol. 6, No. 2:116–130, March 1998.
- [177] R. Salami, C. Laflamme, J.P. Adoul, A. Kataoka, S. Hayashi, C. Lamblin, D. Massaloux, S. Proust, P. Kroon, and Y. Shoham. Description of the proposed ITU-T 8kbit/s speech coding standard. In *Proceedings of IEEE Workshop on Speech Coding : Speech Coding for Interoperable Global Communications*, pages 3–4, Sept. 20-22, 1995.
- [178] R.A. Salami and D.G. Appleby. A new approach to low bit rate speech coding with low complexity using binary pulse excitation. *IEEE Workshop on Speech Coding for Telecommunication-Vancouver'89*, 1989.
- [179] R.A. Salami, C. Laflamme, and J.P. Adoul. 8 kbit/s ACELP coding of speech with 10ms speech frame - a candidate for CCITT standardization. *Proc. Int. Conf. Acoust., Speech, Signal Processing ICASSP'94*, vol.2:97–100, 1994.
- [180] R.A. Salami, K.H.H. Wong, R. Steele, and D.G. Appleby. Performance of error protected binary pulse excitation coders at 11.4 kb/s over mobile radio channels. *Proc. Int. Conf. Acoust., Speech, Signal Processing*, pages 473–476, 1990.
- [181] G. Schroeder. The standardisation process for the ITU-T 8kbit/s speech codec. In *Proceedings of IEEE Workshop on Speech Coding : Speech Coding for Interoperable Global Communications*, pages 1–2, Sept. 20-22, 1995.
- [182] C.E. Shannon. A mathematical theory of communication. *Bell System Technical Journal*, 27:379–423 and 623–656, 1948.
- [183] C.E. Shannon. Coding theorems for a discrete source with a fidelity criterion. *IRE Nat. Conv. Rec.*, part 4:142–163, 1959.
- [184] W. Skarbek. *Multimedia: algorytmy i standardy kompresji*. Akad. Oficyna Wyd. PLJ, Warszawa, 1998.
- [185] S.M.F. Smyth, J.V. McCanny, and P. Challener. An independent evaluation of the performance of the CCITT G.722 wideband coding recommendation. *Proc. Int. Conf. Acoust., Speech, Signal Processing ICASSP'88*, pages 2544–2547, 1988.
- [186] A.K. Soman and P.P. Vaidyanathan. Coding gain in paraunitary analysis/synthesis systems. *IEEE Trans. on Signal Processing*, Vol. 41, No. 5:1824–1835, May 1993.

- [187] F. Soong and Biing-Hwang Juang. Optimal quantization of LSP parameters using delayed decisions. *Proc. Int. Conf. Acoust., Speech, Signal Processing*, pages 185–188, 1990.
- [188] K. Srinivasan and A. Gersho. Voice activity detector for digital cellular networks. In *Proceedings of IEEE Workshop on Speech Coding : Speech Coding for the Network of the Future*, pages 85–86, October 13-15, 1993.
- [189] P. Strobach. *Linear prediction theory*. Springer, 1990.
- [190] N. Sugamura and F. Itakura. Line spectrum representation of linear predictor coefficients of speech signal and its statistical properties. *Trans. Inst. Electron. Commun. Eng. Japan*, pages 323–340, 1981.
- [191] L.M. Supplee, R.P. Cohn, J.S. Collura, and A.V. McCree. MELP: The new federal standard at 2400 bps. *Proc. Int. Conf. Acoust., Speech, Signal Processing*, pages 1591–1594, 1997.
- [192] J. Szabatin and A. Wojtkiewicz. Blokowe i rekursywne algorytmy estymacji parametrów AR szeregów czasowych. *Rozprawy Elektrotechniczne*, z. 4:993–1046, 1989.
- [193] T. Taniguchi, M. Johnson, and Y. Ohta. Multi-vector pitch-orthogonal LPC : Quality speech with low complexity at rates between 4 and 8 kbps. *Proc. Int. Conf. on Spoken Language Processing*, pages 113–116, 1990.
- [194] T. Taniguchi, M. Johnson, and Y. Ohta. Pitch sharpening for perceptually improved CELP and the sparse-delta codebook for reduced computation. *Proc. Int. Conf. Acoust., Speech, Signal Processing*, pages 241–244, 1991.
- [195] S. Taumi, K. Ozawa, T. Nomura, and M. Serizawa. 13kbps low-delay error-robust speech coding for GSM EFR. In *Proceedings of IEEE Workshop on Speech Coding : Speech Coding for Interoperable Global Communications*, pages 61–62, Sept. 20-22, 1995.
- [196] C. Todd, G. Davidson, M. Davis, L. Fielder, B. Link, and S. Vernon. AC-3 : Flexible perceptual coding for audio transmission and storage. *The 96th Audio Eng. Soc. Convention, Amsterdam*, preprint 3796, February 26- March 1, 1994.
- [197] I.M. Trancoso and B.S. Atal. Efficient search procedures for selecting the optimum innovation in stochastic coders. *IEEE Trans. on Acoust., Speech, and Signal Processing*, 38, No. 3:385–396, March 1990.
- [198] T. Tremain. The government standard linear predictive coding algorithm: LPC-10. *Speech Technology*, pages 40–49, April 1982.
- [199] T.E. Tremain and M.A. Kohler. Philosophy and goals of the D.O.D. 2400bps vocoder selection process. *Proc. Int. Conf. Acoust., Speech, Signal Processing*, pages 1137–1140, 1996.
- [200] K. Tsutsui, H. Suzuki, O. Shimoyoshi, M. Sonohara, K. Agagiri, and R.M. Heddle. ATRAC: Adaptive transform acoustic coding for MiniDisc. *Conf. Rec. Audio Eng. Soc. Convention, San Francisco*, October 1992.
- [201] P. Vary, K. Hellwig, R. Hofmann, R. Sluyter, C. Galand, and M. Rosso. Speech codec for the european mobile radio system. *Proc. Int. Conf. Acoust., Speech, Signal Processing*, pages 227–230, 1988.

- [202] S. Wang and A. Gersho. Phonetically based vector excitation coding of speech at 3.6kbit/s. *Proc. Int. Conf. Acoust., Speech, Signal Processing*, pages 49–52, 1989.
- [203] S. Wang, A. Gersho, and E. Paksoy. Product code vector quantization of LPC parameters. In *Speech and Audio Coding for Wireless and Network Applications*, pages 251–258. Kluwer, 1993.
- [204] S. Wang, A. Sekey, and A. Gersho. Auditory distortion measure for speech coding. *Proc. Int. Conf. Acoust., Speech, Signal Processing*, pages 493–496, 1991.
- [205] V.C. Welch and T.E. Tremain. A new government standard 2400 bps speech coder. In *Proceedings of IEEE Workshop on Speech Coding : Speech Coding for the Network of the Future*, pages 41–42, October 13-15, 1993.
- [206] J.H. Yao, J.J. Shynk, and A. Gersho. Low delay VXC at 8kbit/s with interframe coding. *Proc. Int. Conf. Acoust., Speech, Signal Processing*, pages 45–48, 1992.
- [207] T.Y. Young and T.W. Calvert. *Classification, estimation and pattern recognition*. American Elsevier Publishing Company, New York, 1980.
- [208] P. Zador. Asymptotic quantization error of continuous signals and the quantization dimension. *IEEE Trans. on Information Theory*, vol. IT-28, pages 139–149, March 1982.
- [209] E. Zwicker and E. Feldtkeller. *Psychoacoustique, l'oreille récepteur d'information*. Masson, Collection technique et scientifique des télécommunications, Traduit de l'allemand par C. Sorin, 1981.
- [210] P. Zwierko and P. Dymarski. Modem do pracy poprzez łącze krótkofalowe z możliwością transmisji ucyfrowionej mowy. *Krajowe Sympozjum Telekomunikacji KST'2000*, 2000.