

**Analiza sygnałów biomedycznych - wykład**

**dr inż. Krzysztof Duda**

**Kraków 2010**



## Przedmowa

Analiza sygnałów biomedycznych dotyczy metod przetwarzania pozwalających wydobyć informację użyteczną z zarejestrowanego sygnału. Ze względu na warunki akwizycji, informacja w sygnale użytecznym przeważnie nie jest dostępna wprost i konieczna jest np. poprawa jakości sygnału lub zmiana jego reprezentacji poprzez zastosowanie właściwej transformacji. Często badania medyczne dostarczają tak dużej ilości danych, że wymagane są automatyczne algorytmy analizy, ponieważ manualna analiza tych danych przez lekarza byłaby zbyt czasochłonna lub nawet niewykonalna. W praktyce, w analizie sygnałów biomedycznych, stosowane są bardzo skomplikowane algorytmy przetwarzania sygnałów, a struktury systemów komercyjnych są niejawne. Niemniej jednak, nawet najbardziej skomplikowany system jest zbudowany z elementów podstawowych takich jak filtracja czy analiza częstotliwościowa. Niniejszy skrypt przedstawia właśnie te elementy podstawowe, których znajomość umożliwia projektowanie złożonych, dedykowanych algorytmów analizy sygnałów biomedycznych uwzględniających specyfikę konkretnego przypadku, a także świadome użytkowanie dostępnej aparatury i oprogramowania.

Niniejszy skrypt został napisany dla specjalności *Pomiary technologiczne i biomedyczne* studiów II stopnia na kierunku *Elektrotechnika*. Wykład z przedmiotu *Analiza sygnałów biomedycznych* trwa jeden semestr, a jego uzupełnieniem jest laboratorium komputerowe. Biorąc pod uwagę, że dla wielu studentów wykład ten może być pierwszym spotkaniem z zagadnieniami analizy i przetwarzania sygnałów, spora część skryptu dotyczy zagadnień podstawowych, do których należą: twierdzenie o próbkowaniu, projektowanie filtrów analogowych i cyfrowych, filtracja cyfrowa, analiza częstotliwościowa i czasowo-częstotliwościowa.

W odróżnieniu od typowych podręczników z podstaw analizy i przetwarzania sygnałów w skrypcie uwzględniono również metody szczególnie przydatne w zastosowaniu do sygnałów biomedycznych. Omówiono algorytmy filtracji adaptacyjnej RLS i LMS, transformację falkową w wersji klasycznej i liftingowej dla sygnałów jedno i dwuwymiarowych, algorytmy kodowania entropijnego oraz podstawy analizy obrazów wraz z zagadnieniem tomografii komputerowej.

Opisane w skrypcie metody analizy sygnałów nie odnoszą się tylko do sygnałów biomedycznych, ale w zasadzie do wszystkich sygnałów pomiarowych. Jako przykład można podać tomografię komputerową jednoznacznie kojarzoną z diagnostyką medyczną, jednak narzędzie to jest również wykorzystywane do oceny jakości detali przemysłowych, np. odlewów.

Zakres wiadomości podanych w skrypcie jest zaledwie wstępem do rozległej dyscypliny analizy i przetwarzania sygnałów. Niemniej jednak zakres wybranych zagadnień dostarcza solidnej wiedzy podstawowej, którą następnie można rozwijać samodzielne w wybranym kierunku. Spis literatury podaje źródła, na podstawie których powstał niniejszy skrypt. Do samodzielnego poszerzania wiedzy i umiejętności z zakresu analizy i przetwarzania sygnałów szczególnie polecam książkę prof. Zielińskiego [7] (lub wydania wcześniejsze), w której większość omawianych zagadnień teoretycznych zilustrowana jest programami w Matlabie, co znakomicie ułatwia ich zrozumienie. Nieocenionym źródłem wiedzy praktycznej i przykładów zastosowań są instrukcje obsługi do bibliotek Matlaba [8-11] dostępne na stronie [www.mathworks.com](http://www.mathworks.com). Pozycje [7-11] i cytowania w nich zawarte poruszają większość zagadnień spotykanych w praktyce analizy i przetwarzania sygnałów.

Uzupełnieniem niniejszego skryptu jest zestaw ćwiczeń przeznaczonych do realizacji w trakcie laboratorium komputerowego. Zajęcia w tym laboratorium polegają na samodzielnym programowaniu wybranych algorytmów analizy i przetwarzania sygnałów w

środowisku Matlab. W trakcie zajęć, pisane przez studentów, programy są na bieżąco sprawdzane i konsultowane przez prowadzącego.

Głównym celem laboratorium jest praktyczna ilustracja zagadnień teoretycznych omawianych na wykładzie. Studenci zdobywają umiejętność programowania typowych algorytmów analizy i przetwarzania sygnałów. W przypadku niektórych zadań umieszczone rozwiązania w postaci programów bądź funkcji Matlaba. Podane programy i funkcje należy traktować jako standardy programowania i w miarę możliwości modyfikować je do rozwiązywania zbliżonych problemów.

# Spis treści

Spis ważniejszych skrótów i oznaczeń.....	7
1. Sygnały dyskretne - ciągi .....	9
1.1 Podstawowe operacje na ciągach .....	10
1.2 Ciągi deterministyczne .....	10
1.3 Ciągi losowe .....	15
2. Układy dyskretne.....	21
2.1 Układ liniowy .....	21
2.2 Układ niezmienny względem przesunięcia .....	22
2.3 Układ przyczynowy.....	22
2.4 Stabilność .....	22
2.5 Układ liniowy, niezmienny względem przesunięcia.....	22
2.6 Własności układów LTI .....	24
2.7 Reprezentacja częstotliwościowa sygnałów i układów dyskretnych .....	25
2.8 Reprezentacja Fouriera ciągów .....	25
3. Układy analogowe.....	33
4. Transformacja Z .....	39
4.1 Analiza własności układów LTI.....	40
4.2 Charakterystyki częstotliwościowe układu dyskretnego.....	46
4.3 Układ wszechprzepustowy .....	47
4.4 Układ odwrotny .....	48
4.5 Dekompozycja minimalnofazowo-wszechprzepustowa .....	49
5. Próbkowanie sygnałów .....	51
5.1 Rekonstrukcja sygnału ciągłego z reprezentacji dyskretnej.....	57
5.2 Metoda niezmiennej odpowiedzi impulsowej .....	58
5.3 Zmiana częstotliwości próbkowania .....	60
5.4 Cyfrowy tor analizy i przetwarzania sygnałów .....	70
6. Projektowanie filtrów analogowych.....	71
6.1 Filtr Butterwortha .....	71
6.2 Filtry Czebyszewa .....	74
6.3 Filtr eliptyczny .....	77
6.4 Porównanie charakterystyk filtrów analogowych .....	77
6.5 Transformacja częstotliwości - filtry LP, HP, BP, BS .....	79
7. Projektowanie filtrów cyfrowych.....	83
7.1 Transformacja biliniowa .....	83
7.2 Filtry FIR.....	88
7.2.1 Układy LTI z liniową charakterystyką fazową .....	91
7.2.2 Projektowanie filtrów FIR metodą okien .....	94
7.2.3 Optymalny filtr FIR - algorytm Parks-McClellana .....	98
7.2.4 Filtr Hilberta .....	99
7.2.5 Dyskretne układy różniczkujące .....	103
8. Dyskretne przekształcenie Fouriera .....	105
8.1 Dyskretny Szereg Fouriera.....	105

8.2 Dyskretne przekształcenie Fouriera .....	106
8.2.1 Splot kołowy .....	108
8.2.2 Obliczanie splotu liniowego za pomocą splotu kołowego .....	110
8.2.3 Splot sekcjonowany .....	110
8.2.4 Obliczanie DFT algorytmem Goertzla .....	112
8.2.5 Transformacja Chirp-Z .....	113
8.2.6 Algorytmy FFT .....	115
8.2.7 DFT sygnałów o wartościach rzeczywistych .....	121
9. Analiza sygnałów z wykorzystaniem DFT .....	123
9.1 Analiza DFT sygnałów sinusoidalnych.....	124
9.2 Interpolowane DFT .....	129
9.3 Krótkoczasowa transformacja Fouriera, spektrogram .....	136
9.4 Analiza stacjonarnych sygnałów losowych - periodogram.....	141
10. Aproksymacja sygnałów, filtry optymalne i filtry adaptacyjne .....	145
10.1 Aproksymacja sygnałów w przestrzeni wielowymiarowej .....	145
10.2 Filtracja optymalna w sensie najmniejszych kwadratów .....	149
10.3 Filtry adaptacyjne RLS.....	152
10.4 Estymacja w sensie minimum średnich kwadratów.....	155
10.5 Filtry adaptacyjne LMS.....	157
10.6 Zastosowania filtrów adaptacyjnych.....	159
11. Transformacja falkowa.....	163
11.1 Ciągła transformacja falkowa.....	163
11.2 Dyskretna transformacja falkowa.....	166
11.2.1 DWT w wersji predykcyjnej .....	174
12. Kodowanie entropijne .....	179
12.1 Kodowanie Huffmana .....	179
12.2 Kodowanie arytmetyczne .....	182
12.3 Kodowanie słownikowe .....	186
13. Obrazy cyfrowe .....	189
13.1 Filtracja 2D.....	189
13.1.1 Filtry specjalne .....	190
13.2 Transformacja Fouriera obrazów .....	194
13.3 Projektowanie filtrów 2D metodą okien .....	195
13.4 DFT obrazów.....	201
13.5 DWT obrazów .....	202
13.6 Tomografia komputerowa .....	203
Literatura .....	209

## Spis ważniejszych skrótów i oznaczeń

W niniejszym skrypcie przyjęto konwencję bazującą na skrótach nazw angielskich, np. stosowane jest oznaczanie FIR (*Finite Impulse Response*), a nie SOI (skończona odpowiedź impulsowa).

1D, 2D	Sygnal jedno-, dwuwymiarowy ( <i>Dimensional</i> ).
BP	Filtr pasmowoprzepustowy ( <i>bandpass</i> )
BS	Filtr pasmowozaporowy ( <i>bandstop</i> )
$C$	Zbiór liczb całkowitych.
CT	Tomografia komputerowa ( <i>Computed Tomography</i> ).
CWT	Ciągła transformacja falkowa ( <i>Continuous Wavelet Transform</i> ).
DC	Składowa stała ( <i>Direct Current</i> ).
DFT	Dyskretne przekształcenie Fouriera ( <i>Discrete Fourier Transform</i> ).
DWT	Dyskretna transformacja falkowa ( <i>Discrete Wavelet Transform</i> ).
FFT	Szybka transformacja Fouriera ( <i>Fast Fourier Transform</i> ).
FIR	Filtr o skończonej odpowiedzi impulsowej ( <i>Finite Impulse Response</i> ).
$F_s$	Częstotliwość próbkowania w hercach.
FT	Transformacja Fouriera ( <i>Fourier Transform</i> ).
HP	Filtr górnoprzepustowy ( <i>highpass</i> )
IIR	Filtr o nieskończonej odpowiedzi impulsowej ( <i>Infinite Impulse Response</i> ).
LP	Filtr dolnoprzepustowy ( <i>lowpass</i> ).
LS	Metoda najmniejszych kwadratów ( <i>Least-Squares</i> ).
MMS	Metoda najmniejszych średnich kwadratów ( <i>Minimum Mean-Square</i> ).
SA	Sygnal analityczny.
STFT	Krótkoczasowa Transformacja Fouriera ( <i>Short-Time Fourier Transform</i> ).
$X(e^{j\omega})$	Widmo ciągłe sygnału dyskretnego $x[n]$ .
$X(j\Omega)$	Widmo sygnału ciągłego $x_c(t)$ .
$x[n]$	Sygnal dyskretny, ciąg.
$x_c(t)$	Sygnal ciągły, analogowy.
$\omega$	Pulsacja (częstotliwość) sygnału dyskretnego $x[n]$ w radianach.
$\Omega$	Pulsacja (częstotliwość) sygnału ciągłego $x_c(t)$ w radianach/sekundę.



## 1. Sygnały dyskretne - ciągi

Sygnały ciągłe (analogowe) są określone dla zmiennej ciągłej (we wszystkich chwilach czasu). Sygnały dyskretne są określone dla wybranych chwil czasu i tworzą ciągi liczb. Dodatkowo, wartości sygnałów dyskretnych mogą być kwantowane. Skwantowane sygnały dyskretne są określane jako sygnały cyfrowe.

Sygnały takie jak dźwięk czy obraz mogą mieć reprezentację ciągłą lub dyskretną i w pewnych warunkach obie te reprezentacje są równoważne, tzn. zawierają tyle samo informacji.

Sygnały dyskretne mogą być otrzymywane przez próbkowanie sygnałów ciągłych lub jako wyjście z układu dyskretnego.

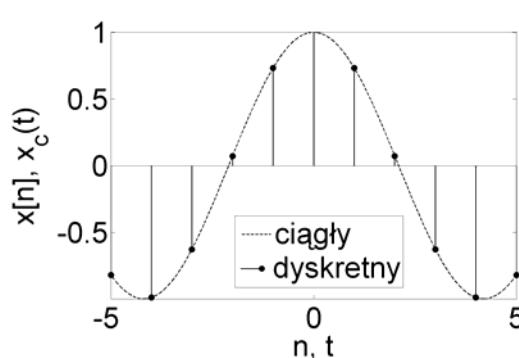
Sygnały dyskretne tworzą nieskończone ciągi liczb. Zapis  $x[n]$  (1.1) oznacza wszystkie wyrazy ciągu, tzn. dla indeksów  $n$ , które są liczbami całkowitymi i zmieniają się od minus do plus nieskończoności. Indeks  $n$  jest więc numerem (pozycją) danego wyrazu w ciągu.

$$x = \{x[n]\}, \quad -\infty < n < \infty, \quad n \in C, \quad \text{np. } x = \{\dots, x[-1], x[0], x[1], \dots\} = x[n]. \quad (1.1)$$

Jeżeli sygnał dyskretny  $x[n]$  powstał przez próbkowanie sygnału ciągłego  $x_c(t)$ , to ma on w dyskretnych chwilach czasu  $nT$  te same wartości, co sygnał ciągły:

$$x[n] = x_c(nT), \quad \frac{1}{T} = F_s, \quad -\infty < n < \infty, \quad n \in C, \quad (1.2)$$

gdzie  $T$  - czas (okres) próbkowania w sekundach,  $F_s$  - częstotliwość próbkowania (*sampling frequency*) w hercach. Zależność (1.2) jest zilustrowana na rys.1.1. W przypadku próbkowania sygnału ciągłego  $n$  oznacza numer próbki. Jak widać z zapisu (1.2), ciągi dyskretne są, w niniejszym skrypcie, oznaczane przez nazwę ciągu i nawias kwadratowy, w którym podawana jest nazwa indeksu, natomiast sygnały ciągłe oznaczane są przez nazwę funkcji i nawias okrągły, w którym podawana jest nazwa zmiennej.



Rys.1.1 Utworzenie sygnału dyskretnego przez próbkowanie sygnału ciągłego.

Ciągi nieskończone mają głównie znaczenie teoretyczne i służą do wyprowadzania zależności analitycznych. W praktyce przetwarzanie sygnałów dyskretnych realizowane jest za pomocą komputerów, które operują na danych skończonych. Dlatego zakłada się skończoną liczbę próbek sygnału  $N$ , wyrażenie (1.1) przyjmuje więc postać:

$$x = \{x[n]\}, \quad 0 < n \leq N-1, \quad n \in C, \quad \text{np. } x = \{x[0], x[1], \dots, x[N-1]\} = x[n]. \quad (1.3)$$

Sygnal  $x[n]$ , zdefiniowany przez (1.3), jest to wektor zawierający  $N$  elementów. Przetwarzanie sygnałów dyskretnych zajmuje się więc wektorami, lub ogólniej, macierzami próbek.

## 1.1 Podstawowe operacje na ciągach

Iloczyn i suma dwóch ciągów  $x[n]$  i  $y[n]$  są zdefiniowane jako iloczyny, bądź sumy odpowiednich próbek (wyrazów ciągu), np. dla  $x=[1, 2, 3]$ ,  $y=[4, 5, 6]$  iloczyn  $x \cdot y = [4, 10, 18]$ . Mnożenie ciągu przez liczbę oznacza pomnożenie każdego z jego wyrazów przez tę liczbę.

Ciąg  $y[n]$  jest przesuniętym (opóźnionym) ciągiem  $x[n]$ , jeżeli zachodzi:

$$y[n] = x[n - n_0], \quad n \in C. \quad (1.4)$$

## 1.2 Ciągi deterministyczne

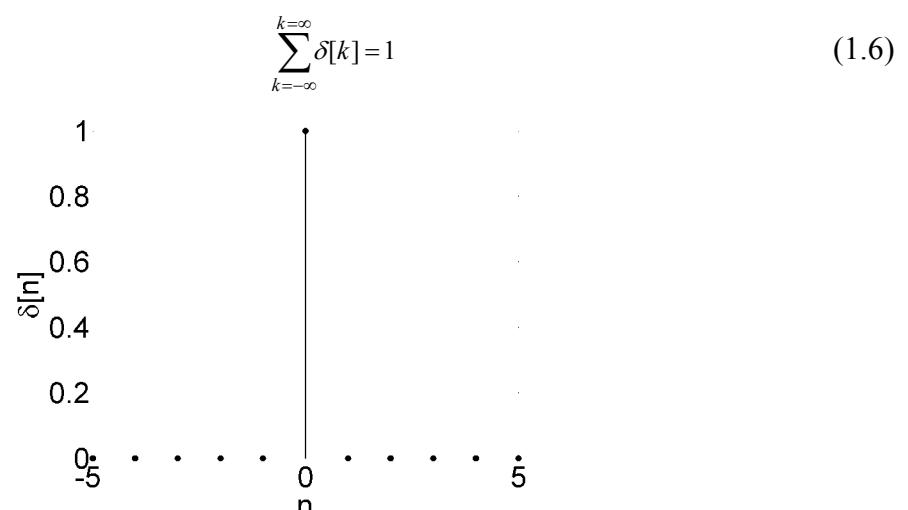
Istnieją ciągi, które ze względu na swoje własności mają szczególne znaczenie w przetwarzaniu sygnałów. Należą do nich zdefiniowane poniżej: impuls jednostkowy, skok jednostkowy, ciąg eksponentjalny, ciąg sinusoidalny i zespolony ciąg eksponentjalny.

### 1.2.1 Impuls jednostkowy

Impuls jednostkowy zdefiniowany jest następująco:

$$\delta[n] = \begin{cases} 0, & n \neq 0, \\ 1, & n = 0. \end{cases} \quad (1.5)$$

Fragment impulsu jednostkowego, dla  $-5 \leq n \leq 5$  jest przedstawiony na rys.1.2. Suma wszystkich wyrazów ciągu impulsu jednostkowego wynosi 1, tzn.:



Rys.1.2 Fragment impulsu jednostkowego.

Dowolny ciąg może być przedstawiony jako suma poprzesuwanych i przeskalowanych impulsów jednostkowych:

$$x[n] = \sum_{k=-\infty}^{\infty} x[k] \delta[n - k]. \quad (1.7)$$

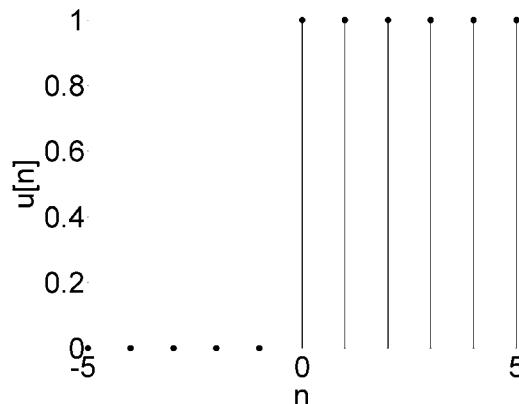
Zależność (1.7) zostanie wykorzystana do wyprowadzenia splotu liniowego. Dla klasy układów LTI, opisanych w rozdziale 2, odpowiedź układu na impuls jednostkowy całkowicie określa własności tego układu. Jest to tzw. odpowiedź impulsowa układu.

### 1.2.2 Skok jednostkowy

Skok jednostkowy zdefiniowany jest następująco:

$$u[n] = \begin{cases} 1, & n \geq 0, \\ 0, & n < 0. \end{cases} \quad (1.8)$$

Fragment skoku jednostkowego, dla  $-5 \leq n \leq 5$  jest przedstawiony na rys.1.3.



Rys.1.3 Fragment skoku jednostkowego.

Skok jednostkowy można przedstawić w postaci sumy poprzesuwanych impulsów jednostkowych

$$u[n] = \sum_{k=0}^{\infty} \delta[n-k] \text{ lub } u[n] = \sum_{k=-\infty}^n \delta[k]. \quad (1.9)$$

Impuls jednostkowy można przedstawić w postaci różnicy dwóch skoków jednostkowych:

$$\delta[n] = u[n] - u[n-1]. \quad (1.10)$$

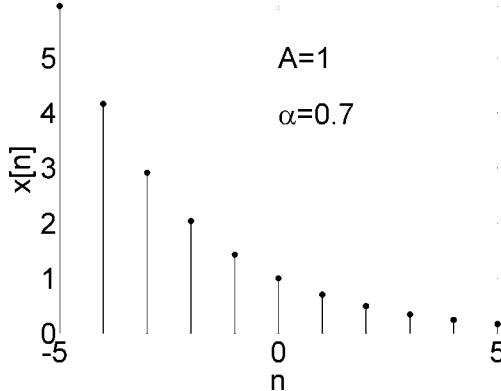
Skok jednostkowy jest powszechnym sygnałem testowym. Odpowiedź układu na skok jednostkowy jest to tzw. odpowiedź skokowa układu.

### 1.2.3 Ciąg wykładniczy

Ciąg wykładniczy zdefiniowany jest następująco:

$$x[n] = A\alpha^n, \quad -\infty < n < \infty, \quad A, \alpha \in \mathbb{R}. \quad (1.11)$$

Parametry  $A$  i  $\alpha$  przyjmują wartości ze zbioru liczb rzeczywistych. Fragment ciągu wykładniczego, dla  $-5 \leq n \leq 5$ ,  $A=1$  i  $\alpha=0.7$  jest przedstawiony na rys.1.4. Odpowiedzi impulsowe stabilnych układów LTI mają charakter malejących (gasących) ciągów wykładniczych.



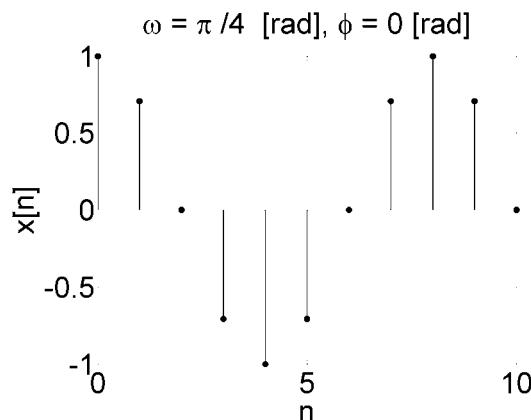
Rys.1.4 Fragment ciągu wykładniczego.

#### 1.2.4 Ciąg sinusoidalny

Ciąg sinusoidalny zdefiniowany jest następująco:

$$x[n] = A \cos(\omega n + \phi), \quad -\infty < n < \infty, \quad (1.12)$$

gdzie  $\omega$  jest pulsacją w radianach, a  $\phi$  jest kątem fazowym w radianach. W definicji (1.12) użyto funkcji kosinus, a nie sinus, w celu zachowania zgodności z większością podręczników. Definicję (1.12) można również zapisać jako  $x[n] = A \cos(\omega n + \phi) = A \sin(\omega n + \phi_s)$ ,  $\phi_s = \phi + \pi/2$ . W niniejszym skrypcie 'ciąg sinusoidalny' odnosi się do kształtu przebiegu czasowego sygnału, który może być opisany funkcją sinus lub kosinus. Ze względu na okresowość funkcji sinusoidalnych ciągi  $x[n] = A \cos(\omega n + \phi)$  i  $x[n] = A \cos((\omega + 2\pi)n + \phi)$  nie są rozróżnialne, ponieważ  $A \cos(\omega n + 2\pi n + \phi) = A \cos(\omega n + \phi)$ , dlatego zakres zmienności pulsacji wynosi  $\langle 0, 2\pi \rangle$  lub  $\langle -\pi, \pi \rangle$ . Fragment ciągu sinusoidalnego dla  $0 \leq n \leq 10$  jest przedstawiony na rysunku 1.5.



Rys.1.5 Fragment ciągu sinusoidalnego (1.12),  $x[n]=\cos((\pi/4)n)$ ,  $n=0,1,2,\dots$

Dla ciągu sinusoidalnego szybkość oscylacji rośnie, gdy  $\omega$  rośnie od  $\omega=0$  do  $\omega=\pi$ , a następnie maleje wraz ze wzrostem  $\omega$  od  $\omega=\pi$  do  $\omega=2\pi$ . Własność ta zilustrowana jest na rys.1.6, na którym lewa kolumna pokazuje ciąg sinusoidalny o pulsacjach z przedziału od 0 do  $\pi$ , a prawa ciąg sinusoidalny o pulsacjach z przedziału od  $\pi$  do  $2\pi$ . Dla porównania linią ciągłą zaznaczono sygnał ciągły o tych samych pulsacjach, którego szybkość oscylacji rośnie wraz ze wzrostem pulsacji. Jak widać z rys.1.6 ciąg sinusoidalny o pulsacji cyfrowej  $7\pi/4$  ma takie same wartości jak ciąg sinusoidalny o pulsacji cyfrowej  $\pi/4$ .

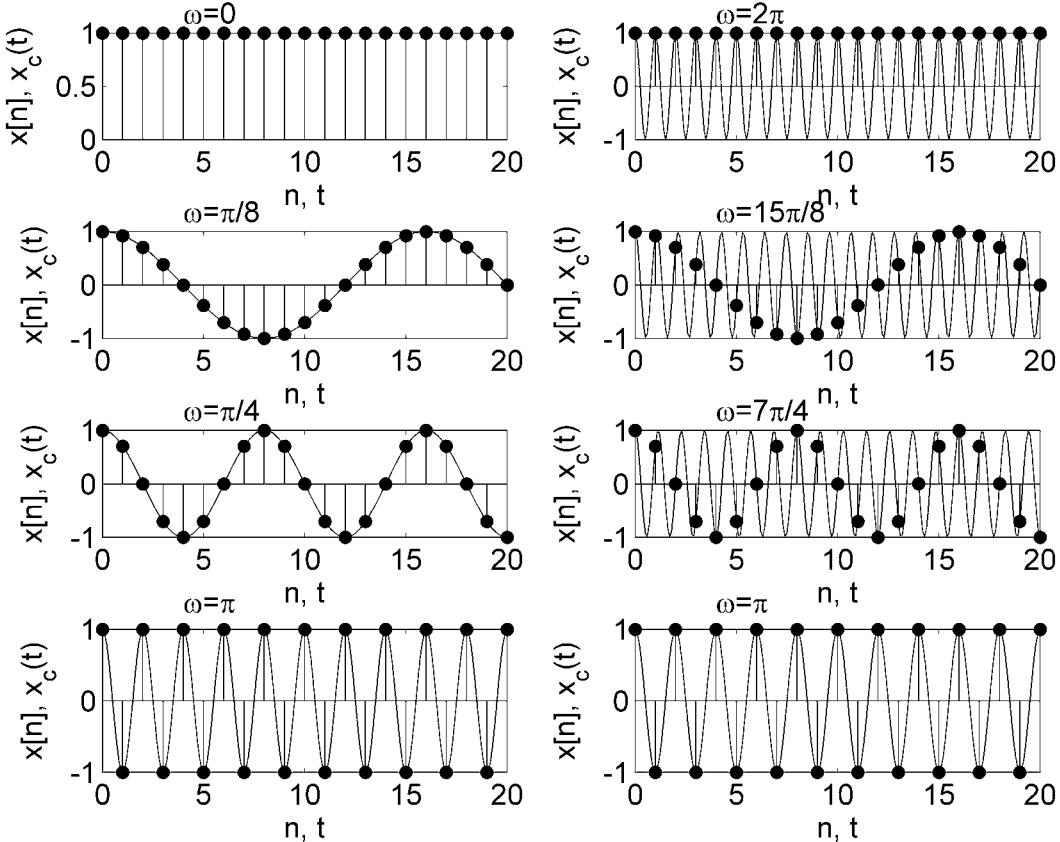
Z rys.1.6 wynika, że zakres pulsacji ciągu sinusoidalnego wynosi  $\langle 0, \pi \rangle$ .

W przypadku próbkowania sygnału ciągłego, częstotliwość sygnału sinusoidalnego można określić w hercach. Kolejne próbki ciągu  $x[n]$  są wyliczane wg (1.2) jako wartości sygnału ciągłego:

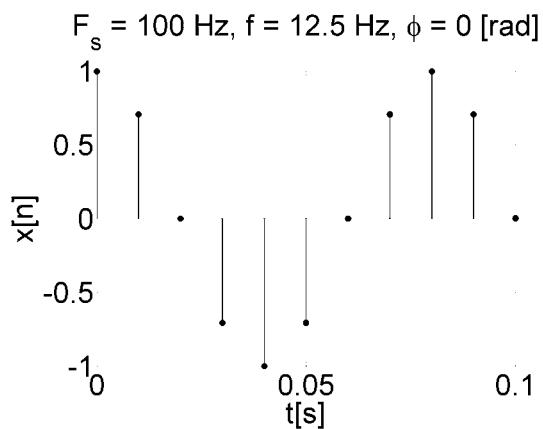
$$x[n] = A \cos(\omega t + \phi), \quad \omega = 2\pi f, \quad t = n / F_s, \quad (1.13)$$

gdzie  $f$  jest częstotliwością sygnału w hercach. Dokonując podstawień we wzorze (1.13) otrzymujemy:

$$x[n] = A \cos\left(2\pi \frac{f}{F_s} n + \phi\right). \quad (1.14)$$



Rys. 1.6 Szybkość oscylacji ciągu sinusoidalnego.



Rys. 1.7 Fragment ciągu sinusoidalnego (1.14),  $x[n] = \cos(2\pi(12.5/100)n)$ ,  $n=0,1,2,\dots$

Ponieważ zakres pulsacji ciągu sinusoidalnego wynosi  $\langle 0, \pi \rangle$  (co pokazano na rys.1.6) więc stosunek  $f/F_s$  w (1.14) musi być mniejszy niż 0.5, czyli częstotliwość próbkowania musi być przynajmniej dwa razy większa niż częstotliwość sinusoidalnego sygnału ciągłego.

Rys.1.7 przedstawia fragment ciągu sinusoidalnego powstałego przez próbkowanie sygnału ciągłego o częstotliwości 12.5 Hz z częstotliwością 100 Hz. Oś  $OX$  na rys.1.7 wyskalowana jest w sekundach, a nie w indeksach próbek jak na rys.1.5.

Pomiędzy częstotliwością sygnału  $f$  w hercach i pulsacją  $\omega$  radianach zachodzi następująca zależność:

$$f = \frac{\omega}{2\pi} F_s . \quad (1.15)$$

### 1.2.5 Zespolony ciąg eksponentjalny

Zespolony ciąg eksponentjalny zdefiniowany jest następująco:

$$x[n] = A\alpha^n = |A| |\alpha|^n e^{j(\omega n + \phi)}, \quad A = |A| e^{j\phi}, \quad \alpha = |\alpha| e^{j\omega} \quad (1.16)$$

dla  $|\alpha|=1$  otrzymujemy:

$$x[n] = A\alpha^n = |A| e^{j(\omega n + \phi)} = |A| \cos(\omega n + \phi) + j |A| \sin(\omega n + \phi) \quad (1.17)$$

gdzie  $\omega$  jest pulsacją w radianach lub [rad/próbkę], a  $\phi$  jest fazą w radianach.

Podobnie jak ciągi sinusoidalne, zespolone ciągi eksponentjalne o częstotliwościach  $\omega n$  i  $(\omega+2\pi)n$  są nieroróżnicalne (w odróżnieniu do sygnałów ciągłych):

$$x[n] = Ae^{j(\omega+2\pi)n} = Ae^{j\omega n} e^{j2\pi n} = Ae^{j\omega n}, \quad (1.18)$$

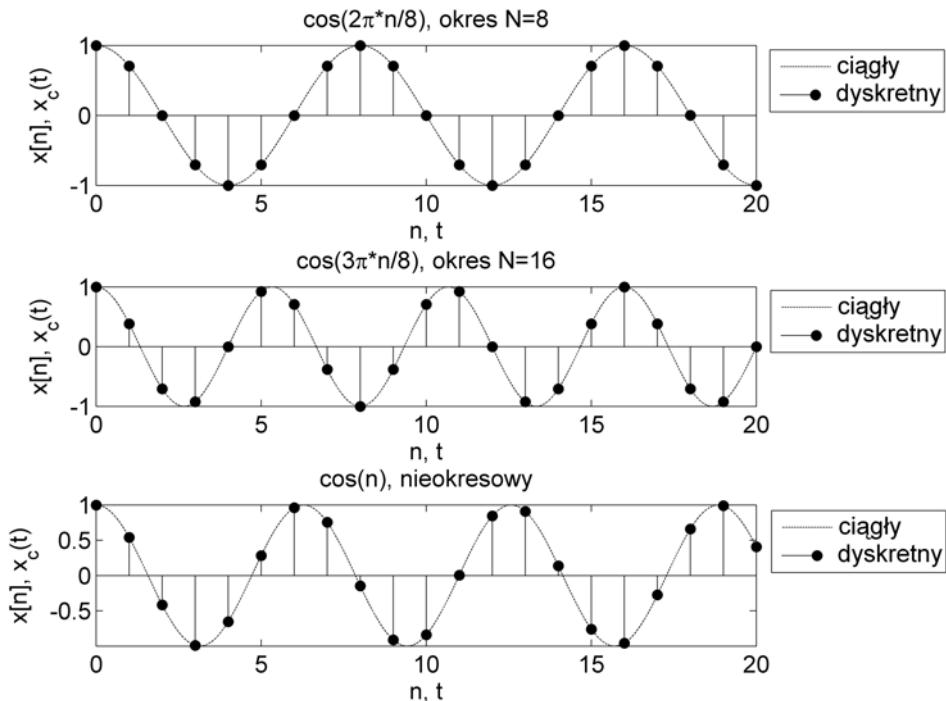
dlatego przy analizie ciągów w postaci  $Ae^{j\omega n}$  wystarczy rozpatrzyć tylko przedział częstotliwości o długości  $2\pi$ , np.  $-\pi < \omega \leq \pi$  lub  $0 \leq \omega < 2\pi$ .

### 1.2.6 Ciągi okresowe

Dla dyskretnego ciągu okresowego o okresie  $N$  zachodzi:

$$x[n] = x[n+N]. \quad (1.19)$$

Dla ciągu sinusoidalnego warunek (1.19) oznacza  $A \cos(\omega n + \phi) = A \cos(\omega n + \omega N + \phi)$ , co pociąga  $\omega N = 2\pi k$ ,  $k \in C$ . Dla ciągu eksponentjalnego  $Ae^{j\omega n}$  okresowość wymaga spełnienia  $e^{j\omega(n+N)} = e^{j\omega n}$ , co również zachodzi dla  $\omega N = 2\pi k$ ,  $k \in C$ . W rezultacie ciągi sinusoidalny i eksponentjalny niekoniecznie muszą być okresowe w  $n$  z okresem  $2\pi/\omega$  i w zależności od  $\omega$  mogą nawet nie być okresowe. Rys.1.8 pokazuje przykład dwóch ciągów sinusoidalnych okresowych i jednego ciągu sinusoidalnego nieokresowego. Dla porównania zaznaczono również ciągle sygnały sinusoidalne o tych samych pulsacjach, które są zawsze okresowe.



Rys.1.8 Okresowość ciągu sinusoidalnego.

### 1.3 Ciągi losowe

W przypadku sygnałów deterministycznych każda próbka ciągu jest jednoznacznie określona przez regułę (np. wyrażenie matematyczne lub tabelę danych). Dla sygnałów losowych (stochastycznych) konkretna realizacja procesu stochastycznego jest opisana poprzez funkcję gęstości prawdopodobieństwa. Zapis sygnału losowego oznacza zapis jednej z wielu możliwych реализациj sygnału.

Przykładem procesu stochastycznego może być rzut monetą w dyskretnych chwilach czasu  $n$ ,  $-\infty < n < \infty$ . Wartość ciągu losowego wynosi  $x[n]=1$ , gdy wypadnie awers i  $x[n]=-1$ , gdy wypadnie rewers. Jeśli prawdopodobieństwo otrzymania 1 wynosi  $p$  to prawdopodobieństwo otrzymania -1 wynosi  $1-p$ .

Wartość  $x[n]$  może być interpretowana jako wartość zmiennej losowej  $x_n$ . Zbiór zmiennych losowych  $\{x_n\}$  dla  $-\infty < n < \infty$  łącznie z opisem probabilistycznym każdej z tych zmiennych stanowi definicję procesu stochastycznego ( $x_n=\{-1,1\}$ ,  $p_n=\{1-p,p\}$ ).

Dany ciąg wartości  $\{x[n]\}$  dla  $-\infty < n < \infty$  jest realizacją procesu stochastycznego i jest nazywany ciągiem próbковym tego procesu. Ciągów takich jest nieskończoność wiele. Przykładowa realizacja tego procesu może być następującą:  $x[n]=\{\dots, 1, 1, -1, 1, -1, -1, 1, 1, 1, \dots\}$ . Zbiór wszystkich realizacji nazywany jest przestrzenią ciągów próbkovych.

#### 1.3.1 Opis procesu stochastycznego

Dystrybuanta zdefiniowana jest następująco:

$$P_{x_n}(x_n, n) = \Pr\{x_n \leq x_n\}, \quad (1.20)$$

gdzie  $x_n$  - jest zmienną losową,  $x_n$  - jest określona wartością  $x_n$ , a  $\Pr\{\cdot\}$  oznacza prawdopodobieństwo. Dystrybuanta określa, więc prawdopodobieństwo tego, że zmienna losowa przyjmie wartość mniejszą lub równą niż pewna wartość ustalona. Dla przykładu rzutu monetą dystrybuanta wynosi:

$$P_{x_n}(x_n, n) = \begin{cases} 1, & x_n \geq 1 \\ 1-p, & -1 \leq x_n < 1 \\ 0, & x_n < -1 \end{cases} \quad (1.21)$$

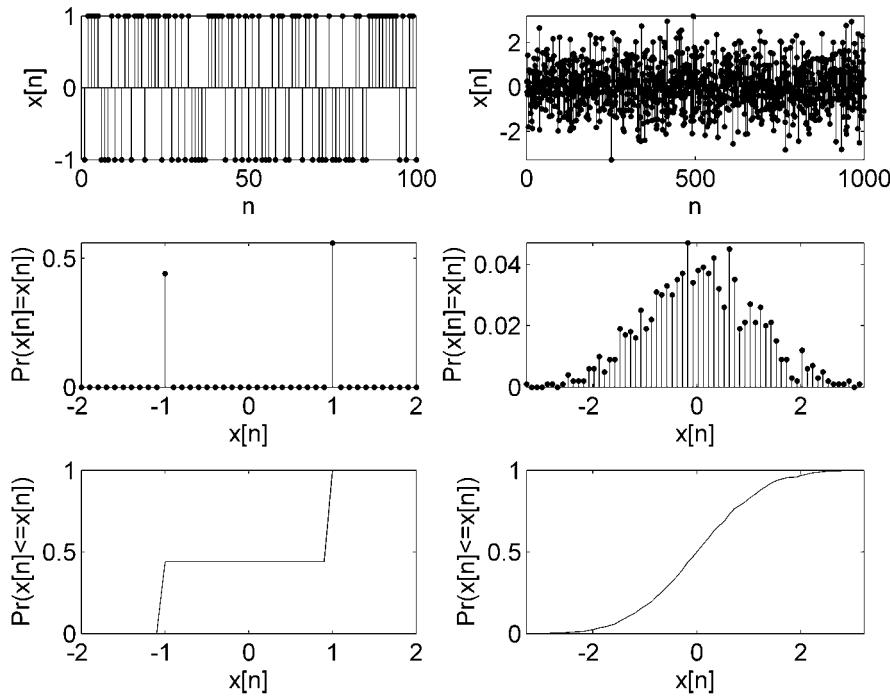
Zgodnie z (1.21), prawdopodobieństwo tego, że w eksperymencie z monetą wypadnie wartość mniejsza niż -1 wynosi zero, a tego że wypadnie wartość mniejsza lub równa 1 wynosi 1.

Funkcja gęstości prawdopodobieństwa dla ciągłej zmiennej losowej  $x_n$  jest to pochodna cząstkowa z dystrybuanty po wartości ustalonej:

$$p_{x_n}(x_n, n) = \frac{\partial P_{x_n}(x_n, n)}{\partial x_n}. \quad (1.22)$$

Dystrybuantę można, więc przedstawić jako całkę z gęstości prawdopodobieństwa.

$$P_{x_n}(x_n, n) = \int_{-\infty}^{x_n} p_{x_n}(x_n, n) dx_n. \quad (1.23)$$



Rys.1.9 Przykładowe realizacje procesów losowych oraz ich histogramy i dystrybuanty.

Dla dyskretniej zmiennej losowej  $x_n$  definiuje się masę prawdopodobieństwa:

$$p_{x_n}(x_n, n) = \Pr\{x_n = x_n\}, \quad (1.24)$$

która estymuje się (szacuje) na podstawie realizacji za pomocą histogramu dla wartości (lub przedziałów)  $x_n$ . Dystrybuanta dla dyskretniej zmiennej losowej  $x_n$  jest to skumulowana suma masy prawdopodobieństwa:

$$P_{x_n}(x_n, n) = \sum_{x \leq x_n} p_{x_n}(x_n, n). \quad (1.25)$$

Dla przykładu rzutu monetą masa prawdopodobieństwa wynosi:

$$p_{x_n}(x_n, n) = \begin{cases} p, & x_n = 1 \\ 1-p, & x_n = -1 \\ 0, & x_n \neq -1 \text{ i } x_n \neq 1 \end{cases}. \quad (1.26)$$

Rys.1.9 przedstawia przykładowe realizacje procesów losowych oraz ich histogramy i dystrybuanty dla doświadczenia z monetą (kolumna lewa) i liczb pseudolosowych o rozkładzie normalnym (kolumna prawa).

Wartość oczekiwana (przeciętna) dla zmiennej ciągłej definiuje się w sposób następujący:

$$m_{x_n} = E[x_n] = \int_{-\infty}^{\infty} x_n p_{x_n}(x_n, n) dx_n, \quad (1.27)$$

a dla zmiennej dyskretniej (skokowej):

$$E[x_n] = \sum_{x_n} x_n p_{x_n}(x_n, n). \quad (1.28)$$

Jeżeli wszystkie zmienne losowe  $x_n$  są jednakowo prawdopodobne, (1.28) upraszcza się do średniej arytmetycznej. Dla rzutu monetą wartość oczekiwana wynosi:

$$E[x_n] = 1 \cdot \Pr\{x_n = 1\} + (-1) \cdot \Pr\{x_n = -1\} = p - 1 + p = 2p - 1. \quad (1.29)$$

Dla wartości oczekiwanych zachodzą zależności:

$$E[x_n + y_m] = E[x_n] + E[y_m], \quad (1.30)$$

$$E[ax_n] = aE[x_n]. \quad (1.31)$$

Jeżeli dwie zmienne losowe są linowo niezależne (nieskorelowane), to:

$$E[x_n y_m] = E[x_n] E[y_m]. \quad (1.32)$$

Wszystkie zmienne losowe tworzące proces stacjonarny mają stałą, niezależną od  $n$ , wartość oczekowaną  $m_x$ .

Wartość średniokwadratowa (moment zerowy rzędu drugiego, moc średnia) dla zmiennej ciągłej zdefiniowana jest następująco:

$$E[x_n^2] = \int_{-\infty}^{\infty} x_n^2 p_{x_n}(x_n, n) dx_n \quad (1.33)$$

a dla zmiennej dyskretniej:

$$E[x_n^2] = \sum_{x_n} x_n^2 p_{x_n}(x_n, n). \quad (1.34)$$

Wariancja (moment centralny rzędu drugiego) zdefiniowana jest zależnością:

$$\sigma_{x_n}^2 = E[(x_n - m_{x_n})^2] = E[x_n^2] - m_{x_n}^2 . \quad (1.35)$$

Wariancja jest więc proporcjonalna do stopnia rozrzutu zmiennej losowej wokół jej wartości średniej. Pierwiastek kwadratowy z wariancji jest to odchylenie standardowe. Odchylenie standardowe jest również proporcjonalne do rozrzutu zmiennej losowej wokół jej wartości średniej i jest wyrażone w tych samych jednostkach, co zmienna losowa.

Ciąg autokorelacji i autokowariancji ciągłej zmiennej są zdefiniowane odpowiednio:

$$\phi_{xx}(n, m) = E[x_n x_m^*] = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} x_n x_m^* p_{x_n, x_m}(x_n, n, x_m, m) dx_n dx_m , \quad (1.36)$$

$$\gamma_{xx}(n, m) = E[(x_n - m_{x_n})(x_m - m_{x_m})^*] = \phi_{xx}(n, m) - m_{x_n} m_{x_m} . \quad (1.37)$$

Ciąg korelacji wzajemnej i kowariancji wzajemnej zmiennych ciągłych są zdefiniowane odpowiednio:

$$\phi_{xy}(n, m) = E[x_n y_m^*] = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} x_n y_m^* p_{x_n, y_m}(x_n, n, y_m, m) dx_n dy_m \quad (1.38)$$

$$\gamma_{xy}(n, m) = E[(x_n - m_{x_n})(y_m - m_{y_m})^*] = \phi_{xy}(n, m) - m_{x_n} m_{y_m} \quad (1.39)$$

Gwiazdka we wzorach oznacza sprzężenie zespolone. Jak widać z powyższych równań, kowariancja jest to korelacja liczona z sygnału po odjęciu wartości oczekiwanej (średniej).

Dla pojedynczej realizacji dyskretnego procesu stochastycznego, średnią czasową procesu stochastycznego wyznacza się z zależności:

$$\langle x[n] \rangle = \lim_{N \rightarrow \infty} \frac{1}{N} \sum_{n=0}^{N-1} x[n] \quad (1.40)$$

a w praktyce, dla skończonego rozmiaru danych:

$$\boxed{\langle x[n] \rangle_N = \frac{1}{N} \sum_{n=0}^{N-1} x[n]} . \quad (1.41)$$

Dla pojedynczej realizacji dyskretnego procesu stochastycznego czasowy ciąg autokorelacji wyznacza się z zależności:

$$\langle x[n], x^*[n+m] \rangle = \lim_{N \rightarrow \infty} \frac{1}{N} \sum_{n=0}^{N-1} x[n] x^*[n+m] \quad (1.42)$$

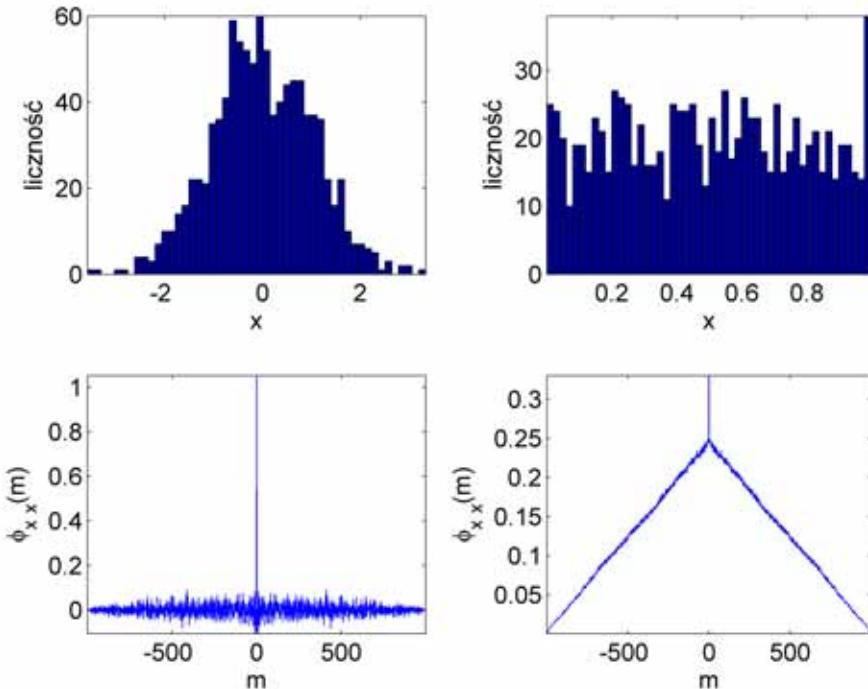
a w praktyce, dla skończonego rozmiaru danych:

$$\boxed{\langle x[n], x^*[n+m] \rangle_N = \frac{1}{N} \sum_{n=0}^{N-1} x[n] x^*[n+m]} . \quad (1.43)$$

Dla sygnałów ergodycznych równania (1.40) i (1.42) przyjmują wartości stałe w tym sensie, że średnie czasowe prawie wszystkich ciągów próbkoowych są równe tej samej stałej. Ponadto są one równe odpowiednim wartościom przeciętnym (uśrednionym po przestrzeni), tj. dla dowolnego pojedynczego ciągu próbkoowego  $\{x[n]\}, -\infty < n < \infty$  zachodzi:

$$\langle x_n \rangle = \lim_{N \rightarrow \infty} \frac{1}{N} \sum_{n=0}^{N-1} x_n = E[x_n] = m_x, \quad (1.44)$$

$$\langle x_n, x_{n+m}^* \rangle = \lim_{N \rightarrow \infty} \frac{1}{N} \sum_{n=0}^{N-1} x_n x_{n+m}^* = E[x_n x_{n+m}^*] = \phi_{xx}(m). \quad (1.45)$$



Rys.1.10 Histogramy oraz oszacowane za pomocą (1.43) funkcje autokorelacji przykładowych realizacji procesów losowych: rozkład normalny (kolumna lewa) i jednostajny (kolumna prawa).

Funkcja korelacji odgrywa bardzo ważną rolę w analizie sygnałów, jest ona stosowana m.in. w modelowaniu parametrycznym i filtracji adaptacyjnej, a także jako kryterium podobieństwa sygnałów, czy do wykrywania okresowości w zakłóconych danych. Pod względem obliczeniowym (1.43) składa się z sumy iloczynów, czyli tzw. operacji MAC (*Multiple and Accumulate*). Obliczenia (1.43) przebiegają w ten sposób, że sygnały są przesuwane względem siebie o jedną próbkę przez wskaźnik  $m$ , a następnie wyznaczana jest suma iloczynów wszystkich odpowiadających sobie próbek. W przypadku autokorelacji największa wartość zostanie, więc uzyskana dla przesunięcia  $m=0$ , czyli gdy próbki są mnożone przez siebie same, tj. podnoszone do kwadratu.

Rys.1.10 Przedstawia histogramy i funkcje autokorelacji estymowane wg (1.43) dla realizacji liczb pseudolosowych o rozkładzie normalnym (kolumna lewa) i jednostajnym (kolumna prawa). W obu przypadkach funkcja autokorelacji jest zbliżona do impulsu jednostkowego. Składowa liniowa (trójkąt) w wykresie autokorelacji dla rozkładu jednostajnego spowodowana jest niezerową wartością średnią sygnału, czyli składową stałą.



## 2. Układy dyskretnie

Układ (system) z czasem dyskretnym definiuje się matematycznie jako operator, który odwzorowuje ciąg wejściowy  $x[n]$  w ciąg wyjściowy  $y[n]$ :

$$y[n] = T\{x[n]\} . \quad (2.1)$$

Przykładem układu dyskretnego jest idealny układ opóźniający:

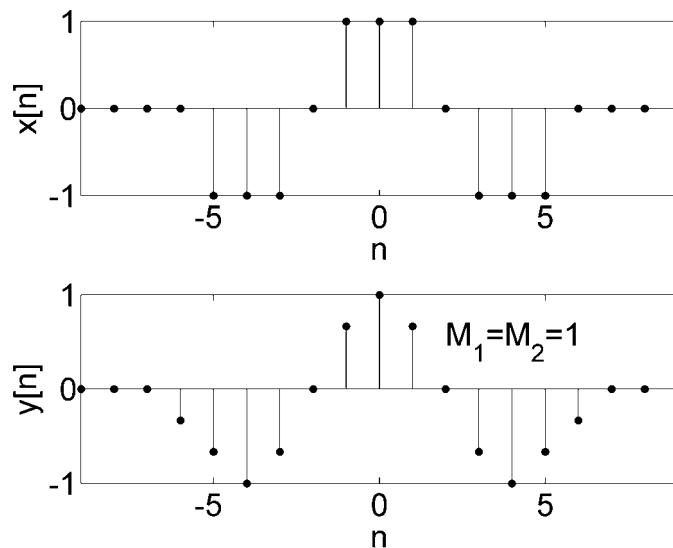
$$y[n] = x[n - n_d], \quad -\infty < n < \infty, \quad n_d = const, \quad (2.2)$$

gdzie:  $n_d$  jest opóźnieniem wnoszonym przez układ.

Innym przykładem jest średnia ruchoma:

$$\begin{aligned} y[n] &= \frac{1}{M_1 + M_2 + 1} \sum_{k=-M_1}^{M_2} x[n - k] = \\ &= \frac{1}{M_1 + M_2 + 1} (x[n + M_1] + x[n + M_1 - 1] + \dots + x[n] + x[n - 1] + \dots + x[n - M_2]) \end{aligned} . \quad (2.3)$$

Zgodnie z (2.3)  $n$ -ty wyraz  $y[n]$  jest średnią arytmetyczną ( $M_1+M_2+1$ ) wyrazów  $x[n]$  wokół wyrazu  $n$ . Obliczenia te zilustrowane są na rys.2.1.



Rys.2.1 Przykład obliczania średniej ruchomej wg (2.3).

### 2.1 Układ liniowy

Zgodnie z (2.1) można zdefiniować nieskończenie wiele układów dyskretnych, jednak analiza teoretyczna ich własności jest praktycznie ograniczona tylko do podzbioru układów liniowych. Układ liniowy spełnia postulat addytywności:

$$T\{x_1[n] + x_2[n]\} = T\{x_1[n]\} + T\{x_2[n]\} \quad (2.4)$$

i skalowalności (homogeniczności):

$$T\{ax[n]\} = aT\{x[n]\} \quad (2.5)$$

Warunki (2.4) i (2.5) można wyrazić w postaci zasady superpozycji:

$$T\{ax_1[n] + bx_2[n]\} = aT\{x_1[n]\} + bT\{x_2[n]\}, \quad (2.6)$$

zgodnie z którą układ jest liniowy, jeżeli odpowiedź układu na sumę przeskalowanych wymuszeń jest równa sumie przeskalowanych odpowiedzi układu na każde wymuszenie z osobna.

## 2.2 Układ niezmienny względem przesunięcia

Układ jest niezmienny względem przesunięcia (*time invariant system*), jeżeli przesunięcie (opóźnienie) ciągu wejściowego  $x[n]$  powoduje analogiczne przesunięcie ciągu wyjściowego  $y[n]$ , tzn:

$$\text{jeżeli } y[n] = T\{x[n]\} \quad \text{to} \quad y[n-n_0] = T\{x[n-n_0]\}. \quad (2.7)$$

## 2.3 Układ przyczynowy

Układ jest przyczynowy, jeżeli wartość wyrazu ciągu wyjściowego  $y[n]$  o indeksie  $n=n_0$  zależy tylko od wyrazów o indeksach  $n \leq n_0$  (tj. bieżącego i przeszłych). Przykładem układu przyczynowego jest wyznaczanie różnicy wstecz (*backward difference*) zdefiniowane jako  $y[n] = x[n] - x[n-1]$ , natomiast układ obliczania różnicy wprzód (*forward difference*) zdefiniowany jako  $y[n] = x[n+1] - x[n]$  jest niepryczynowy.

## 2.4 Stabilność

Układ jest stabilny, jeżeli każde ograniczone pobudzenie powoduje ograniczoną odpowiedź. Ciąg  $x[n]$  jest ograniczony, jeżeli istnieje skończona wartość dodatnia  $B_x$  taka, że  $|x[n]| \leq B_x < \infty, \forall n$ .

## 2.5 Układ liniowy, niezmienny względem przesunięcia

Układy liniowe niezmienne względem przesunięcia LTI (*linear time invariant system*), stanowią podgrupę wszystkich układów dyskretnych. Spełnienie postulatu liniowości i niezmienności względem przesunięcia ułatwia analizę teoretyczną układów LTI.

Rozważmy dowolny układ LTI opisany równaniem (2.1) oraz przedstawmy ciąg wejściowy  $x[n]$  w postaci sumy poprzesuwanych i przeskalowanych impulsów jednostkowych zgodnie z (1.7), wówczas otrzymujemy:

$$y[n] = T \left\{ \sum_{k=-\infty}^{\infty} x[k] \delta[n-k] \right\}. \quad (2.8)$$

Korzystając z zasady superpozycji (2.6) przepisujemy (2.8) w postaci:

$$y[n] = \sum_{k=-\infty}^{\infty} x[k] T\{\delta[n-k]\} = \sum_{k=-\infty}^{\infty} x[k] h_k[n], \quad (2.9)$$

gdzie  $h_k[n] = T\{\delta[n-k]\}$  jest to odpowiedź impulsowa układu, tj. odpowiedź na  $\delta[n-k]$  (impuls jednostkowy o indeksie  $n=k$ ).

Odpowiedź układu liniowego na dowolne wymuszenie może być wyznaczona na podstawie (2.9) jeżeli znana jest odpowiedź impulsowa układu  $h_k[n]$ .

Z niezmienności względem przesunięcia, wynika że, jeżeli  $h[n]$  jest odpowiedzią na  $\delta[n]$ , to odpowiedzią na  $\delta[n-k]$  jest  $h[n-k]$  i (2.9) można zapisać:

$$y[n] = \sum_{k=-\infty}^{\infty} x[k]h[n-k] = x[n]*h[n], \quad (2.10)$$

czyli  $y[n]$  jest sumą po  $k$  poprzesuwanych ( $h[n-k]$ ) i przeskalowanych (przez  $x[n]$ ) odpowiedzi impulsowych układu.

Układ liniowy niezmienny względem przesunięcia jest całkowicie określony przez odpowiedź impulsową  $h[n]$ . Równanie (2.10) jest to splot dyskretny ( $y[n]$  jest splotem dyskretnym  $x[n]$  i  $h[n]$ ). Gwiazdka '\*' w (2.10) oznacza operację splotu, a nie iloczyn.

Równanie (2.10) jest jednym z najważniejszych równań w cyfrowym przetwarzaniu sygnałów. Podobnie jak korelacja (1.43), splot dyskretny składa się z sumy iloczynów.

Bezpośrednie liczenie splotu dyskretnego wg (2.10) wymaga zastosowania dwóch pętli: 1) zewnętrznej po  $n$  i 2) wewnętrznej po  $k$ . Rys.2.2 przedstawia obliczenia splotu dyskretnego wg (2.10) dla dwóch nieskończonych ciągów, które mają elementy niezerowe tylko na wymienionych pozycjach.

$$\begin{aligned} h[n] &= [\dots 0, h[1], h[2], h[3], 0, \dots] = [\dots 0, 1, 3, 5, 0, \dots], \quad x[n] = [\dots 0, x[1], x[2], x[3], 0, \dots] = [\dots 0, 2, 4, 6, 0, \dots] \\ y[n] &= \sum_{k=-\infty}^{\infty} x[k]h[n-k] \\ y[1] &= \sum_{k=-\infty}^{\infty} x[k]h[1-k] = \dots 0 + x[1]h[0] + x[2]h[-1] + x[3]h[-2] + 0 \dots = \dots 0 + 2 \cdot 0 + 4 \cdot 0 + 6 \cdot 0 + 0 \dots = 0 \\ y[2] &= \sum_{k=-\infty}^{\infty} x[k]h[2-k] = \dots 0 + x[1]h[1] + x[2]h[0] + x[3]h[-1] + 0 \dots = \dots 0 + 2 \cdot 1 + 4 \cdot 0 + 6 \cdot 0 + 0 \dots = 2 \\ y[3] &= \sum_{k=-\infty}^{\infty} x[k]h[3-k] = \dots 0 + x[1]h[2] + x[2]h[1] + x[3]h[0] + 0 \dots = \dots 0 + 2 \cdot 3 + 4 \cdot 1 + 6 \cdot 0 + 0 \dots = 10 \\ y[4] &= \sum_{k=-\infty}^{\infty} x[k]h[4-k] = \dots 0 + x[1]h[3] + x[2]h[2] + x[3]h[1] + 0 \dots = \dots 0 + 2 \cdot 5 + 4 \cdot 3 + 6 \cdot 1 + 0 \dots = 28 \\ y[5] &= \sum_{k=-\infty}^{\infty} x[k]h[5-k] = \dots 0 + x[1]h[4] + x[2]h[3] + x[3]h[2] + 0 \dots = \dots 0 + 2 \cdot 0 + 4 \cdot 5 + 6 \cdot 3 + 0 \dots = 38 \\ y[6] &= \sum_{k=-\infty}^{\infty} x[k]h[6-k] = \dots 0 + x[1]h[5] + x[2]h[4] + x[3]h[3] + 0 \dots = \dots 0 + 2 \cdot 0 + 4 \cdot 0 + 6 \cdot 5 + 0 \dots = 30 \\ y[n] &= x[n]*h[n] = [\dots 0, 2, 10, 28, 38, 30, 0, \dots] \end{aligned}$$

Rys.2.2 Obliczenia splotu dyskretnego wg (2.10).

Z rys.2.2 widać, że znak '-' w  $h[n-k]$  powoduje odwrócenie w czasie tego ciągu. Obliczenia splotu dyskretnego przebiegają więc następująco: jeden z ciągów jest odwracany w czasie, a następnie ciągi są przesuwane względem siebie o jedną próbke i liczona jest suma iloczynów odpowiednich próbek dla każdego przesunięcia (podobnie jak w przypadku korelacji). Obliczenia te są zilustrowane na rys.2.3.

$$\begin{array}{r} h[-n] = [\dots 0, \quad 5, \quad 3, \quad 1, \quad 0, \dots] \\ x[n] = [\dots 0, \quad 2, \quad 4, \quad 6, \quad 0, \dots] \\ \hline y[1] = 1 \cdot 2 = 2 \end{array}$$

$$\begin{array}{r} h[-n] = [\dots 0, \quad 5, \quad 3, \quad 1, \quad 0, \dots] \\ x[n] = [\dots 0, \quad 2, \quad 4, \quad 6, \quad 0, \dots] \\ \hline y[2] = 3 \cdot 2 + 1 \cdot 4 = 10 \end{array}$$

$$\begin{array}{r} h[-n] = [\dots 0, \quad 5, \quad 3, \quad 1, \quad 0, \dots] \\ x[n] = [\dots 0, \quad 2, \quad 4, \quad 6, \quad 0, \dots] \\ \hline y[3] = 5 \cdot 2 + 3 \cdot 4 + 1 \cdot 6 = 28 \end{array}$$

$$\begin{array}{r} h[-n] = [\dots 0, \quad 5, \quad 3, \quad 1, \quad 0, \dots], \\ x[n] = [\dots 0, \quad 2, \quad 4, \quad 6, \quad 0, \dots] \\ \hline y[4] = 5 \cdot 4 + 3 \cdot 6 = 38 \end{array}$$

$$\begin{array}{r} h[-n] = [\dots 0, \quad 5, \quad 3, \quad 1, \quad 0, \dots], \\ x[n] = [\dots 0, \quad 2, \quad 4, \quad 6, \quad 0, \dots] \\ \hline y[5] = 5 \cdot 6 = 30 \end{array}$$

Rys.2.3 Obliczenia splotu dyskretnego.

## 2.6 Własności układów LTI

Splot dyskretny posiada następujące własności:

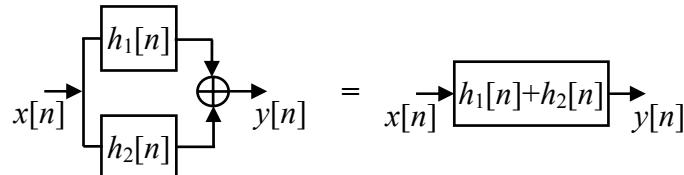
1. Przemienność:  $x[n]*h[n] = h[n]*x[n]$  (2.11)

2. Rozdzielność względem dodawania:  $x[n]*(h_1[n] + h_2[n]) = x[n]*h_1[n] + x[n]*h_2[n]$  (2.12)

3. Połączenie kaskadowe

$$x[n] \rightarrow \boxed{h_1[n]} \rightarrow \boxed{h_2[n]} \rightarrow y[n] = x[n] \rightarrow \boxed{h_2[n]} \rightarrow \boxed{h_1[n]} \rightarrow y[n] = x[n] \rightarrow \boxed{h_1[n]*h_2[n]} \rightarrow y[n]$$

4. Połączenie równoległe



5. Układ odwrotny  $h_i[n]$  (*inverse*),  $h[n]*h_i[n] = \delta[n]$

$$x[n] \rightarrow \boxed{h[n]} \rightarrow \boxed{h_i[n]} \rightarrow x[n]$$

## 2.7 Reprezentacja częstotliwościowa sygnałów i układów dyskretnych

Rozważmy układ LTI zdefiniowany przez odpowiedź impulsową  $h[n]$ , na którego wejście podany jest zespolony ciąg eksponentowy w postaci  $x[n] = e^{j\omega n}$ ,  $-\infty < n < \infty$ . Odpowiedź układu jest określona przez splot dyskretny (2.10):

$$y[n] = \sum_{k=-\infty}^{\infty} h[k] e^{j\omega(n-k)} = e^{j\omega n} \left( \sum_{k=-\infty}^{\infty} h[k] e^{-j\omega k} \right) = e^{j\omega n} H(e^{j\omega}). \quad (2.13)$$

Jak widać z (2.13) dla układów liniowych niezmiennych względem przesunięcia odpowiedzią na wejściowy sygnał sinusoidalny jest sygnał sinusoidalny o tej samej częstotliwości, którego amplituda i faza zależą od własności układu.  $H(e^{j\omega})$  określa zmianę amplitudy i fazy zespolonego ciągu wykładniczego  $e^{j\omega n}$  i jest nazywana odpowiedzią częstotliwościową układu. Jest to funkcja zespolona i może być przedstawiona w następującej postaci:

$$H(e^{j\omega}) = \operatorname{Re}\{H(e^{j\omega})\} + j \operatorname{Im}\{H(e^{j\omega})\} = |H(e^{j\omega})| e^{j\arg\{H(e^{j\omega})\}}, \quad (2.14)$$

gdzie  $|H(e^{j\omega})|$  jest to charakterystyka częstotliwościowo-amplitudowa, lub krócej amplitudowa, a  $\arg\{|H(e^{j\omega})|\}$  jest to charakterystyka częstotliwościowo-fazowa, lub krócej fazowa. Odpowiedź częstotliwościowa układu dyskretnego jest zawsze okresowa z okresem  $2\pi$ :

$$H(e^{j(\omega+2\pi)}) = \sum_{k=-\infty}^{\infty} h[k] e^{-j(\omega+2\pi)k} = \sum_{k=-\infty}^{\infty} h[k] e^{-j\omega k} e^{-j2\pi k} = H(e^{j\omega}), \quad (2.15)$$

dlatego wystarczy określić ją dla przedziału o długości  $2\pi$ .

## 2.8 Reprezentacja Fouriera ciągów

Odwrotna transformacja Fouriera sygnału dyskretnego (synteza) jest zdefiniowana:

$$x[n] = \frac{1}{2\pi} \int_{-\pi}^{\pi} X(e^{j\omega}) e^{j\omega n} d\omega. \quad (2.16)$$

Transformacja Fouriera sygnału dyskretnego (analiza) jest zdefiniowana:

$$X(e^{j\omega}) = \sum_{n=-\infty}^{\infty} x[n] e^{-j\omega n}. \quad (2.17)$$

Dla pokazania, że (2.16) i (2.17) tworzą parę przekształceń odwrotnych, podstawmy  $X(e^{j\omega})$  do wyrażenia na  $x[n]$ , otrzymujemy:

$$x[n] = \frac{1}{2\pi} \int_{-\pi}^{\pi} \left( \sum_{m=-\infty}^{\infty} x[m] e^{-j\omega m} \right) e^{j\omega n} d\omega = \sum_{m=-\infty}^{\infty} x[m] \left( \frac{1}{2\pi} \int_{-\pi}^{\pi} e^{j\omega(n-m)} d\omega \right), \quad (2.18)$$

funkcje bazowe  $e^{j\omega n}$  są ortogonalne i całka w nawiasach (2.18) ma wartość:

$$\frac{1}{2\pi} \int_{-\pi}^{\pi} e^{j\omega(n-m)} d\omega = \frac{\sin(\pi(n-m))}{\pi(n-m)} = \begin{cases} 1, & m=n \\ 0, & m \neq n \end{cases} = \delta[n-m], \quad (2.19)$$

ostatecznie:

$$x[n] = \sum_{m=-\infty}^{\infty} x[m] \delta[n-m] = x[n]. \quad (2.20)$$

Równie syntezы (2.16) przedstawia ciąg  $x[n]$  jako sumę nieskończonej liczby zespolonych sinusoid w postaci  $e^{j\omega n}$  przeskalowanych w amplitudzie i przesuniętych w fazie przez liczby zespolone  $X(e^{j\omega})$ .  $X(e^{j\omega})$  określa, więc "ilość" każdej z zespolonych sinusoid w ciągu  $x[n]$ . Pulsacja  $\omega$  jest zmienną ciągłą w przedziale o długości  $2\pi$ .

Transformacja Fouriera jest funkcją zespoloną zmiennej  $\omega$ , podobnie jak w przypadku (2.14) można wyznaczyć charakterystykę amplitudową i częstotliwościową transformacji Fouriera:

$$X(e^{j\omega}) = \operatorname{Re}\{X(e^{j\omega})\} + j \operatorname{Im}\{X(e^{j\omega})\} = |X(e^{j\omega})| e^{j\arg\{X(e^{j\omega})\}}. \quad (2.21)$$

Transformacja Fouriera jest określana jako widmo (*spectrum*) Fouriera lub widmo. Charakterystyka fazowa nie jest określona jednoznacznie, ponieważ do  $\arg\{X(e^{j\omega})\}$  można dodać dowolną krotność  $2\pi$  bez zmiany  $\arg\{X(e^{j\omega})\}$ .

Porównując (2.13) z (2.17) można stwierdzić, że odpowiedź częstotliwościowa układu liniowego niezmiennego w czasie jest transformatą Fouriera jego odpowiedzi impulsowej. Odpowiedź impulsową można wyznaczyć z odpowiedzi częstotliwościowej za pomocą odwrotnej transformaty Fouriera.

Transformata Fouriera istnieje dla wszystkich sygnałów, dla których nieskończona suma (2.17) jest ograniczona:

$$|X(e^{j\omega})| < \infty, \quad \forall \omega. \quad (2.22)$$

Warunkiem wystarczającym zbieżności  $X(e^{j\omega})$  jest bezwzględna sumowalność ciągu  $x[n]$ :

$$|X(e^{j\omega})| = \left| \sum_{n=-\infty}^{\infty} x[n] e^{-j\omega n} \right| \leq \sum_{n=-\infty}^{\infty} |x[n]| |e^{-j\omega n}| \leq \sum_{n=-\infty}^{\infty} |x[n]| < \infty. \quad (2.23)$$

Ponieważ ciąg stabilny jest z definicji bezwzględnie sumowalny, więc wszystkie ciągi stabilne mają transformatę Fouriera (w szczególności wszystkie ciągi o skończonej długości są bezwzględnie sumowalne). Wszystkie układy (systemy) stabilne mają odpowiedź częstotliwościową (w szczególności wszystkie układy o skończonej odpowiedzi impulsowej są bezwzględnie sumowalne).

Istnieją sygnały, które nie są bezwzględnie sumowalne, ale są sumowalne w kwadracie, tzn.:

$$\sum_{n=-\infty}^{\infty} |x[n]|^2 < \infty. \quad (2.24)$$

W takim wypadku definiuje się zbieżność średnio-kwadratową:

$$\lim_{M \rightarrow \infty} \int_{-\pi}^{\pi} |X(e^{j\omega}) - X_M(e^{j\omega})|^2 d\omega = 0, \quad X(e^{j\omega}) = \sum_{n=-\infty}^{\infty} x[n] e^{-j\omega n}, \quad X_M(e^{j\omega}) = \sum_{n=-M}^{M} x[n] e^{-j\omega n}, \quad (2.25)$$

co oznacza, że wartość błędu  $|X(e^{j\omega}) - X_M(e^{j\omega})|$  może nie maleć do zera dla wszystkich wartości  $\omega$ , gdy  $M \rightarrow \infty$ , ale całkowita energia błędu (2.25) zmierza do zera z  $M \rightarrow \infty$ .

### 2.8.1 Transformata Fouriera ciągu stałego

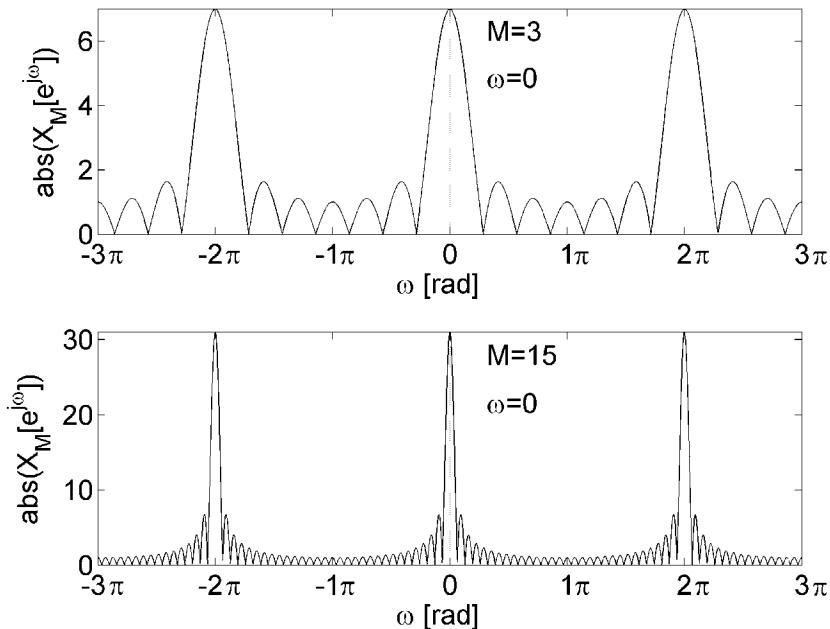
Ciąg stały o nieskończonej długości  $x[n] = 1, -\infty < n < \infty$ , nie jest bezwzględnie sumowalny i nie jest sumowalny w kwadracie. Transformata Fouriera ciągu stałego składa się z poprzesuwanych, ciągłych impulsów Diraca (o nieskończonej amplitudzie, zerowej szerokości i jednostkowym polu):

$$X(e^{j\omega}) = \sum_{r=-\infty}^{\infty} 2\pi\delta(\omega + 2\pi r). \quad (2.26)$$

Dla ciągu jednostkowego o skończonej długości  $x_M[n] = 1, -M < n < M$  widmo Fouriera wynosi:

$$X_M(e^{j\omega}) = \frac{\sin((\omega + 2M + 1)/2)}{\sin(\omega/2)}. \quad (2.27)$$

Charakterystyki amplitudowe widma (2.27) są przedstawione na rys.2.4, dla skończonych ciągów jednostkowych o długościach 3 i 15. Widoczny jest tzw. listek główny dla pulsacji  $\omega=0$ , czyli dla składowej stałej. Wysokość listka jest równa sumie wyrazów ciągu. Szerokość listka zwęża się wraz ze wzrostem długości ciągu. Widmo jest okresowe z okresem  $2\pi$ . Amplituda listków bocznych nie zależy od długości ciągu.



Rys.2.4 Transformata Fouriera ciągu  $x_M[n] = 1, -M < n < M$ .

### 2.8.2 Transformata Fouriera zespolonego ciągu eksponentjalnego

Rozważmy  $x[n]$ , którego transformata Fouriera jest w postaci:

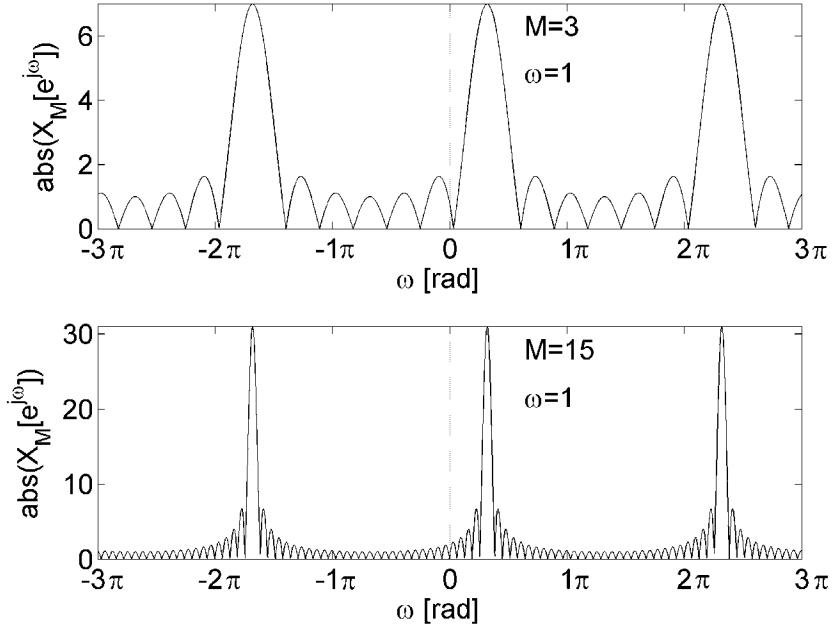
$$X(e^{j\omega}) = \sum_{r=-\infty}^{\infty} 2\pi\delta(\omega - \omega_0 + 2\pi r). \quad (2.28)$$

Odwrotna transformata Fouriera, przy założeniu  $-\pi < \omega_0 \leq \pi$  i  $r = 0$ , ma postać:

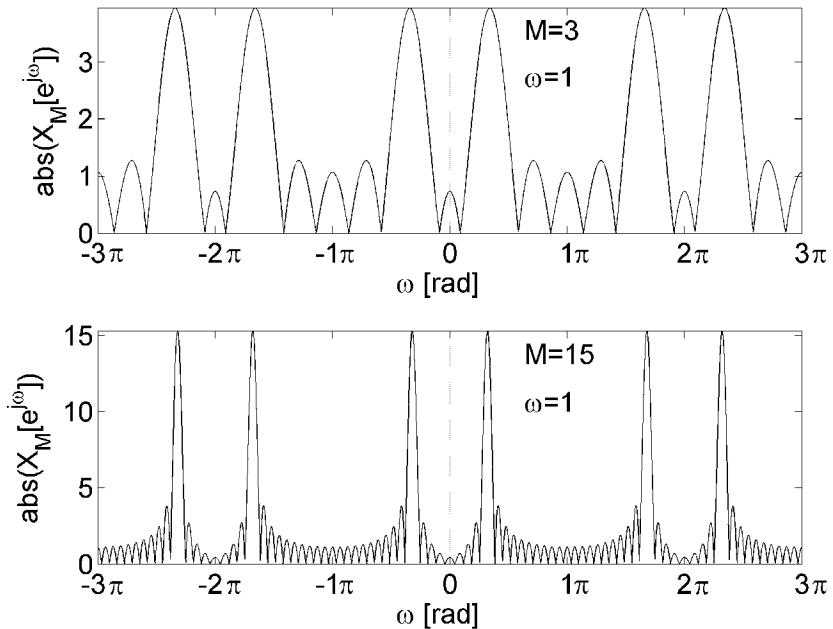
$$x[n] = \frac{1}{2\pi} \int_{-\pi}^{\pi} X(e^{j\omega}) e^{j\omega n} d\omega = \frac{1}{2\pi} \int_{-\pi}^{\pi} 2\pi\delta(\omega - \omega_0) e^{j\omega n} d\omega, \quad (2.29)$$

co z definicji funkcji impulsowej oznacza:

$$x[n] = e^{j\omega_0 n}, \quad -\infty < n < \infty. \quad (2.30)$$



Rys.2.5 Transformata Fouriera ciągu  $x_M[n] = e^{j\omega_0 n}$ ,  $-M < n < M$ .



Rys.2.6 Transformata Fouriera ciągu  $x_M[n] = \cos(\omega_0 n)$ ,  $-M < n < M$ .

Dla  $\omega_0=0$  otrzymujemy z (2.30) ciąg stały. Charakterystyki amplitudowe transformaty Fouriera zespolonego ciągu eksponentialnego dla ciągów o długościach 7 i 31 przedstawione są na rys.2.5. W porównaniu do rys.2.4 listki główne są przesunięte do pulsacji sygnału  $\omega_0$ .

Rys 2.6 przedstawia charakterystyki amplitudowe ciągu sinusoidalnego. Dla sygnałów o wartościach rzeczywistych występuje symetria widma wokół pulsacji  $\omega=0$  (lub  $\omega=\pi$ ), tzn. sygnał sinusoidalny ma dwa listki główne w widmie Fouriera dla częstotliwości  $\omega_0$  i  $-\omega_0$ . Wysokości obu tych listków są równe amplitudzie sygnału pomnożonej przez  $N/2$ , tj. przez połowę długości ciągu.

### 2.8.3 Transformaty Fouriera wybranych ciągów

Poniżej podane są transformaty Fouriera dla wybranych ciągów.

$$x[n] = \quad X(e^{j\omega}) =$$

$$a) \quad \delta[n] \quad 1 \quad (2.31)$$

$$b) \quad \delta[n - n_0] \quad e^{-j\omega n_0} \quad (2.32)$$

$$c) \quad 1, \quad (-\infty < n < \infty) \quad \sum_{k=-\infty}^{\infty} 2\pi\delta(\omega + 2\pi k) \quad (2.33)$$

$$d) \quad \boxed{\frac{\sin(\omega_c n)}{\pi n}} \quad \boxed{\begin{cases} 1, & |\omega| < \omega_c \\ 0, & \omega_c < |\omega| \leq \pi \end{cases}} \quad (2.34)$$

$$e) \quad \boxed{\begin{cases} 1, & 0 \leq n \leq M \\ 0, & 0 > n > M \end{cases}} \quad \boxed{\frac{\sin(\omega \cdot (M+1)/2)}{\sin(\omega/2)} e^{-j\omega M/2}} \quad (2.35)$$

$$f) \quad e^{j\omega_0 n} \quad \sum_{k=-\infty}^{\omega} 2\pi\delta(\omega - \omega_0 + 2\pi k) \quad (2.36)$$

$$g) \quad \boxed{\cos(\omega_0 n + \phi)} \quad \boxed{\sum_{k=-\infty}^{\infty} [\pi e^{j\phi} \delta(\omega - \omega_0 + 2\pi k) + \pi e^{-j\phi} \delta(\omega + \omega_0 + 2\pi k)]} \quad (2.37)$$

$$h) \quad a^n u[n], \quad |a| < 1 \quad \frac{1}{1 - ae^{-j\omega}} \quad (2.38)$$

$$i) \quad u[n] \quad \frac{1}{1 - e^{-j\omega}} + \sum_{k=-\infty}^{\infty} \pi\delta(\omega + 2\pi k) \quad (2.39)$$

Jak widać z (2.31) transformata Fouriera impulsu jednostkowego jest funkcją stałą. Oznacza to, że impuls jednostkowy zawiera wszystkie częstotliwości (całe widmo), dlatego odpowiedź impulsowa układu LTI jednoznacznie określa jego charakterystyki częstotliwościowe.

Szczególnie ważne są transformaty (2.34) i (2.35) określające idealny filtr dolnoprzepustowy oraz widmo okna prostokątnego, o których będzie mowa w dalszych rozdziałach.

Transformata (2.37) stanowi podstawę analizy częstotliwościowej sygnałów.

## 2.8.4 Symetria transformacji Fouriera

Symetria widma transformacji Fouriera jest wykorzystywana w algorytmach obliczeniowych. Poniżej zestawiono podstawowe symetrie widma Fouriera.

Każdy ciąg  $x[n]$  można przedstawić w postaci sumy składowej symetrycznej  $x_e[n]$  i antysymetrycznej  $x_o[n]$ :

$$x[n] = x_e[n] + x_o[n] \quad (2.40)$$

gdzie:

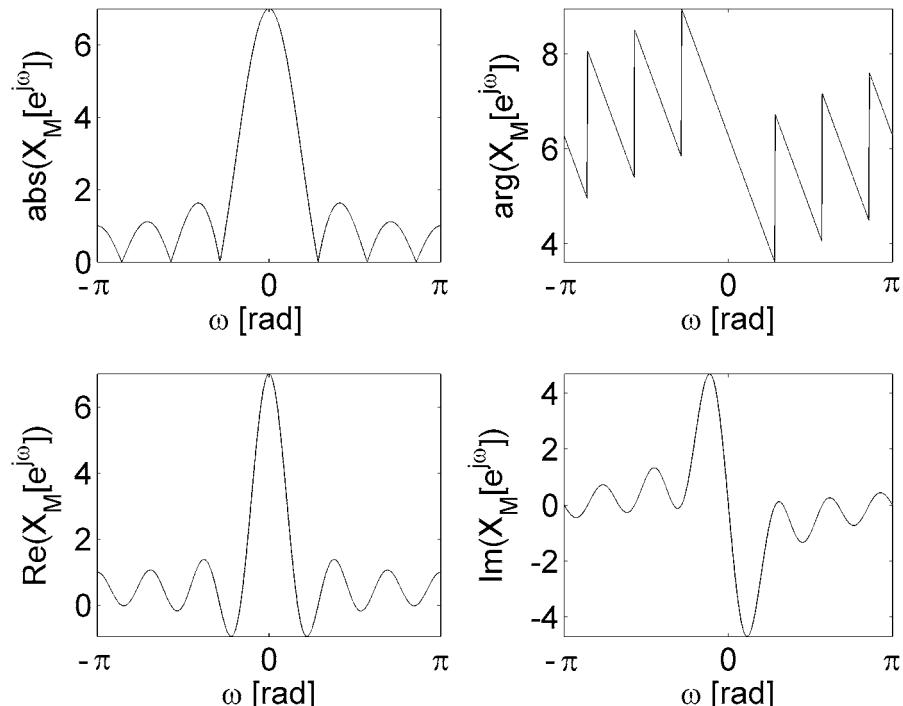
$$x_e[n] = \frac{1}{2}(x[n] + x^*[-n]), \quad x_o[n] = \frac{1}{2}(x[n] - x^*[-n]) = -x_o^*[-n]. \quad (2.41)$$

Transformata Fouriera  $X(e^{j\omega})$  może zostać rozłożona na sumę funkcji sprzężonej symetrycznej i sprzężonej antysymetrycznej:

$$X(e^{j\omega}) = X_e(e^{j\omega}) + X_o(e^{j\omega}), \quad (2.42)$$

gdzie:

$$X_e(e^{j\omega}) = \frac{1}{2}(X(e^{j\omega}) + X^*(e^{-j\omega})) = X_e^*(e^{-j\omega}), \quad X_o(e^{j\omega}) = \frac{1}{2}(X(e^{j\omega}) - X^*(e^{-j\omega})) = -X_o^*(e^{-j\omega}). \quad (2.43)$$



Rys.2.7 Symetrie widma Fouriera dla sygnału o wartościach rzeczywistych.

### Symetrie transformacji Fouriera

- |                          |  |
|--------------------------|--|
| a) $x^*[n]$              | $X^*(e^{-j\omega})$                                |
| b) $x^*[-n]$             | $X^*(e^{j\omega})$                                 |
| c) $\text{Re}\{x[n]\}$   | $X_e(e^{-j\omega})$                                |
| d) $j \text{Im}\{x[n]\}$ | $X_o(e^{-j\omega})$                                |
| e) $x_e[n]$              | $X_R(e^{j\omega}) = \text{Re}\{X(e^{j\omega})\}$   |
| f) $x_o[n]$              | $jX_I(e^{j\omega}) = j\text{Im}\{X(e^{j\omega})\}$ |

Dla ciągu  $x[n]$  o wartościach rzeczywistych (tzn. nie zespolonych) zachodzi:

- |  |  |
|--|--|
| g) $X(e^{j\omega}) = X^*(e^{-j\omega})$                | transformata Fouriera jest sprzężona symetryczna |
| h) $X_R(e^{j\omega}) = X_R(e^{-j\omega})$              | część rzeczywista jest funkcją parzystą          |
| i) $X_I(e^{j\omega}) = -X_I(e^{-j\omega})$             | część urojona jest funkcją nieparzystą           |
| j) $ X(e^{j\omega})  =  X(e^{-j\omega}) $              | moduł jest funkcją parzystą                      |
| k) $\arg\{X(e^{j\omega})\} = -\arg\{X(e^{-j\omega})\}$ | faza jest funkcją nieparzystą                    |

Symetrie widma Fouriera dla sygnału o wartościach rzeczywistych przedstawione są na rys.2.7 dla przyczynowego ciągu stałego  $x[n] = 1, \quad 0 \leq n \leq 6$ .

## 2.8.5 Własności transformacji Fouriera dla ciągów

Poniżej zestawione są wybrane własności transformacji Fouriera dla ciągów, są one wykorzystywane do obliczania widm ciągów przekształconych w dziedzinie czasu przy znajomości widma ciągu oryginalnego.

ciąg	Transformata Fouriera	
a) $ax[n] + by[n]$	$aX(e^{j\omega}) + bY(e^{j\omega})$	liniowość
b) $x[n - n_d]$	$e^{-j\omega n_d} X(e^{j\omega})$	przesunięcie w czasie
c) $e^{j\omega_0 n} x[n]$	$X(e^{j(\omega - \omega_0)})$	przesunięcie w częstotliwości
d) $x[-n]$	$X(e^{-j\omega}), \quad X^*(e^{j\omega}) \text{ dla } \{x[n]\} \in R$	odwrócenie w czasie
e) $nx[n]$	$j \frac{dX(e^{j\omega})}{d\omega}$	różniczkowanie w częstotliwości
f) $x[n] * y[n]$	$X(e^{j\omega})Y(e^{j\omega})$	splot w czasie
g) $x[n] \cdot y[n]$	$\frac{1}{2\pi} \int_{-\pi}^{\pi} X(e^{j\Theta})Y(e^{j(\omega - \Theta)}) d\Theta$	splot w częstotliwości

Jak widać z własności b) charakterystyka amplitudowa nie zależy od przesunięcia sygnału, tak więc np. funkcja sinus i kosinus mają takie same charakterystyki amplitudowe, różnią się natomiast charakterystykami fazowymi.

Zależność c) jest to własność modulacji, która pozwala przesunąć widmo sygnału o wybraną częstotliwość i jest ona często wykorzystywana w telekomunikacji.

Szczególne ważne są własności f) i g), które mówią, że widmo splotu dwóch ciągów jest równoważne iloczynowi widm tych ciągów, oraz że widmo iloczynu dwóch ciągów jest równe splotowi widm tych ciągów.

Transformacja Fouriera zachowuje energię sygnału, tzn. zgodnie z twierdzeniem Parsevala energia sygnału jest równa energii widma tego sygnału:

$$\sum_{n=-\infty}^{\infty} |x[n]|^2 = \frac{1}{2\pi} \int_{-\pi}^{\pi} |X(e^{j\omega})|^2 d\omega, \quad (2.44)$$

gdzie  $|X(e^{j\omega})|^2$  jest tzw. widmową gęstością mocy.

## 2.8.6 Twierdzenie o splocie

Własność splotu f) transformacji Fouriera jest często wykorzystywana ze względu na wydajność obliczeniową, zwłaszcza w przypadku, kiedy filtr nie zmienia się w czasie i wystarczy tylko raz policzyć jego widmo. Zamiast stosowania wzoru definicyjnego na splot (2.10) oblicza się widmo filtra (tylko raz) i widmo sygnału, a następnie wyznacza się odwrotną transformatę Fouriera iloczynu tych widm.

Dowód własności splotu jest oparty na zamianie kolejności sumowania, co pokazano poniżej. Rozważmy dwa ciągi oraz ich widma Fouriera  $x[n] \leftrightarrow H(e^{j\omega})$ ,  $x[n] \leftrightarrow X(e^{j\omega})$ . Licząc z definicji transformacji Fouriera (2.17) widmo wyniku splotu (2.10) otrzymujemy:

$$\sum_{n=-\infty}^{\infty} \left( \sum_{k=-\infty}^{\infty} x[k]h[n-k] \right) e^{-j\omega n} = \sum_{k=-\infty}^{\infty} x[k] \sum_{n=-\infty}^{\infty} h[n-k]e^{-j\omega n}. \quad (2.45)$$

Po prawej stronie (2.45) zamieniono kolejność sumowania. Poniżej przedstawiony jest rozpisany fragment wzoru (2.45). Lewa część (2.45) jest to sumowanie po wierszach, natomiast prawa część (2.45) jest to sumowanie po kolumnach:

$$\begin{aligned} & \vdots \\ & \dots x[k_{-1}]h[n_{-1}+1]e^{j\omega n_{-1}} + x[k_0]h[n_{-1}+0]e^{j\omega n_{-1}} + x[k_1]h[n_{-1}-1]e^{j\omega n_{-1}} \dots \\ & \dots x[k_{-1}]h[n_0+1]e^{j\omega n_0} + x[k_0]h[n_0+0]e^{j\omega n_0} + x[k_1]h[n_0-1]e^{j\omega n_0} \dots \\ & \dots x[k_{-1}]h[n_1+1]e^{j\omega n_1} + x[k_0]h[n_1+0]e^{j\omega n_1} + x[k_1]h[n_1-1]e^{j\omega n_1} \dots \\ & \vdots \end{aligned} \quad (2.46)$$

Zmieniając wskaźnik sumowania w drugiej sumie (2.46) na  $m=n-k$  otrzymujemy:

$$\sum_{k=-\infty}^{\infty} x[k] \sum_{m=-\infty}^{\infty} h[m]e^{-j\omega(k+m)} = \sum_{k=-\infty}^{\infty} x[k]e^{-j\omega k} \sum_{m=-\infty}^{\infty} h[m]e^{-j\omega m} = X(e^{j\omega})H(e^{j\omega}), \quad (2.47)$$

co oznacza, że widmo splotu dwóch ciągów jest równoważne iloczynowi widm tych ciągów.

### 3. Układy analogowe

Układy analogowe opisywane są za pomocą transformacji Laplace'a. Transformacja Laplace'a pozwala przekształcić liniowe równania różniczkowe w łatwiejsze do rozwiązania równania algebraiczne. Transformację Laplace'a (jednostronną) analogowego sygnału  $x(t), t \geq 0$  definiuje się następująco:

$$X(s) = \int_0^{\infty} x(t)e^{-st} dt, \quad s = \sigma + j\Omega, \quad (3.1)$$

gdzie  $\sigma$  jest liczbą rzeczywistą, a  $\Omega$  jest pulsacją sygnału analogowego w [rad/s]. Dla odróżnienia od pulsacji sygnału cyfrowego  $\omega$  w [rad], oznaczanej małą literą, pulsacja sygnału analogowego będzie oznaczana przez  $\Omega$ , tj. literą wielką.

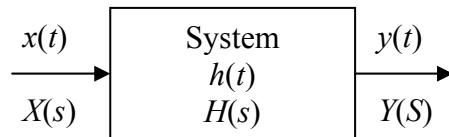
Dla podstawienia:

$$\boxed{s = j\Omega} \quad (3.2)$$

z (3.1) otrzymujemy transformację Fouriera sygnału analogowego:

$$X(j\Omega) = \int_0^{\infty} x(t)e^{-j\Omega t} dt. \quad (3.3)$$

Rozważmy liniowy układ analogowy przedstawiony na rys.3.1. Odpowiedź  $y(t)$  układu na wymuszenie  $x(t)$  określona jest przez splot ciągły sygnału  $x(t)$  z odpowiedzią impulsową  $h(t)$ .



Rys. 3.1 Układ analogowy.

Zakładamy, że układ z rys.3.1 jest opisany jednorodnym równaniem różniczkowym o stałych współczynnikach:

$$a_2 \frac{d^2 y(t)}{dt^2} + a_1 \frac{dy(t)}{dt} + a_0 y(t) = b_1 \frac{dx(t)}{dt} + b_0 x(t). \quad (3.4)$$

Upraszczając notację otrzymujemy:

$$a_2 y''(t) + a_1 y'(t) + a_0 y(t) = b_1 x'(t) + b_0 x(t). \quad (3.5)$$

Licząc transformatę Laplace'a obu stron równania (3.5):

$$\int_0^{\infty} [a_2 y''(t) + a_1 y'(t) + a_0 y(t)] e^{-st} dt = \int_0^{\infty} [b_1 x'(t) + b_0 x(t)] e^{-st} dt, \quad (3.6)$$

oraz korzystając z liniowości przekształcenia i twierdzenia o pochodnej:

$$\int_0^\infty f'(t)e^{-st}dt = s \int_0^\infty f(t)e^{-st}dt - f(0+) = sF(s) - f(0+) , \quad (3.7)$$

otrzymujemy równanie algebraiczne w postaci:

$$a_2s^2Y(s) + a_1sY(s) + a_0Y(s) = b_1sX(s) + b_0X(s) . \quad (3.8)$$

Przekształcając równanie (3.8) otrzymujemy wyrażenie na transmitancję układu analogowego (3.4):

$$H(s) = \frac{Y(s)}{X(s)} = \frac{b_1s + b_0}{a_2s^2 + a_1s + a_0} = k \frac{s - z}{(s - p_1)(s + p_1)} . \quad (3.9)$$

Transmitancja  $H(s)$  układu analogowego jest wyrażona w postaci ilorazu dwóch wielomianów. Po przedstawieniu tych wielomianów w postaci iloczynowej otrzymujemy tzw. zera transmitancji, czyli pierwiastki wielomianu licznika, oraz bieguny transmitancji, czyli pierwiastki wielomianu mianownika.

Przez podstawienie (3.2) do (3.9) otrzymujemy charakterystyki częstotliwościowe układu analogowego. Dla przeanalizowania wpływu położenia zer i biegunów transmitancji  $H(s)$  na charakterystykę amplitudową, rozpatrzmy przykładową transmitancję:

$$H(s) = \frac{Y(s)}{X(s)} = \frac{s - 1}{s^2 + 2s + 2} = \frac{s - 1}{(s + 1 - j)(s + 1 + j)} . \quad (3.10)$$

Transmitancja (3.10) ma jedno zero w punkcie  $1+j0$ , tj. na osi rzeczywistej oraz jeden biegun zespolony, sprzężony w  $-1\pm j$ , pokazane na rys.2.2. Charakterystykę amplitudową wyznaczamy przez podstawienie  $s=j\Omega$ :

$$|H(s = j\Omega)| = \frac{|j\Omega - 1|}{|(j\Omega)^2 + j2\Omega + 2|} = \frac{|j\Omega - 1|}{|-\Omega^2 + j2\Omega + 2|} \quad (3.11)$$

Na podstawie (3.11) można wyznaczyć wzmacnienie układu dla każdej częstotliwości, np. wzmacnienie składowej stałej wynosi:

$$|H(\Omega = 0)| = \frac{|j0 - 1|}{|-0^2 + j2 \cdot 0 + 2|} = \frac{1}{2} . \quad (3.12)$$

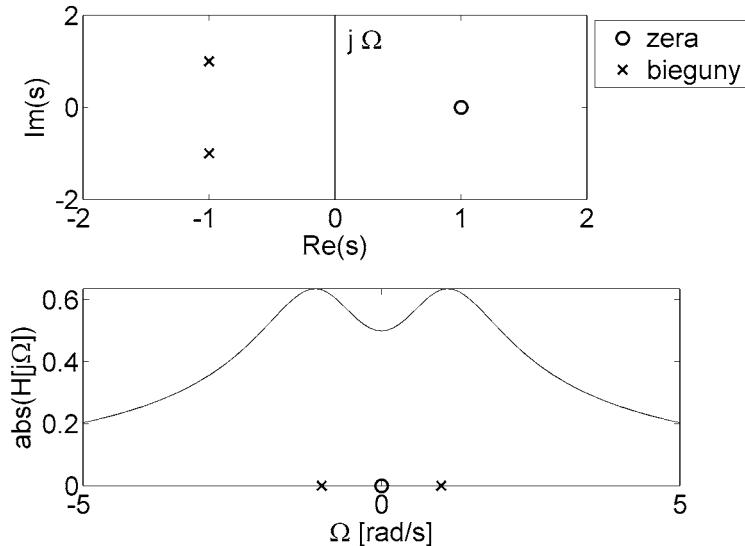
Jak widać z (3.12) wzmacnienie składowej stałej układów analogowych jest równe stosunkowi wyrazów wolnych transmitancji.

Charakterystyka amplitudowa układu (3.10) jest przedstawiona na rys.3.2 dla zakresu częstotliwości od  $-5$  [rad/s] do  $5$  [rad/s]. W odróżnieniu od układów dyskretnych, dla których charakterystyki częstotliwościowe są okresowe z okresem  $2\pi$  i zakres częstotliwości wynosi od  $-\pi$  do  $\pi$ , zakres częstotliwości dla układów analogowych rozciąga się od minus do plus nieskończoności.

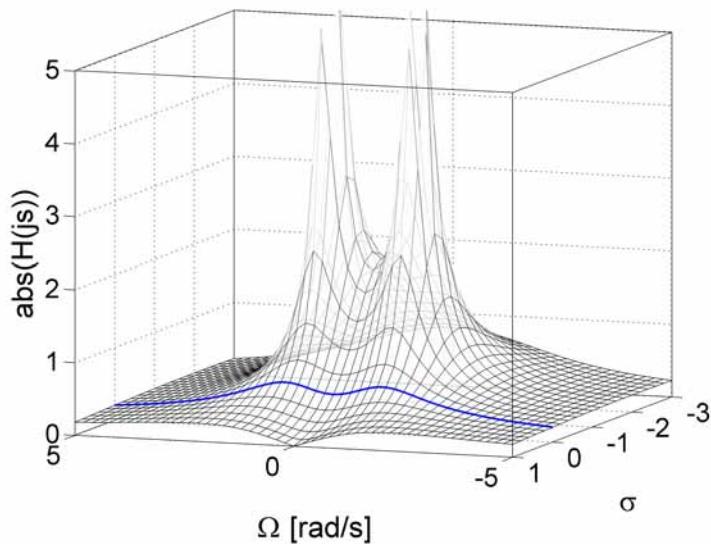
Na rys.3.2 zaznaczono, że oś urojona na płaszczyźnie zespolonej jest osią częstotliwości.

Zera transmitancji powodują lokalne minima w charakterystyce amplitudowej, czyli tłumienie sygnałów o tych częstotliwościach.

Bieguny transmitancji powodują lokalne maksima w charakterystyce amplitudowej, czyli wzmacnienie sygnałów o tych częstotliwościach.



Rys.3.2 Położenie zer i biegunów transmitancji  $H(s)$  na płaszczyźnie zespolonej, oraz charakterystyka amplitudowa układu.



Rys.3.3 Charakterystyka amplitudowa transmitancji (3.10).

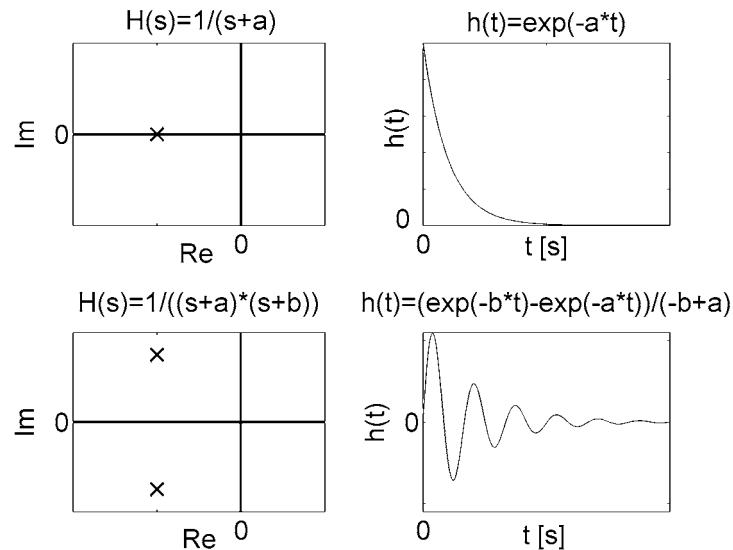
Siła oddziaływania zer i biegunów transmitancji zależy od ich odległości od osi urojonej i rośnie wraz ze zbliżaniem się do tej osi. Rys.3.3 przedstawia charakterystykę amplitudową transmitancji (3.10). Charakterystyka amplitudowa z rys.3.2 jest to przypadek dla  $\sigma=0$  i jest ona zaznaczona linią pogrubioną na rys.3.3. Jak widać z rys.3.3, bieguny transmitancji powodują dzielenie przez zero i wzmacnianie układu rośnie do nieskończoności, natomiast zera transmitancji sprowadzają wzmacnianie układu do zera.

Położenie biegunów transmitancji ma bezpośredni wpływ na stabilność układu. Rys.3.4-3.6 przedstawiają położenie biegunów transmitancji na płaszczyźnie zespolonej oraz odpowiedzi impulsowe układów. W tytułach wykresów podane są transmitancje  $H(s)$  oraz teoretyczne odpowiedzi impulsowe  $h(t)$ .

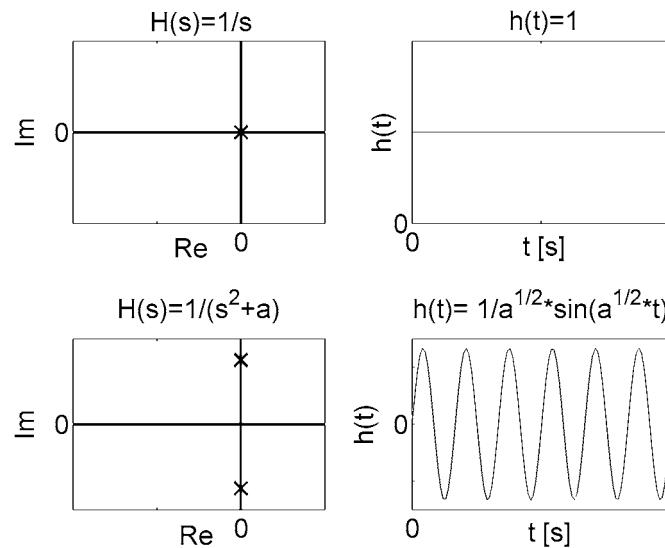
Aby układ analogowy był stabilny, bieguny jego transmitancji muszą leżeć w lewej półpłaszczyźnie płaszczyzny zespolonej, rys.3.4, tj. muszą mieć ujemną część rzeczywistą. Tylko w takim przypadku odpowiedź impulsowa gaśnie do zera.

Dla biegunów położonych na osi urojonej, rys.3.5, układ wzbudza się i zaczyna oscylować, jest to przypadek stabilności warunkowej.

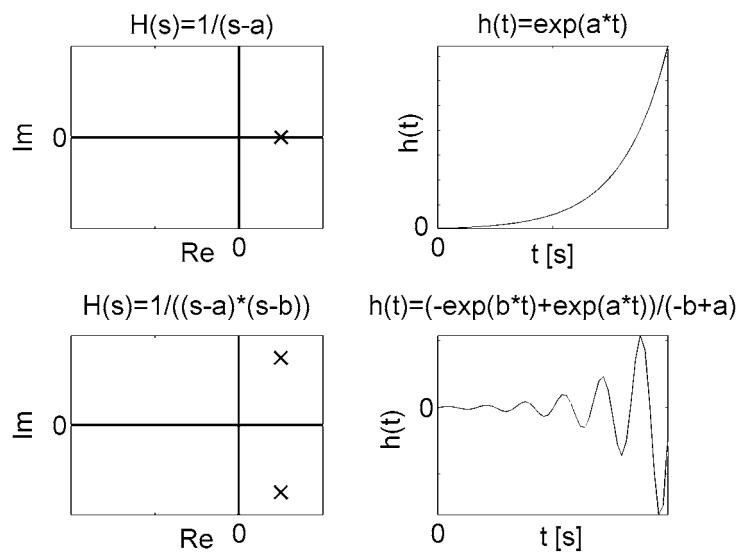
Dla biegunów położonych w prawej półpłaszczyźnie układ jest niestabilny, a jego odpowiedź impulsowa rośnie do nieskończoności.



Rys.3.4 Położenie biegunów transmitancji  $H(s)$  i odpowiedzi impulsowej dla układów stabilnych.



Rys.3.5 Położenie biegunów transmitancji  $H(s)$  i odpowiedzi impulsowej dla układów warunkowo stabilnych.



Rys.3.6 Położenie biegunków transmitancji  $H(s)$  i odpowiedzi impulsowe dla układów niestabilnych.



## 4. Transformacja Z

Transformacja Z jest narzędziem opisu układów dyskretnych i spełnia analogiczną rolę dla układów dyskretnych jak transformacja Laplace'a dla układów analogowych.

Transformata Z istnieje dla szerszej klasy sygnałów niż transformata Fouriera. Notacja transformacji Z jest bardziej dogodna do analitycznego rozwiązywania problemów niż zapis transformacji Fouriera. Transformacja Z sygnału dyskretnego zdefiniowana jest następująco:

$$Z\{x[n]\} = X(z) = \sum_{n=-\infty}^{\infty} x[n]z^{-n}, \quad z \in C \quad (4.1)$$

Ciąg wykładniczy w postaci (4.1) nosi nazwę ciągu Laurenta. Do analizy takich ciągów stosuje się dobrze rozwiniętą teorię funkcji zmiennej zespolonej.

W przypadku ogólnym  $z=re^{j\omega}$  i (4.1) można zapisać jako transformację Fouriera iloczynu sygnału  $x[n]$  i ciągu wykładniczego  $r^{-n}$ :

$$X(z) = X(re^{j\omega}) = \sum_{n=-\infty}^{\infty} x[n](re^{j\omega})^{-n} = \sum_{n=-\infty}^{\infty} (x[n]r^{-n})e^{-j\omega n}. \quad (4.2)$$

Obszar wartości zmiennej  $z$  na płaszczyźnie zespolonej, dla których transformata Z jest ograniczona, nazywany jest obszarem zbieżności. Podobnie jak w przypadku transformacji Fouriera, dla której warunkiem wystarczającym zbieżności jest bezwzględna sumowalność ciągu  $x[n]$  (2.23), warunkiem zbieżności transformacji Z jest bezwzględna sumowalność poniższego ciągu:

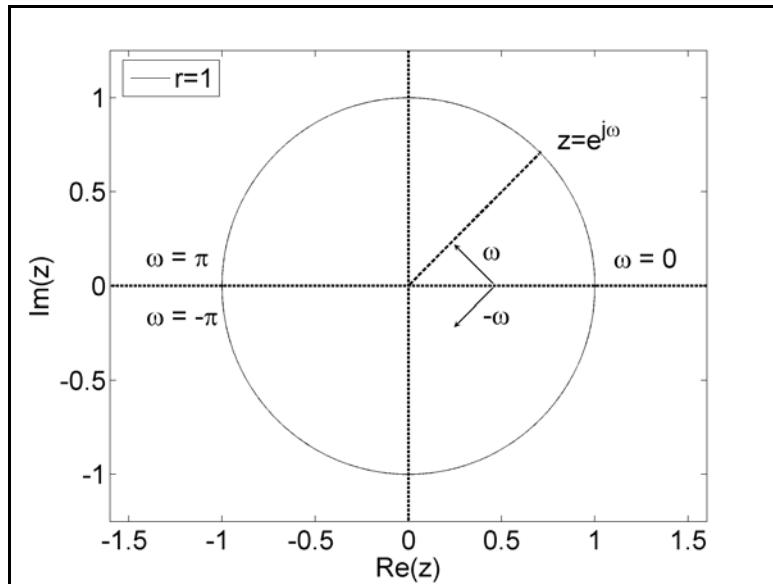
$$\sum_{n=-\infty}^{\infty} |x[n]r^{-n}| < \infty. \quad (4.3)$$

Dzięki czynnikowi  $r^{-n}$  transformata Z może być zbieżna nawet dla ciągów, dla których transformata Fouriera nie jest (np. dla skoku jednostkowego  $u[n]$  dla  $r>1$ , co określa obszar zbieżności jako  $|z|>1$ ). Zbieżność transformaty Z zależy tylko od  $|z|$ , dlatego obszary zbieżności mają postać pierścieni na płaszczyźnie zespolonej.

Jeżeli do wzoru definicyjnego (4.1) podstawimy

$$z = e^{j\omega} \quad (4.4)$$

otrzymamy transformację Fouriera  $X(e^{j\omega})$  (2.17) sygnału dyskretnego (pod warunkiem, że ona istnieje). Transformata Z wyznaczona na okręgu jednostkowym  $|z|=1$  jest równoważna transformacie Fouriera. Rys.4.1 przedstawia interpretację częstotliwościową transformacji Z, dla  $z=e^{j\omega}$ . Dla zmiany  $z$  od 0 do 1 (po okręgu jednostkowym) otrzymujemy zmianę  $\omega$  z przedziału  $<0, \pi>$ , a dla zmiany  $z$  od 0 do -1 (po okręgu jednostkowym) otrzymujemy zmianę  $\omega$  z przedziału  $<0, -\pi>$ . (W sposób naturalny widoczna jest okresowość transformacji Fouriera ciągu dyskretnego). Tak więc częstotliwość w układach dyskretnych zmienia się po okręgu jednostkowym; dla porównania w przypadku układów analogowych częstotliwość zamienia się wzdłuż osi urojonej.



Rys.4.1 Interpretacja częstotliwościowa transformacji  $Z$ .

#### 4.1 Analiza własności układów LTI

Układ LTI (liniowy niezmienny względem przesunięcia) jest całkowicie określony przez odpowiedź impulsową  $h[n]$ , która wiąże odpowiedź układu  $y[n]$  na zadane wymuszenie  $x[n]$  zgodnie z zależnością (2.10). W praktyce stosuje się połączenie układów dyskretnych pokazane na rys.4.2, gdzie  $b[n]$  i  $a[n]$  są wektorami współczynników o długościach odpowiednio  $M+1$  i  $N+1$ . Układ z rys.4.2 spełnia następujące, liniowe równanie różnicowe o stałych współczynnikach:

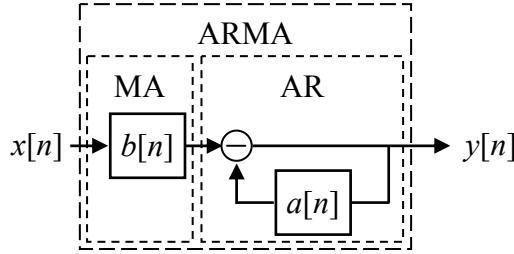
$$\sum_{k=0}^N a[k]y[n-k] = \sum_{m=0}^M b[m]x[n-m]. \quad (4.5)$$

Równanie różnicowe (4.5) jest odpowiednikiem równania różniczkowego dla układów analogowych. Wyłączając pierwszy składnik z lewej sumy w (4.5) otrzymujemy odpowiedź układu na wymuszenie  $x[n]$ :

$$y[n] = \sum_{m=0}^M \frac{b[m]}{a[0]} x[n-m] - \sum_{k=1}^N \frac{a[k]}{a[0]} y[n-k], \quad (4.6)$$

która zależy od aktualnej i poprzednich wartości ciągu wejściowego  $x[n]$  i poprzednich wartości ciągu wyjściowego  $y[n]$  (sumowanie w drugiej sumie (4.6) rozpoczyna się od 1).

Układ dyskretny opisany równaniem (4.5) jest liniowy, niezmienny względem przesunięcia i przyczynowy, jeżeli w stanie początkowym znajduje się w stanie spoczynku (tzn., jeżeli  $x[n]=0$  to  $y[n]=0$  dla  $n < n_0$ ).



Rys.4.2 Struktury układów dyskretnych.

Licząc z definicji (4.1) transformatę  $Z$  obu stron równania różnicowego (4.5) oraz korzystając z twierdzenia o przesunięciu

$$x[n-n_0] \xrightarrow{Z} z^{-n_0} X(z) \quad (4.7)$$

otrzymujemy:

$$\sum_{k=0}^N a[k]z^{-k} Y(z) = \sum_{m=0}^M b[m]z^{-m} X(z), \quad (4.8)$$

lub równoważnie

$$\left( \sum_{k=0}^N a[k]z^{-k} \right) Y(z) = \left( \sum_{m=0}^M b[m]z^{-m} \right) X(z). \quad (4.9)$$

Na podstawie (4.9) można zapisać transmitację  $Z$  układu dyskretnego (dla uproszczenia notacji przyjęto konwencję  $a[k]=a_k$ ,  $b[m]=b_m$ ) w postaci:

$$H(z) = \frac{Y(z)}{X(z)} = \frac{\sum_{m=0}^M b_m z^{-m}}{\sum_{k=0}^N a_k z^{-k}} = \frac{b_0 + b_1 z^{-1} + \dots + b_M z^{-M}}{a_0 + a_1 z^{-1} + \dots + a_N z^{-N}}. \quad (4.10)$$

Transmitanca  $Z$  (4.10) jest ilorazem wielomianów zmiennej  $z^{-1}$ .

Jeżeli  $M=0$  ( $b$  jest skalarem) i  $N>0$  to  $H(z)$  jest transmitancją filtra o nieskończonej odpowiedzi impulsowej IIR (*Infinite Impulse Response*). Jest to filtr rekursywny, ponieważ zawiera sprzężenie zwrotne. Filtr ten nazywany jest filtrem autoregresji (*autoregressive*) AR.

Jeżeli  $N=0$  ( $a$  jest skalarem) i  $M>0$  to  $H(z)$  jest transmitancją filtra o skończonej odpowiedzi impulsowej FIR (*Finite Impulse Response*). Jest to filtr nierekursywny, ponieważ sygnał na jego wyjściu zależy tylko od sygnału na jego wejściu. Filtr ten nazywany jest filtrem ruchomej średniej (*moving-average*) MA.

Jeżeli  $M>0$  i  $N>0$  to  $H(z)$  jest transmitancją filtra IIR. Jest to filtr rekursywny. Jest on nazywany filtrem autoregresji ruchomej średniej (*autoregressive moving-average*) ARMA.

Wielomiany w (4.10) można przedstawić w postaci iloczynowej w celu wyodrębnienia zer i biegunów transmitancji  $H(z)$ :

$$H(z) = w \frac{\prod_{m=1}^M (1 - c_m z^{-1})}{\prod_{k=1}^N (1 - d_k z^{-1})} = w \frac{(1 - c_1 z^{-1}) \dots (1 - c_M z^{-1})}{(1 - d_1 z^{-1}) \dots (1 - d_N z^{-1})}, \quad (4.11)$$

gdzie  $w$  jest wzmocnieniem. Każdy z czynników  $(1-c_m z^{-1})$  wprowadza zero dla  $z=c_m$  i biegun dla  $z=0$ , analogicznie każdy z czynników  $(1-d_k z^{-1})$  wprowadza zero dla  $z=0$  i biegun dla  $z=d_k$ . Wyciągając przed nawiasy  $z^{-1}$  w (4.11) możemy otrzymać wielomian zmiennej  $z$  (a nie  $z^{-1}$ ):

$$H(z) = w \frac{\prod_{m=1}^M (1-c_m z^{-1})}{\prod_{k=1}^N (1-d_k z^{-1})} = w \frac{z^{-M} (z-c_1) \dots (z-c_M)}{z^{-N} (z-d_1) \dots (z-d_N)}. \quad (4.12)$$

Podobnie jak w przypadku układów analogowych również dla układów dyskretnych położenie zer i biegunów transmitancji określa własności układu. Dla układów dyskretnych istotne jest położenie zer i biegunów względem okręgu jednostkowego, co wyjaśniono na rys.4.1.

Jako przykład przeanalizujmy układ dyskretny zdefiniowany przez zera  $c = [1.1e^{j\pi/2} \ 1.1e^{-j\pi/2}]$ , bieguny  $d = [0.7e^{j\pi/8} \ 0.7e^{-j\pi/8}]$  i wzmocnienie  $w=1$  transmitancji  $H(z)$ . Zarówno zera jak i bieguny  $H(z)$ , jeżeli są zespolone to występują w parach sprzężonych, aby współczynniki  $b, a$  wielomianów transmitancji w postaci (4.10) były rzeczywiste (a nie zespolone). Transmitancja tego układu wynosi:

$$H(z) = \frac{(1-1.1e^{j\pi/2}z^{-1})(1-1.1e^{-j\pi/2}z^{-1})}{(1-0.7e^{j\pi/8}z^{-1})(1-0.7e^{-j\pi/8}z^{-1})} = \frac{1+1.21z^{-2}}{1-1.2934z^{-1}+0.49z^{-2}}, \quad (4.13a)$$

lub według (4.12)

$$H(z) = \frac{(z-1.1e^{j\pi/2})(z-1.1e^{-j\pi/2})}{(z-0.7e^{j\pi/8})(z-0.7e^{-j\pi/8})} = \frac{z^2 + 1.21}{z^2 - 1.2934z + 0.49}. \quad (4.13a)$$

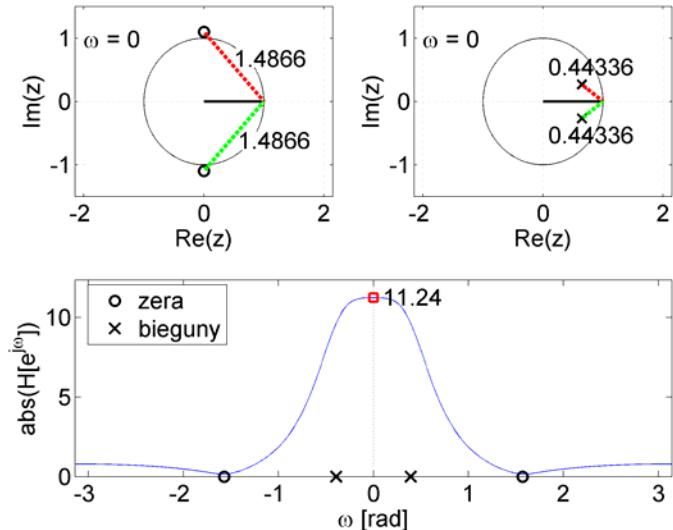
Charakterystykę amplitudową otrzymujemy przez podstawienie (4.4) do (4.13):

$$|H(e^{j\omega})| = \frac{|(e^{j\omega} - c)| \cdot |(e^{j\omega} - c^*)|}{|(e^{j\omega} - d)| \cdot |(e^{j\omega} - d^*)|}. \quad (4.14)$$

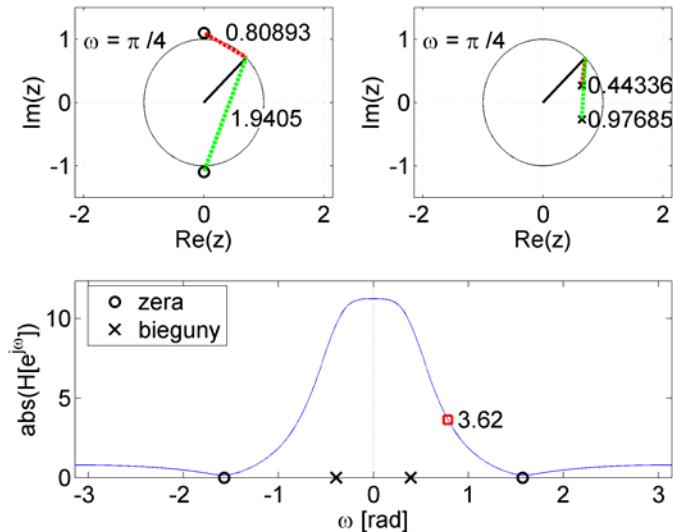
Jak widać z (4.14), w liczniku występuje iloczyn odległości zera  $c$  i jego sprzężenia zespolonego  $c^*$  od okręgu jednostkowego  $z=e^{j\omega}$  w punkcie  $\omega$ . Analogiczna sytuacja występuje w mianowniku. Rysunki 4.3-4.4 przedstawiają interpretację graficzną zależności (4.14) dla dwóch różnych częstotliwości. Rys.4.3 przedstawia odległości zer (lewy róg) i biegunów (prawy róg) od pulsacji  $\omega=0$ , czyli od punktu  $z=e^{j0}=1$ . W rezultacie otrzymujemy wzmocnienie składowej stałej:

$$|H(e^{j\omega})| = \frac{1.4866 \cdot 1.4866}{0.44336 \cdot 0.44336} = 11.24. \quad (4.15)$$

Jak widać z (4.15), dla  $\omega=0$  wpływ biegunów na charakterystykę amplitudową pokazaną na rys.4.3 jest mocniejszy niż wpływ zer. Gdyby bieguny leżały bliżej okręgu, ich wpływ byłby jeszcze większy, ponieważ w mianowniku (4.15) byłaby mniejsza wartość i w rezultacie wzmocnienie byłoby większe. Analogiczne wykresy dla  $\omega=\pi/4$  przedstawia rys.4.4. Wzmocnienie charakterystyki amplitudowej dla tej częstotliwości, wyznaczone jak poprzednio, wynosi 3.62.



Rys.4.3 Wpływ położenia zer i biegunów transmitancji  $H(z)$  na charakterystykę amplitudową.



Rys.4.4 Wpływ położenia zer i biegunów transmitancji  $H(z)$  na charakterystykę amplitudową.

Wzmocnienie składowej stałej można również wyznaczyć z postaci (4.10) transmitancji, przez podstawienie  $z=e^{j0}=1$ . Dla (4.13) otrzymujemy:

$$|H(z=1)| = \left| \frac{1+1.21}{1-1.2934+0.49} \right| = 11.24 . \quad (4.16)$$

Jeżeli przyjmiemy wzmocnienie  $w=1/11.24$ , otrzymamy wzmocnienie składowej stałej równe 1, kształt charakterystyki częstotliwościowej pozostanie bez zmian.

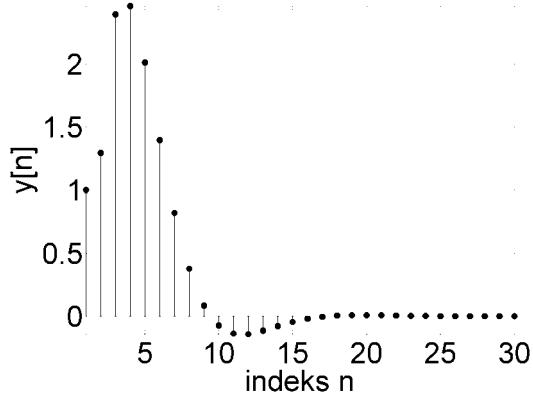
W celu określenia odpowiedzi układu na dowolny sygnał wejściowy należy wykonać odwrotną transformację  $Z$ , w następujący sposób:

$$H(z) = \frac{Y(z)}{X(z)} = \frac{1+1.21z^{-2}}{1-1.2934z^{-1}+0.49z^{-2}} \rightarrow Y(z) = X(z) + 1.21X(z)z^{-2} + 1.2934Y(z)z^{-1} - 0.49Y(z)z^{-2} \quad (4.17)$$

i ostatecznie

$$y[n] = x[n] + 1.21x[n-2] + 1.2934y[n-1] - 0.49y[n-2]. \quad (4.18)$$

Równanie (4.18) jest to równanie różnicowe w postaci (4.6). Dla ciągu wejściowego będącego impulsem jednostkowym na podstawie (4.18) można wyznaczyć odpowiedź impulsową pokazaną na rys.4.5.



Rys.4.5 Odpowiedź impulsowa układu (4.13).

Rys.4.6-4.8 przedstawiają położenie biegunk transmitancji  $H(z)$  oraz odpowiadające im odpowiedzi impulsowe układów. Układ dyskretny jest stabilny tylko w przypadku, gdy biegunki jego transmitancji leżą wewnątrz okręgu jednostkowego na płaszczyźnie zespolonej, tzn. tylko wtedy odpowiedź impulsowa gaśnie do zera, co pokazano na rys.4.6.

Dla biegunków położonych na okręgu jednostkowym odpowiedź impulsowa układu oscyluje jak na rys.4.7. Jest to stabilności warunkowa.

Biegunki układu dyskretnego położone poza okręgiem jednostkowym, tak jak na rys.4.8 powodują, że układ jest niestabilny, tzn. jego odpowiedź impulsowa rośnie do nieskończoności.

Rozważmy układ o transmitancji:

$$H(z) = \frac{Y(z)}{X(z)} = \frac{1}{z-a}. \quad (4.19)$$

Układ (4.19) opisuje następujące równanie różnicowe pierwszego rzędu:

$$y[n+1] = x[n] + ay[n]. \quad (4.20)$$

Aby układ był stabilny, odpowiedź na ograniczone wymuszenie musi być również ograniczona. Przyjmijmy, że  $x[n]$  jest impulsem jednostkowym (1.5), wówczas na podstawie równania różnicowego mamy (dla warunku początkowego  $y[n]=0, n \leq 0$ ):

$$\begin{aligned} n=0, \quad y[1] &= x[0] + ay[0] = 1, \\ n=1, \quad y[2] &= x[1] + ay[1] = a, \\ n=2, \quad y[3] &= x[2] + ay[2] = a^2, \\ n=3, \quad y[4] &= x[3] + ay[3] = a^3, \end{aligned} \quad (4.21)$$

ogólnie:

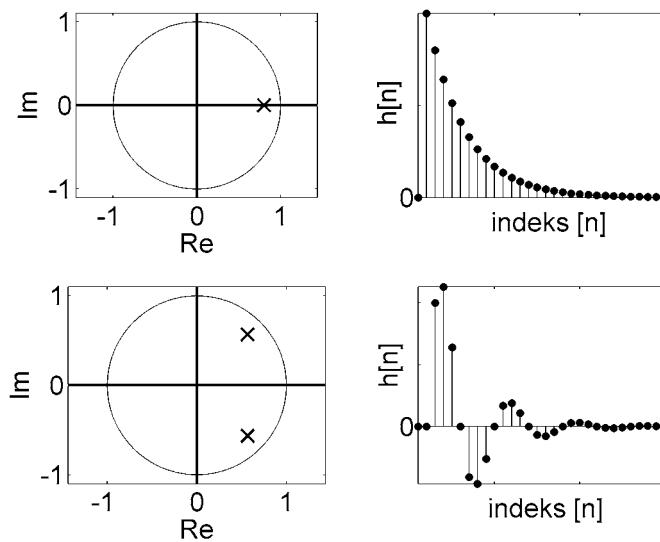
$$y[n] = a^n, \quad n \geq 0. \quad (4.22)$$

Żądamy, aby  $\sum_{n=-\infty}^{\infty} |y[n]| < \infty$ . Stosując wzór na sumę wyrazów ciągu geometrycznego:

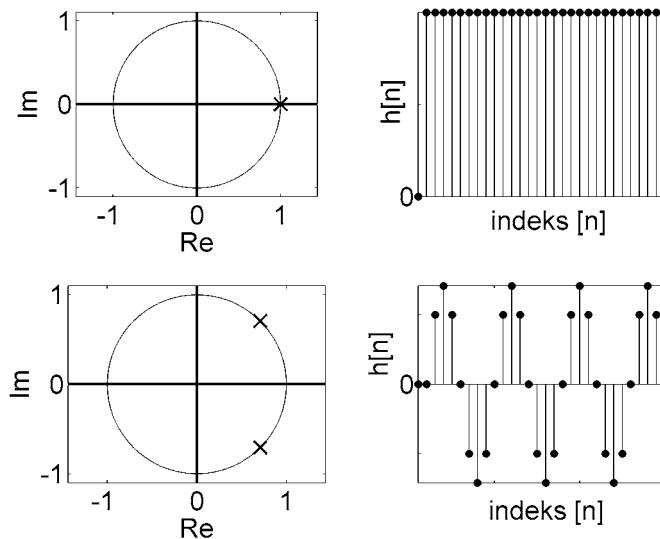
$$\sum_{n=p}^{N-1} a^n = \frac{a^p - a^N}{1-a}$$

$$\sum_{n=0}^{\infty} |a|^n = \frac{1}{1-|a|} < \infty \quad \text{dla } |a| < 1. \quad (4.23)$$

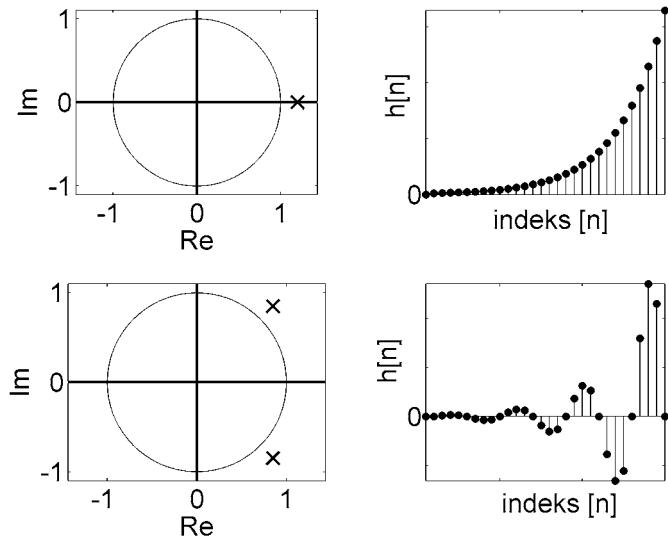
Ciąg  $y[n]$  jest ograniczony dla  $|a| < 1$ , co oznacza, że biegun transmitancji (4.19) musi leżeć wewnątrz koła jednostkowego.



Rys.4.6 Położenie biegunów transmitancji  $H(z)$  i odpowiedzi impulsowe dla układów stabilnych.



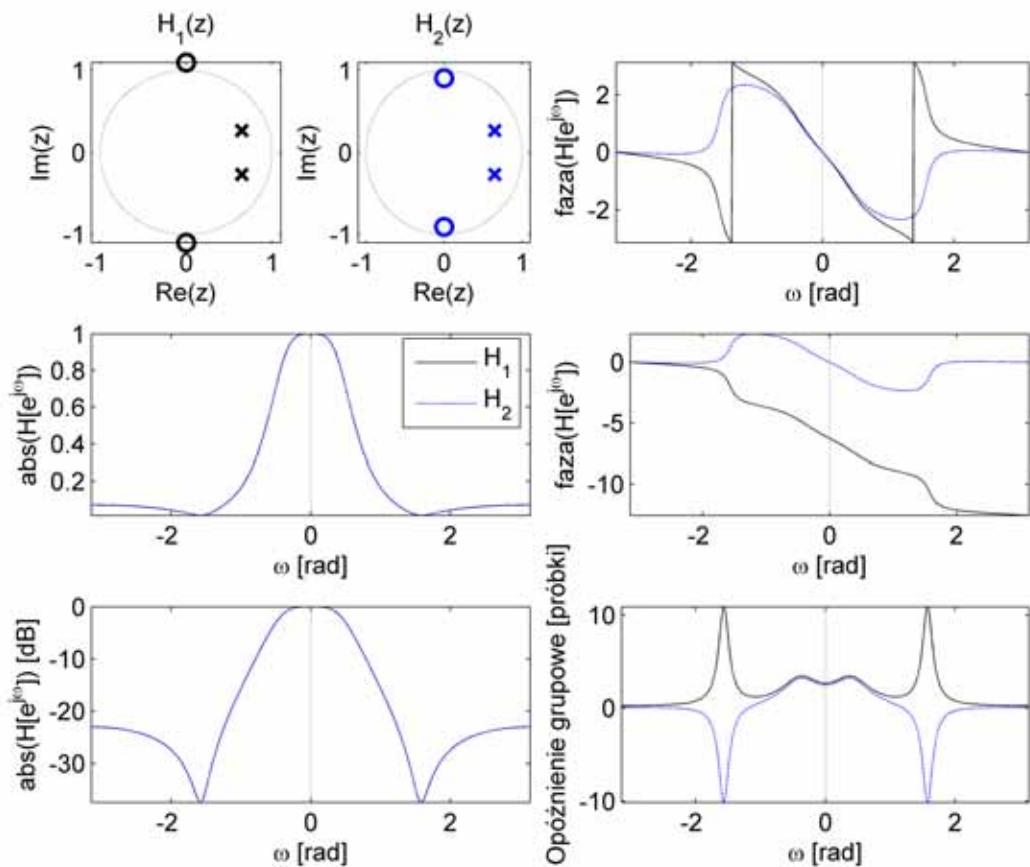
Rys.4.7 Położenie biegunów transmitancji  $H(z)$  i odpowiedzi impulsowe dla układów warunkowo stabilnych.



Rys.4.8 Położenie biegunków transmitancji  $H(z)$  i odpowiedzi impulsowe dla układów niestabilnych.

## 4.2 Charakterystyki częstotliwościowe układu dyskretnego

Rys.4.9 przedstawia charakterystyki częstotliwościowe dwóch układów dyskretnych. Układy te mają te same biegunki, natomiast zera spełniają zależność  $z_2=1/z_1$ , czyli zera  $H_2(z)$  są odwrotnościami zer  $H_1(z)$ .



Rys.4.9 Charakterystyki częstotliwościowe układów dyskretnych.

Charakterystyki amplitudowe obu układów są takie same. Rys.4.9 przedstawia wzmacnienie w skali liniowej oraz w skali decybelowej policzone jako:

$$|H(e^{j\omega})| [\text{dB}] = 20 \log_{10} |H(e^{j\omega})|. \quad (4.24)$$

Stosowanie skali decybelowej umożliwia obserwację sygnałów o bardzo dużym zakresie dynamiki (zmienności).

Charakterystyki fazowe przedstawiają wykresy policzone bezpośrednio za pomocą funkcji arctg, oraz wykresy ciągłe, które uwzględniają skoki fazy spowodowane sposobem liczenia funkcji arctg, która typowo jest wyznaczana w ćwiartce I i IV.

Dla ciągłej charakterystyki fazowej opóźnienie grupowe definiuje się jako pochodną charakterystyki fazowej:

$$\tau = -\frac{d}{d\omega} \{\arg[H(e^{j\omega})]\}. \quad (4.25)$$

Opóźnienie grupowe mówi o ile próbka zostanie opóźniona dana częstotliwość na wyjściu układu. W celu zachowania kształtu przetwarzanego sygnału opóźnienie grupowe powinno być stałe, czyli charakterystyka fazowa powinna być liniowa.

### 4.3 Układ wszechprzepustowy

Charakterystyka amplitudowa układu wszechprzepustowego (*allpass*) jest płaska. Rozważmy układ o transmitancji:

$$H_{ap}(z) = \frac{(z-c)(z-c^*)}{\left(z - \frac{1}{c^*}\right)\left(z - \frac{1}{c}\right)}, \quad c = Ce^{j\varphi}, \quad |c| > 1. \quad (4.26)$$

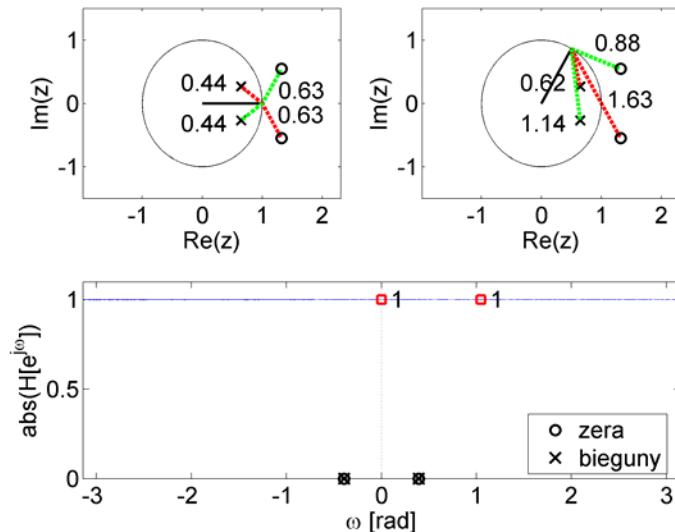
Transformata Fouriera transmitancji (4.26) jest następująca:

$$\begin{aligned} H_{ap}(e^{j\omega}) &= \frac{(e^{j\omega}-c)(e^{j\omega}-c^*)}{\left(e^{j\omega} - \frac{1}{c^*}\right)\left(e^{j\omega} - \frac{1}{c}\right)} = \frac{(e^{j\omega}-c)(e^{j\omega}-c^*)}{\frac{1}{c^*}(e^{j\omega}c^*-1)\frac{1}{c}(e^{j\omega}c-1)} = cc^* \frac{(e^{j\omega}-c)(e^{j\omega}-c^*)}{(e^{j\omega}c^*-1)(e^{j\omega}c-1)} = \\ &= cc^* \frac{(e^{j\omega}-c)(e^{j\omega}-c^*)}{e^{j\omega}(c^* - \frac{1}{e^{j\omega}})e^{j\omega}(c - \frac{1}{e^{j\omega}})} = cc^* \frac{(e^{j\omega}-c)(e^{j\omega}-c^*)}{e^{j2\omega}(c^* - e^{-j\omega})(c - e^{-j\omega})} = e^{-j2\omega} cc^* \frac{(e^{j\omega}-c)(e^{j\omega}-c^*)}{(e^{-j\omega}-c^*)(e^{-j\omega}-c)}, \end{aligned} \quad (4.27)$$

zatem charakterystyka amplitudowa jest stała w całym zakresie częstotliwości:

$$|H_{ap}(e^{j\omega})| = C^2. \quad (4.28)$$

Rys.4.10 przedstawia położenie zer i biegunów przykładowego filtra wszechprzepustowego oraz jego charakterystykę amplitudową. Współczynnik wzmacnienia tego filtra wynosi  $w=0.5$ . Jak widać, dla obu przedstawionych częstotliwości charakterystyka amplitudowa ma wartość 1, tj.  $0.5 \cdot (0.63 \cdot 0.63) / (0.44 \cdot 0.44) = 1$  i  $0.5 \cdot (0.88 \cdot 1.63) / (1.14 \cdot 0.62) = 1$ . Filtry wszechprzepustowe stosowane są do korekcji charakterystyk fazowych.



Rys.4.10 Charakterystyka amplitudowa układu wszechprzepustowego.

#### 4.4 Układ odwrotny

Dla danego układu  $H(z)$  układ odwrotny (*inverse system*)  $H_i(z)$  spełnia zależność:

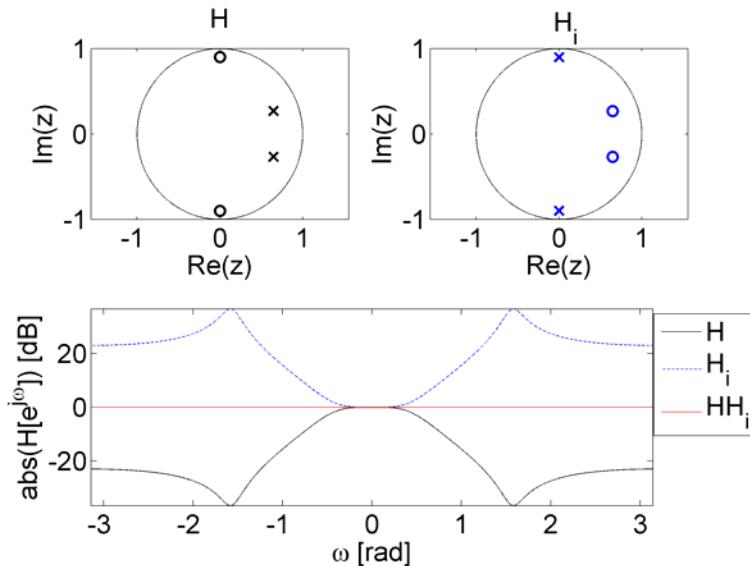
$$H(z)H_i(z)=1, \quad (4.29)$$

co pociąga

$$H_i(z)=\frac{1}{H(z)}, \quad (4.30)$$

a w dziedzinie czasu

$$h[n]*h_i[n]=\delta[n]. \quad (4.31)$$



Rys.4.11 Położenie zer i biegunków układow o transmitancji  $H(z)$  i układow odwrotnego oraz ich charakterystyki amplitudowe.

Zgodnie z (4.30) biegunki filtra odwrotnego są zerami filtra  $H(z)$ , a zera filtra odwrotnego są biegunkami filtra  $H(z)$ , co pokazano na rys.4.11. Żeby filtr odwrotny  $H_i(z)$  był stabilny, zera filtra  $H(z)$  muszą leżeć wewnątrz okręgu jednostkowego na zespolonej płaszczyźnie  $Z$ .

Układ dyskretny LTI, którego zera i bieguny transmitancji leżą wewnątrz okręgu jednostkowego, jest nazywany układem minimalnofazowym (*minimum-phase*).

Układy odwrotne są stosowane do korekcji charakterystyk amplitudowych.

#### **4.5 Dekompozycja minimalnofazowo-wszechprzepustowa**

Dla filtrów nieminimalnofazowych, czyli takich, które mają zera na zewnątrz okręgu jednostkowego, nie można wyznaczyć filtra odwrotnego w sposób bezpośredni, tj. z (4.30). Można natomiast przedstawić dowolny filtr cyfrowy  $H(z)$  w postaci kaskady filtra minimalnofazowego  $H_{min}(z)$  i filtra wszechprzepustowego  $H_{ap}(z)$  (4.26):

$$H(z) = H_{min}(z)H_{ap}(z) . \quad (4.32)$$

Rozważmy filtr o transmitancji:

$$H(z) = \frac{(z-c)(z-c^*)}{(z-d)(z-d^*)}, \quad |c| > 1 . \quad (4.33)$$

Przekształcając (4.33) otrzymujemy żądaną rezultat tj. (4.32):

$$H(z) = \frac{(z-c)(z-c^*)}{(z-d)(z-d^*)} = \frac{(z-c)(z-c^*)}{(z-d)(z-d^*)} \frac{\left(z - \frac{1}{c^*}\right) \left(z - \frac{1}{c}\right)}{\left(z - \frac{1}{c^*}\right) \left(z - \frac{1}{c}\right)} = \frac{\left(z - \frac{1}{c^*}\right) \left(z - \frac{1}{c}\right)}{(z-d)(z-d^*)} \frac{(z-c)(z-c^*)}{\left(z - \frac{1}{c^*}\right) \left(z - \frac{1}{c}\right)} = H_{min}(z)H_{ap}(z) . \quad (4.34)$$

Charakterystyka amplitudowa  $H(z)$  jest taka sama jak  $H_{min}(z)$  (z dokładnością do wzmacnienia):

$$\left|H(e^{j\omega})\right| = \left|H_{min}(e^{j\omega})\right| \left|H_{ap}(e^{j\omega})\right| = w \left|H_{min}(e^{j\omega})\right|. \quad (4.35)$$



## 5. Próbkowanie sygnałów

Dyskretną reprezentację sygnału ciągłego uzyskuje się przez okresowe próbkowanie:

$$x[n] = x_c(nT), \quad -\infty < n < \infty, \quad (5.1)$$

gdzie  $T$  jest okresem próbkowania w sekundach. Odwrotność okresu próbkowania jest to częstotliwość próbkowania w hercach:

$$F_s = \frac{1}{T}. \quad (5.2)$$

Równanie (5.1) opisuje idealny przetwornik analogowo-cyfrowy A/C.

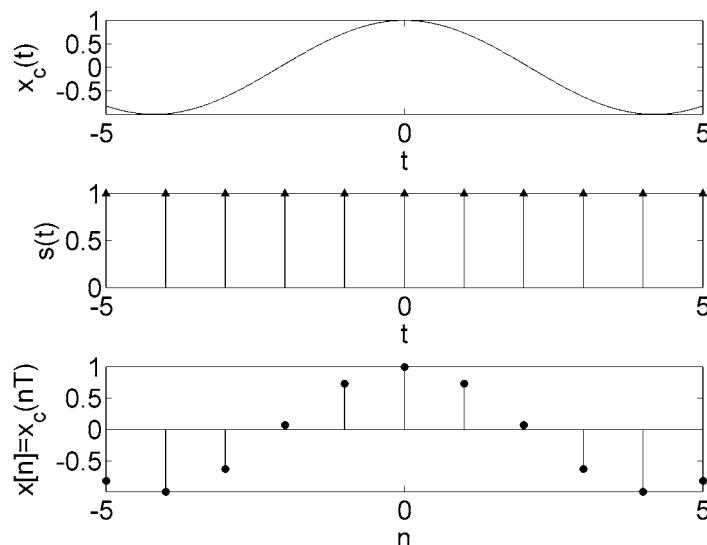
W celu matematycznego opisu operacji próbkowania definiuje się, pokazany na rys.5.1, ciągły sygnał zbudowany z równoodległych impulsów Diraca:

$$s(t) = \sum_{n=-\infty}^{\infty} \delta(t-nT). \quad (5.3)$$

Ciągły sygnał spróbkowany jest przedstawiany jako iloczyn sygnału ciągłego z sygnałem (5.3):

$$x_s(t) = x_c(t)s(t) = x_c(t) \sum_{n=-\infty}^{\infty} \delta(t-nT) = \sum_{n=-\infty}^{\infty} x_c(nT)\delta(t-nT). \quad (5.4)$$

Następnie ciągły sygnał spróbkowany  $x_s(t)$  jest zamieniany na ciąg dyskretny  $x[n]$ . Operacje te przedstawione są na rys.5.1.



Rys.5.1 Próbkowanie sygnału ciągłego.

W dziedzinie częstotliwości widmo iloczynu  $x_c(t)x_s(t)$  jest splotem widm tych sygnałów, tzn. dla sygnałów i ich transformat Fouriera  $x_c(t) \xrightarrow{F} X_c(j\Omega)$  i  $s(t) \xrightarrow{F} S(j\Omega)$  zachodzi:

$$X_s(j\Omega) = \frac{1}{2\pi} X_c(j\Omega) * S(j\Omega). \quad (5.5)$$

Widmo sygnału (5.3) złożonego z impulsów Diraca również składa się z impulsów Diraca:

$$S(j\Omega) = \frac{2\pi}{T} \sum_{k=-\infty}^{\infty} \delta(\Omega - k\Omega_s), \quad \Omega_s = \frac{2\pi}{T} = 2\pi F_s. \quad (5.6)$$

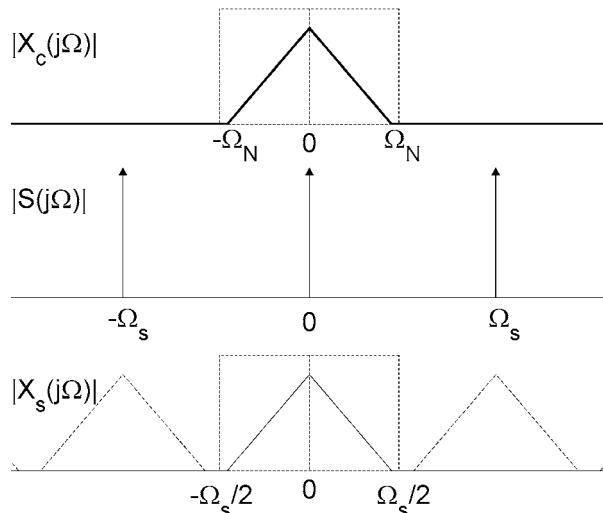
Uwzględniając (5.5) i (5.6) otrzymujemy wyrażenie na widmo sygnału spróbkowanego:

$$X_s(j\Omega) = \frac{1}{T} \sum_{k=-\infty}^{\infty} X_c(j(\Omega - k\Omega_s)). \quad (5.7)$$

Widmo sygnału spróbkowanego składa się z okresowo powtarzanych kopii widma sygnału ciągłego  $X_c(j\Omega)$  (próbkowanego). Kopie te są poprzesuwane o całkowite krotności częstotliwości próbkowania  $\Omega_s$ . Aby sąsiednie kopie widma nie zachodziły na siebie, musi być spełniony warunek:

$$\Omega_s - \Omega_N > \Omega_N \text{ lub } \Omega_s > 2\Omega_N \quad (5.8)$$

gdzie  $\Omega_N$  jest największą częstotliwością w widmie sygnału ciągłego  $X_c(j\Omega)$ . Warunek (5.8) mówi więc, że częstotliwość próbkowania musi być dwa razy większa niż największa częstotliwość występująca w widmie sygnału próbkiowanego.

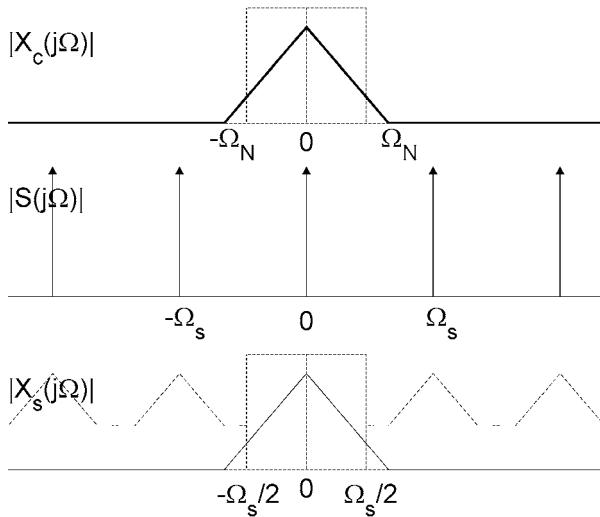


Rys.5.2 Widmo sygnału spróbkowanego dla  $\Omega_s > 2\Omega_N$ .

Rys.5.2 przedstawia widmo sygnału ciągłego, widmo ciągu impulsów Diraca oraz widmo sygnału spróbkowanego. Na rysunku rozpatrywany jest sygnał ciągły, dolnopasmowy zaznaczony trójkątem. Jeżeli operacja próbkowania jest wystarczająco gęsta (szymbka), tzn. impulsy Diraca w ciągu próbkującym (5.3) są odpowiednio blisko siebie, to oddalenie impulsów Diraca w widmie ciągu próbkującego jest wystarczająco duże, żeby sąsiednie obrazy widma sygnału ciągłego nie zachodziły na siebie. Taki przypadek ilustruje rys.5.2. Widmo sygnału spróbkowanego  $X_s(j\Omega)$  jest splotem widm  $X_c(j\Omega)$  i  $S(j\Omega)$ , jednak odległości impulsów w widmie  $S(j\Omega)$  są na tyle duże, że sąsiednie obrazy widma sygnału ciągłego nie zakłócą się. Jak widać z rys.5.2, jeden okres widma sygnału spróbkowanego jest identyczny z widmem sygnału ciągłego.

Rys.5.3 przedstawia przypadek zbyt wolnego próbkowania sygnału ciągłego. Widmo sygnału spróbkowanego  $X_s(j\Omega)$  jest zakłócone przez sąsiednie obrazy tego widma. Zjawisko

to określone jest jako aliasing, czyli nakładanie się widm. W tym przypadku jeden okres widma sygnału spróbkowanego nie jest taki sam jak widmo sygnału analogowego.



Rys.5.3 Widmo sygnału spróbkowanego dla  $\Omega_s < 2\Omega_N$ , widoczne jest nakładanie się widm, tzw. aliasing.

Warunek (5.8) jest to twierdzenie Nyquista o próbkowaniu, które brzmi następująco. Niech  $x_c(t)$  będzie sygnałem o ograniczonym paśmie częstotliwości  $X_c(j\Omega) = 0$ ,  $|\Omega| \geq \Omega_N$ . Wówczas  $x_c(t)$  jest jednoznacznie określony przez próbki  $x[n] = x_c(nT)$ ,  $n = 0, \pm 1, \pm 2, \dots$  jeżeli  $\Omega_s = (2\pi/T) \geq 2\Omega_N$ .

Twierdzenie o próbkowaniu jest sformułowane w sposób bardziej ogólny niż warunek (5.8) i dotyczy sygnałów o ograniczonym paśmie, niekoniecznie dolnoprzepustowych. W praktyce przed próbkowaniem sygnału ciągłego jego pasmo ogranicza się za pomocą filtrów pasmowych, najczęściej dolnoprzepustowych, są to tzw. filtry antialiasingowe.

Z definicji transformacji Fouriera widmo sygnału ciągłego, spróbkowanego wynosi  $X_s(j\Omega) = \sum_{n=-\infty}^{\infty} x_c(nT) e^{-j\Omega T n}$ , a dyskretnego  $X(e^{j\omega}) = \sum_{n=-\infty}^{\infty} x[n] e^{-j\omega n}$ . Ponieważ  $x[n] = x_c(nT)$  można zapisać:  $X_s(j\Omega) = X(e^{j\omega}) \Big|_{\omega=\Omega T} = X(e^{j\Omega T})$ , co po uwzględnieniu (5.7) prowadzi do:

$$X(e^{j\Omega T}) = \frac{1}{T} \sum_{k=-\infty}^{\infty} X_c(j(\Omega - k\Omega_s)), \quad (5.9)$$

lub podstawiając  $\omega = \Omega T$ ,  $\Omega_s = \frac{2\pi}{T}$  do:

$$X(e^{j\omega}) = \frac{1}{T} \sum_{k=-\infty}^{\infty} X_c\left(j\left(\frac{\omega}{T} - \frac{2\pi k}{T}\right)\right). \quad (5.10)$$

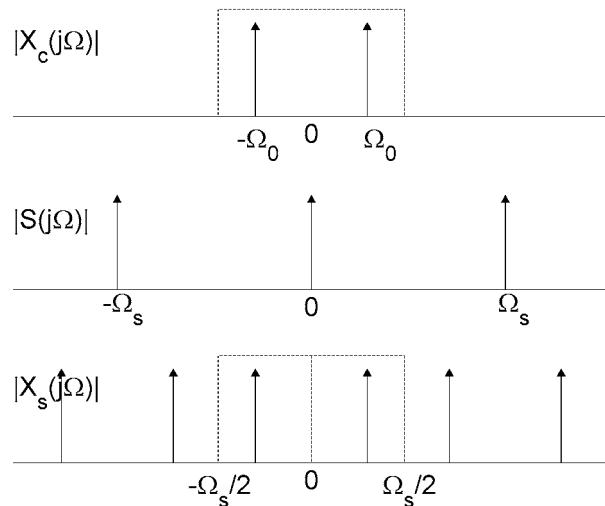
Zgodnie z (5.10) widmo sygnału dyskretnego  $X(e^{j\omega})$  jest przeskalowaną w częstotliwości przez współczynnik  $\omega = \Omega T$  wersję widma sygnału spróbkowanego  $X_s(j\Omega)$ . Częstotliwość próbkowania  $\Omega = \Omega_s$  odpowiada częstotliwość znormalizowana  $\omega = 2\pi$ .

Jako kolejną ilustrację twierdzenia o próbkowaniu rozważmy sygnał sinusoidalny. Widmo tego sygnału składa się z dwóch impulsów Diraca:

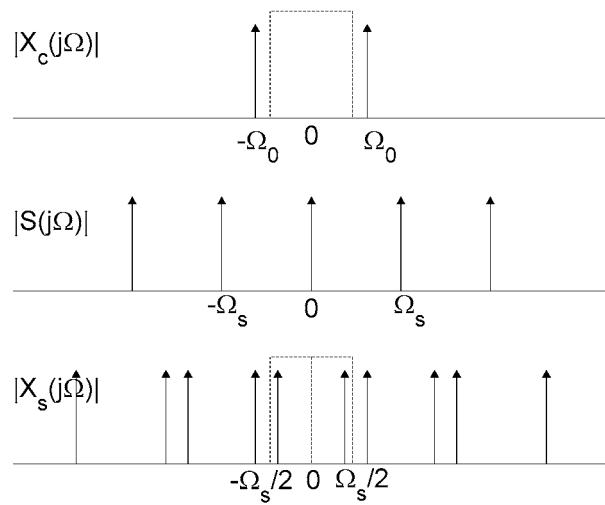
$$x_c(t) = \cos(\Omega_0 t) \quad \longleftrightarrow \quad X_c(j\Omega) = \pi(\delta(\Omega_0) + \delta(-\Omega_0)). \quad (5.11)$$

Rys.5.4 przedstawia przykład próbkowania sygnału sinusoidalnego o częstotliwości  $\Omega_0$  z częstotliwością  $\Omega_s$  większą niż podwojona częstotliwość  $\Omega_0$ , tj.  $\Omega_s > 2\Omega_0$ . Jeden okres widma sygnału spróbkowanego jest taki sam jak widmo sygnału ciągłego.

Rys.5.5 przedstawia przykład próbkowania sygnału sinusoidalnego o częstotliwości  $\Omega_0$  z częstotliwością  $\Omega_s$  mniejszą niż  $2\Omega_0$ . Widmo sygnału sinusoidalnego jest prążkowe i sąsiednie obrazy nie zakłócają się. Jeden okres widma sygnału spróbkowanego jest taki sam jak widma sygnału ciągłego, chociaż prążki położone są w częstotliwości  $\Omega_s - \Omega_0$ , a nie w  $\Omega_0$ . Ogólnie, dla sygnałów o ograniczonym paśmie częstotliwości wystarczy, aby częstotliwość próbkowania była dwukrotnie większa od szerokości pasma sygnału, a nie największej częstotliwości zawartej w tym sygnale, co jest obowiązujące tylko dla sygnałów dolnopasmowych.



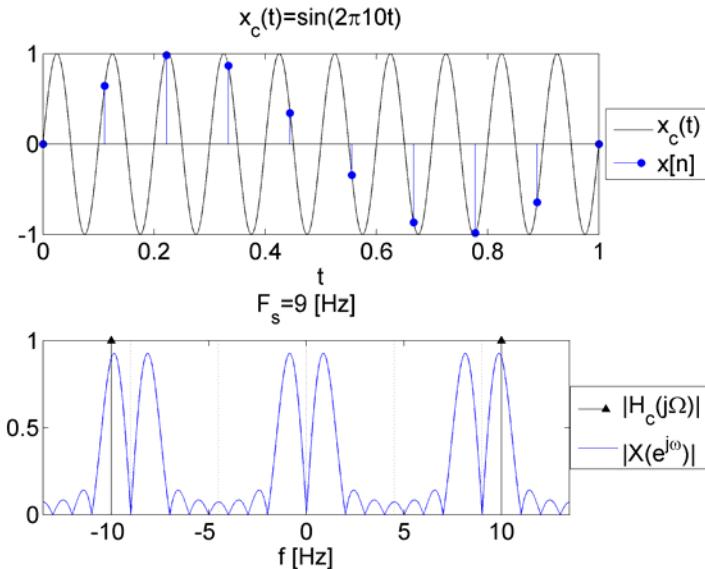
Rys.5.4 Widmo sygnału sinusoidalnego o częstotliwości  $\Omega_0$  spróbkowanego z częstotliwością  $\Omega_s > 2\Omega_0$ .



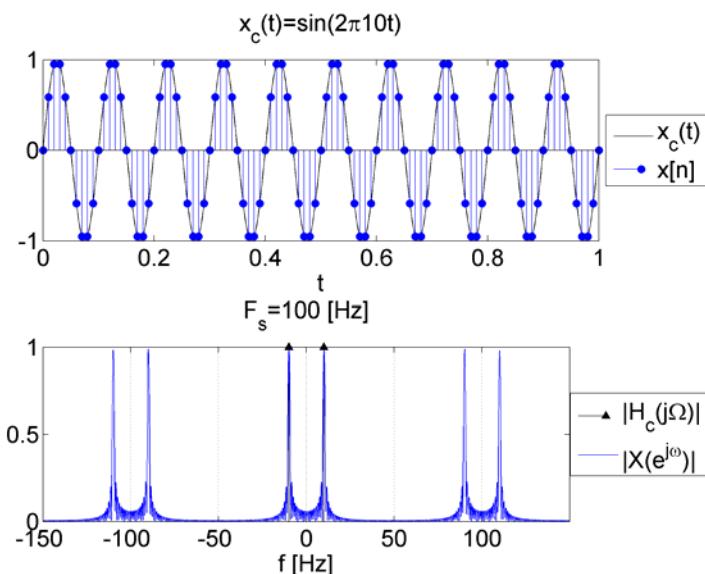
Rys.5.5 Widmo sygnału sinusoidalnego o częstotliwości  $\Omega_0$  spróbkowanego z częstotliwością  $\Omega_s < 2\Omega_0$ .  
Sygnał spróbkowany zawiera prążek o częstotliwości  $\Omega_s - \Omega_0$ .

Na rys.5.6 przedstawiono przebiegi czasowe sygnału sinusoidalnego o częstotliwości 10 Hz spróbkowanego z częstotliwością 9 Hz. W widmie sygnału spróbkowanego widoczna jest częstotliwość  $10-9=1$  Hz. Przedstawione na rys.5.6 widmo sygnału zostało wyznaczone

za pomocą transformacji Fouriera dla skończonego ciągu danych. Widmo to nie jest impulsem jednostkowym, składa się natomiast z tzw. listka głównego i listków bocznych. Własności widm sygnałów dyskretnych są omówione w rozdziale 9. Rys.5.6 pokazuje, że zbyt wolne pobieranie próbek z sygnału sinusoidalnego (okresowego) może skutkować obserwacją sygnału sinusoidalnego o niższej częstotliwości. Jest to efekt stroboskopowy możliwy do zaobserwowania np. na filmach w przypadku, gdy ruch łopat helikoptera lub kół samochodu daje złudzenie, że obracają się one w zwolnionym tempie, a nawet w stronę przeciwną do ruchu.



Rys.5.6 Przykład próbkowania sygnału ciągłego o częstotliwości 10 Hz z częstotliwością 9 Hz.

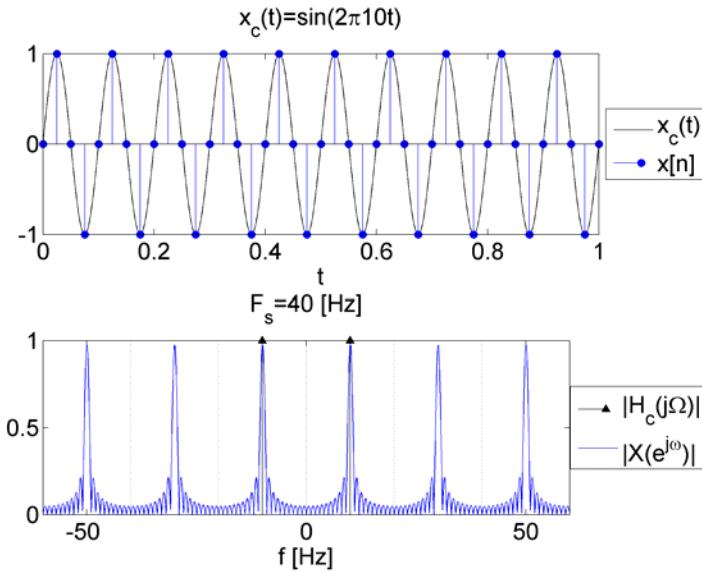


Rys.5.7 Przykład próbkowania sygnału ciągłego o częstotliwości 10 Hz z częstotliwością 100 Hz, tj. 10 próbek na okres.

Rys.5.7 przedstawia przykład próbkowania sygnału z szybkością 10 próbek na okres, co jest dobrym rozwiązaniem praktycznym w większości zastosowań, chociaż stosowane są również systemy, w których sygnał jest próbkowany znacznie częściej.

Rys. 5.8 przedstawia próbkowanie z szybkością 4 próbki na okres. Tak spróbkowany sygnał zawiera tylko wartości -1,0,1 i jest w praktyce stosowany do modulacji innych

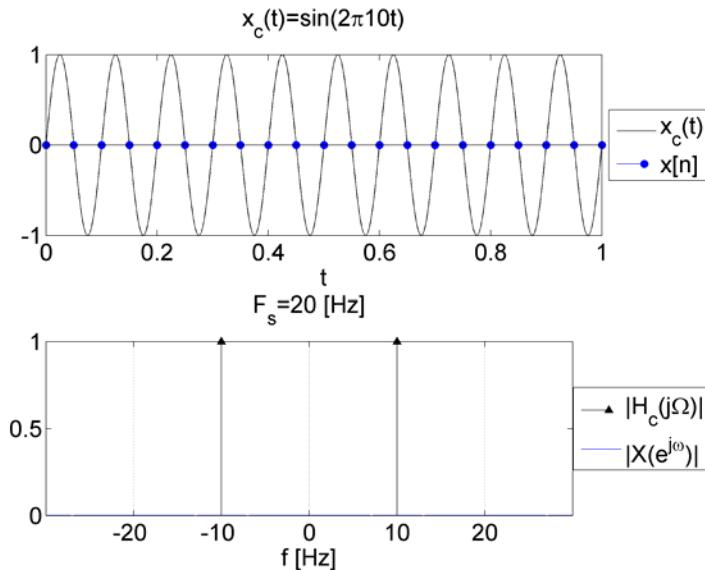
sygnałów. Pomnożenie dowolnego sygnału przez ten ciąg przesuwa widmo tego sygnału o  $F_s/4$  Hz, czyli  $\pi/2$  rad.



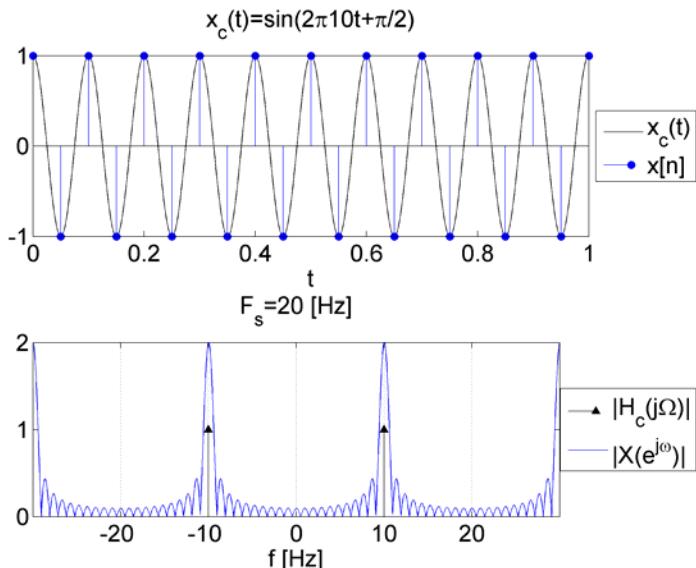
Rys.5.8 Przykład próbkowania sygnału ciągłego o częstotliwości 10 Hz z częstotliwością 40 Hz, tj. 4 próbki na okres.

Rys.5.9 i 5.10 przedstawiają próbkowanie z szybkością 2 próbek na okres. W przypadku funkcji sinus oznacza to wybieranie samych zer z sygnału, natomiast dla funkcji kosinus wybierane są ekstrema. Ciąg z rys.5.10 stosowany jest do modulacji sygnałów i przesuwa widmo sygnału o  $F_s/2$  Hz, czyli  $\pi$  rad.

Na podstawie dwóch próbek pobieranych z okresu nie można odtworzyć sygnału ciągłego, nawet w przypadku funkcji kosinus z rys.5.10 reprezentacja jest niejednoznaczna, ponieważ taki sam ciąg można otrzymać próbując przesuniętą w fazie funkcję kosinus o większej amplitudzie.



Rys.5.9 Przykład próbkowania sygnału ciągłego o częstotliwości 10 Hz z częstotliwością 20 Hz, tj. 2 próbki na okres.



Rys.5.10 Przykład próbkowania sygnału ciągłego o częstotliwości 10 Hz z częstotliwością 20 Hz, tj. 2 próbki na okres.

### 5.1 Rekonstrukcja sygnału ciągłego z reprezentacji dyskretnej

Do rekonstrukcji sygnału ciągłego z jego reprezentacji dyskretnej stosuje się analogowy filtr dolnoprzepustowy, który wycina z okresowego widma sygnału dyskretnego fragment identyczny z widmem sygnału analogowego zaznaczony prostokątem na rys.5.2. Jeśli na wejście analogowego filtra rekonstrukcji  $h_r(t)$  podamy sygnał  $x_s(t) = \sum_{n=-\infty}^{\infty} x[n]\delta(t-nT)$  to na jego wyjściu otrzymamy sygnał zrekonstruowany:

$$x_r(t) = \sum_{n=-\infty}^{\infty} x[n]h_r(t-nT). \quad (5.12)$$

Częstotliwość graniczną filtra  $h_r(t)$  dobiera się jako połowę częstotliwości próbkowania. Odpowiedź impulsowa filtra  $h_r(t)$  (odwrotna transformacja Fouriera zadanej funkcji częstotliwości) ma postać:

$$h_r(t) = \frac{\sin(\pi t/T)}{\pi t/T}, \quad (5.13)$$

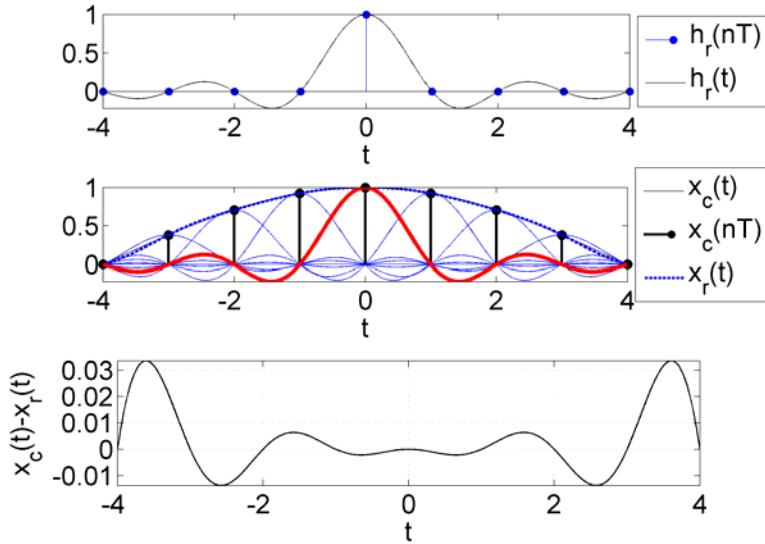
po podstawieniu (5.13) do (5.12):

$$x_r(t) = \sum_{n=-\infty}^{\infty} x[n] \frac{\sin(\pi(t-nT)/T)}{\pi(t-nT)/T}. \quad (5.14)$$

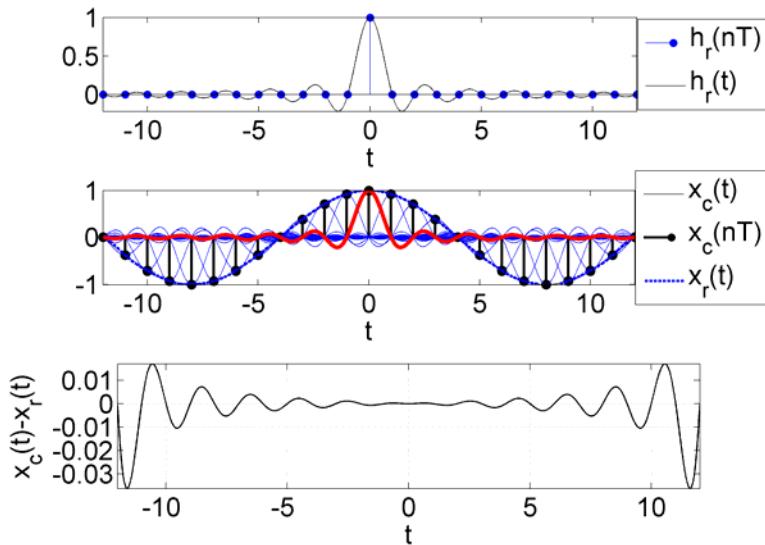
Z reguły d'Hospitala  $h_r(t=0)=1$ , dodatkowo zachodzi  $h_r(nT)=0$ ,  $n=\pm 1, \pm 2, \dots$  co oznacza, że sygnał zrekonstruowany ma te same wartości, co sygnał dyskretny w chwilach próbkowania. Filtr rekonstrukcji interpoluje wartości dla czasów pomiędzy próbami sygnału dyskretnego za pomocą funkcji  $\sin(x)/x$ .

Ilustracja rekonstrukcji sygnału ciągłego z reprezentacji dyskretnej za pomocą filtra dolnoprzepustowego przedstawiona jest na rys.5.11 i 5.12 dla dwóch różnych częstotliwości próbkowania. Rysunki przedstawiają odpowiedzi impulsowe filtra analogowego. W chwilach próbkowania odpowiedź impulsowa filtra rekonstrukcji wynosi zero, za wyjątkiem próbki bieżącej. Wartość sygnału rekonstruowanego pomiędzy próbami jest sumą poprzesuwanych

i przeskalowanych odpowiedzi impulsowych. Pokazany błąd rekonstrukcji jest spowodowany skończoną długością sygnału dyskretnego.



Rys.5.11 Rekonstrukcja sygnału ciągłego z reprezentacji dyskretnej za pomocą filtra dolnoprzepustowego.



Rys.3.12 Rekonstrukcja sygnału ciągłego z reprezentacji dyskretnej za pomocą filtra dolnoprzepustowego.

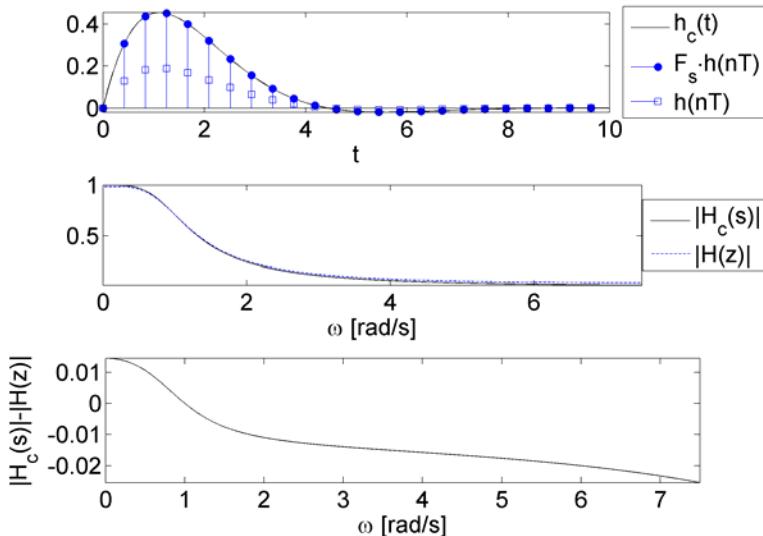
## 5.2 Metoda niezmiennej odpowiedzi impulsowej

Założymy, że chcemy zrealizować funkcję analogowego systemu LTI  $H_c(j\Omega)$  w postaci dyskretnej  $H(e^{j\omega})$ , tzn., aby zachodziło  $H(e^{j\omega}) = H_c(j\Omega/T)$ ,  $|\omega| < \pi$  przy takim doborze  $T$ , że  $H_c(j\Omega) = 0$ ,  $|\Omega| < \pi/T$ , tzn. układ analogowy jest dolnoprzepustowy, wtedy

$$h[n] = T h_c(nT) . \quad (5.15)$$

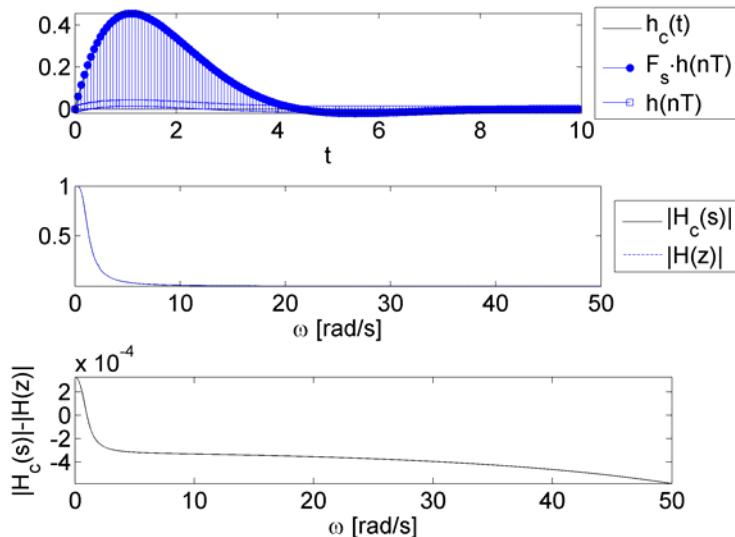
Jako przykład rozważmy transmitancję analogową w postaci:

$$H_c(s) = \frac{1}{s^2 + \sqrt{2}s + 1} . \quad (5.16)$$



Rys.5.13 Realizacja transmitancji analogowej (5.16) w postaci dyskretnej:

$$\Omega_s = 15 \left[ \frac{\text{rad}}{\text{s}} \right] \rightarrow T = \frac{2\pi}{\Omega_s} = 0.4189[\text{s}], H(z) = \frac{0.12858 z}{z^2 - 1.4225 z + 0.55301}.$$



Rys.5.14 Realizacja transmitancji analogowej (5.16) w postaci dyskretnej:

$$\Omega_s = 100 \left[ \frac{\text{rad}}{\text{s}} \right] \rightarrow T = \frac{2\pi}{\Omega_s} = 0.0628[\text{s}], H(z) = \frac{0.003775 z}{z^2 - 1.9112 z + 0.91498}.$$

Odpowiedź impulsowa transmitancji (5.16) wynosi:

$$h_c(t) = \sqrt{2} e^{-\frac{\sqrt{2}}{2}t} \sin\left(\frac{\sqrt{2}}{2}t\right). \quad (5.17)$$

Podstawiając (5.17) do (5.15) otrzymujemy odpowiedź impulsową układu dyskretnego:

$$h[n] = Th_c(nT) = T \sqrt{2} e^{-\frac{\sqrt{2}}{2}nT} \sin\left(\frac{\sqrt{2}}{2}nT\right), \quad (5.18)$$

z której, na podstawie tablic transformaty  $Z$ , wyznaczamy transmitancję układu dyskretnego:

$$H(z) = \frac{z \left[ T\sqrt{2} \sin\left(\frac{T\sqrt{2}}{2}\right) e^{-\frac{T\sqrt{2}}{2}} \right]}{z^2 - z \left[ 2 \cos\left(\frac{T\sqrt{2}}{2}\right) e^{-\frac{T\sqrt{2}}{2}} \right] + e^{-T\sqrt{2}}}. \quad (5.19)$$

Rys.5.13 i rys.5.14 przedstawiają odpowiedzi impulsowe oraz porównanie charakterystyk amplitudowych dla układu analogowego i dyskretnego dla częstotliwości próbkowania równych odpowiednio  $\Omega_s=15$  rad/s i  $\Omega_s=100$  rad/s.

### 5.3 Zmiana częstotliwości próbkowania

Usuniecie próbek z sygnału dyskretnego wg zależności:

$$x_d[n] = x[nM] = x_c[nMT] \quad (5.20)$$

powoduje obniżenie częstotliwości próbkowania  $F_s$  o całkowity czynnik  $M$ ; jest to tzw. decymacja sygnału lub *downsampling*. Powszechnie używane określenie decymacja nie oddaje dobrze zależności (5.20), ponieważ  $M$  nie musi równać się 10.

Ze względu na aliasing  $F_s$  może być obniżona  $M$  razy tylko wtedy, jeżeli najwyższa częstotliwość w widmie sygnału jest mniejsza od  $F_s/M$ . Przed decymacją stosuje się więc antyalisingowy, cyfrowy filtr dolnoprzepustowy.

Widma sygnałów  $x[n]$  i  $x_d[n]$  są następujące:

$$X(e^{j\omega}) = \frac{1}{T} \sum_{k=-\infty}^{\infty} X_c\left(j\left(\frac{\omega}{T} - \frac{2\pi k}{T}\right)\right) \text{ i } X_d(e^{j\omega}) = \frac{1}{MT} \sum_{r=-\infty}^{\infty} X_c\left(j\left(\frac{\omega}{MT} - \frac{2\pi r}{MT}\right)\right), \quad (5.21)$$

podstawiając  $r = i + kM$ ,  $-\infty < k < \infty$ ,  $0 \leq i \leq M-1$  otrzymujemy:

$$X_d(e^{j\omega}) = \frac{1}{MT} \sum_{r=-\infty}^{\infty} X_c\left(j\left(\frac{\omega}{MT} - \frac{2\pi r}{MT}\right)\right) = \frac{1}{MT} \sum_{i+kM=-\infty}^{\infty} X_c\left(j\left(\frac{\omega}{MT} - \frac{2\pi(i+kM)}{MT}\right)\right) = \frac{1}{M} \sum_{i=0}^{M-1} \left[ \frac{1}{T} \sum_{k=-\infty}^{\infty} X_c\left(j\left(\frac{\omega}{MT} - \frac{2\pi k}{T} - \frac{2\pi i}{MT}\right)\right) \right],$$

oznaczmy wyrażenie w nawiasach kwadratowych przez:

$$X\left(e^{j(\omega-2\pi i)/M}\right) = \frac{1}{T} \sum_{k=-\infty}^{\infty} X_c\left(j\left(\frac{\omega-2\pi i}{MT} - \frac{2\pi k}{T}\right)\right). \quad (5.22)$$

Ostatecznie zależność pomiędzy widmami sygnałów  $x[n]$  i  $x_d[n]$  jest następująca:

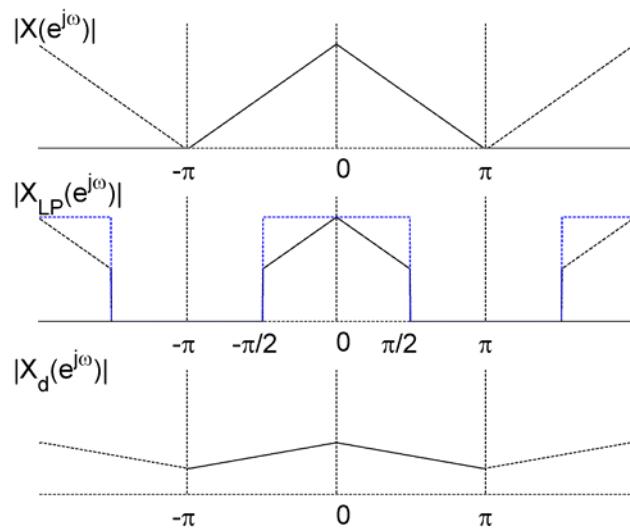
$$X_d(e^{j\omega}) = \frac{1}{M} \sum_{i=0}^{M-1} X\left(e^{j(\omega-2\pi i)/M}\right). \quad (5.23)$$

Widmo sygnału decymowanego  $x_d[n]$  składa się z  $M$  kopii widma sygnału  $x[n]$  przeskalowanych w częstotliwości, "rozciągniętych" przez  $M$  i przesuniętych o całkowite krotności  $2\pi/M$ .

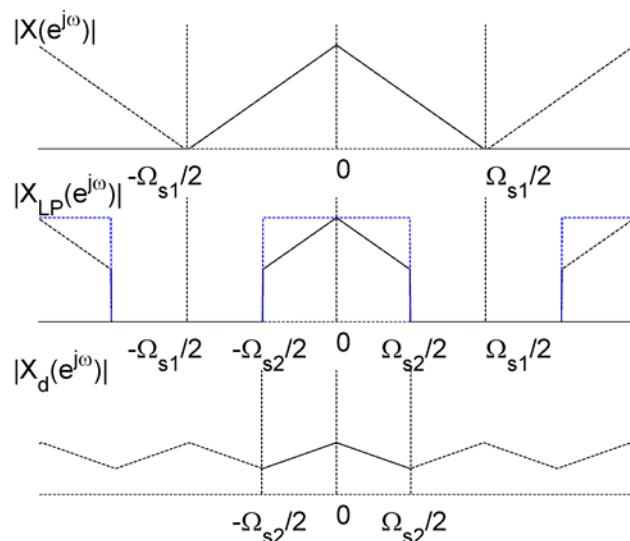
Ilustrację, w dziedzinie częstotliwości, dwukrotnej decymacji sygnału przedstawiają rys.5.15 i rys.5.16. Dla  $M = 2 \rightarrow \Omega_{s2} = \Omega_{s1}/2$ , widmo sygnału decymowanego wynosi:

$$X_d(e^{j\omega}) = \frac{1}{2}[X(e^{j\omega/2}) + X(e^{j(\omega-2\pi)/2})]. \quad (5.24)$$

Jak widać z rys.5.15 i rys.5.16, w pierwszym kroku ograniczane jest widmo sygnału, a następnie wybierane są próbki zgodnie z (5.20). W rezultacie sygnał decymowany zawiera  $M$  razy mniej próbek, a jego częstotliwość próbkowania jest  $M$  razy mniejsza, natomiast czas trwania wyrażony w sekundach jest taki sam.



Rys.5.15 Ilustracja operacji decymacji w dziedzinie częstotliwości - interpretacja dla pulsacji w [rad];  $X(e^{j\omega})$  - widmo sygnału,  $X_{LP}(e^{j\omega})$  - widmo sygnału po filtracji dolnoprzepustowej,  $X_d(e^{j\omega})$  - widmo sygnału po dwukrotnej decymacji.



Rys.5.16 Ilustracja operacji decymacji w dziedzinie częstotliwości - interpretacja dla częstotliwości próbkowania w [rad/s].  $X(e^{j\omega})$  - widmo sygnału,  $X_{LP}(e^{j\omega})$  - widmo sygnału po filtracji dolnoprzepustowej,  $X_d(e^{j\omega})$  - widmo sygnału po dwukrotnej decymacji.

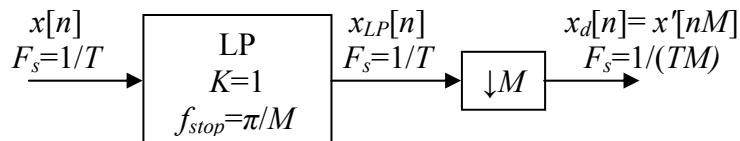
Rys.5.15 przedstawia interpretację operacji decymacji dla pulsacji w [rad]. Zaznaczone trójkątem widmo  $X(e^{j\omega})$  sygnału  $x[n]$  spróbkowanego z częstotliwością  $F_s$  zajmuje cały zakres pulsacji; następnie jest ono ograniczane za pomocą filtra dolnoprzepustowego. Po usunięciu próbki widmo  $X_d(e^{j\omega})$  decymowanego sygnału  $x_d[n]$ , spróbkowanego z częstotliwością  $F_s/2$ , ponownie zajmuje cały zakres pulsacji.

Rys.5.16 przedstawia interpretację operacji decymacji dla częstotliwości próbkowania w [rad/s], która maleje dwukrotnie.

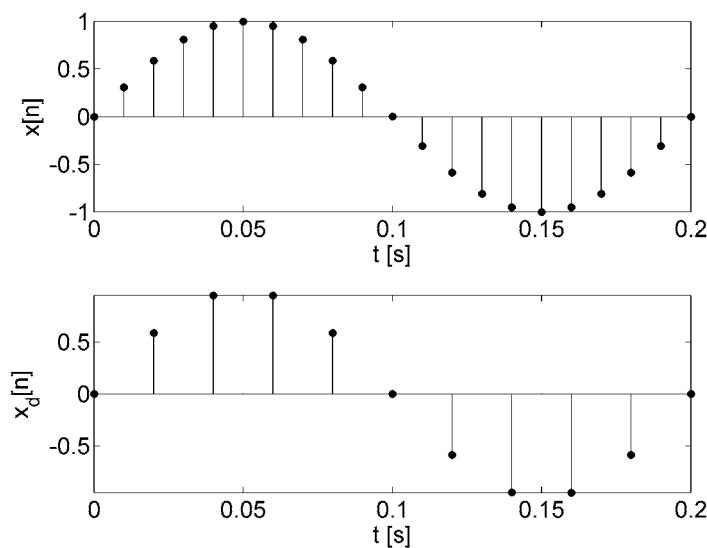
Rys.5.17 Przedstawia układ decymacji sygnałów, który składa się z filtra LP oraz układu decymatora (5.20) oznaczonego jako  $\downarrow M$ .

Decymacja może być przeprowadzana wielostopniowo, dzięki czemu uzyskuje się znaczną redukcję obliczeń oraz niezbędnej pamięci. Na przykład, przeprowadzenie decymacji sygnału o częstotliwości próbkowania  $F_{s1}=96$  kHz do  $F_{s2}=1$  kHz można uzyskać stosując jeden filtr i współczynnik  $M=96$ , lub trzy filtry ze współczynnikami decymacji  $M_1=8$ ,  $M_2=6$ ,  $M_3=2$ . Drugie rozwiązanie jest znacznie bardziej wydajne obliczeniowo. Przy decymacji wielostopniowej nieliniowości w pasmach przepustowych kolejnych filtrów dodają się, co należy uwzględnić przy projektowaniu filtrów składowych.

Dwukrotna decymacja sygnału w dziedzinie czasu przedstawiona jest na rys.5.18.



Rys.5.17 Układ decymacji sygnałów: LP - cyfrowy filtr dolnoprzepustowy,  $K$  - wzmocnienie filtru,  $f_{stop}$  - krawędź pasma zaporowego,  $x_{LP}[n]$  - sygnał po filtracji.

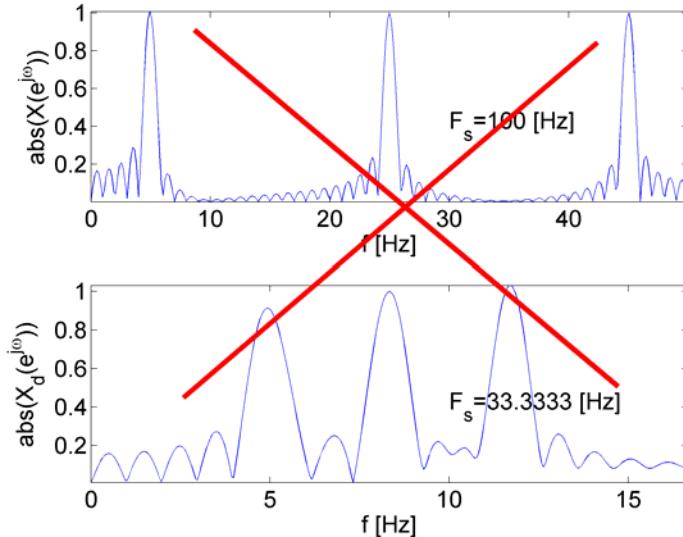


Rys.5.18 Dwukrotna decymacja sygnału.

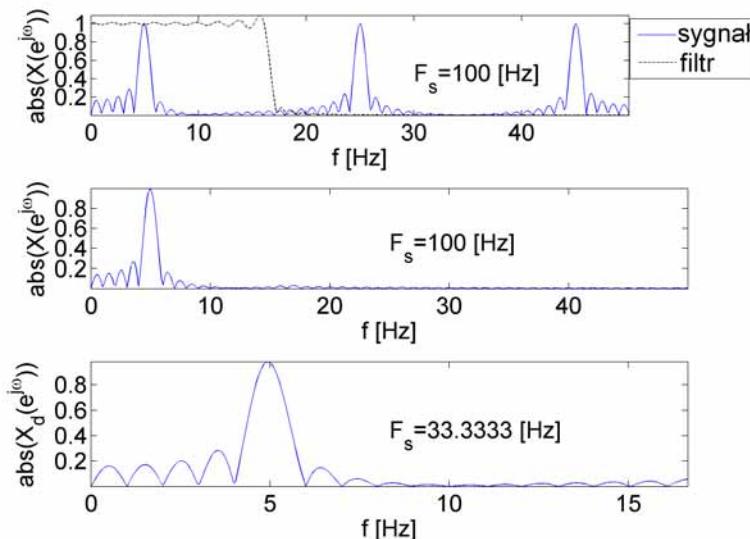
Rys.5.19 przedstawia widmo sygnału spróbkowanego z częstotliwością  $F_s=100$  Hz złożone z trzech częstotliwości: 5 Hz, 25 Hz i 45 Hz. Przed trzykrotną decymacją widmo tego sygnału musi zostać ograniczone do częstotliwości  $(100/3)/2 \approx 16.7$  Hz, aby było spełnione twierdzenie o próbkowaniu. Rys.5.19 przedstawia widmo sygnału po trzykrotnej decymacji bez użycia filtra antialiasingowego. W widmie pojawiają się częstotliwości, których nie było

w sygnale oryginalnym, są to odbicia częstotliwości 25 Hz i 45 Hz. Rys.5.19 obrazuje skutki nieprawidłowego przeprowadzenia decymacji.

Ten sam sygnał jest prawidłowo decymowany na rys.5.20. Za pomocą filtra LP widmo sygnału zostało ograniczone, a następnie przeprowadzono operację decymacji. Dla nowej częstotliwości próbkowania równej 33.3 Hz największa częstotliwość w sygnale może wynosić 16.7 Hz, dlatego częstotliwości 25 Hz i 45 Hz musiały zostać usunięte (odfiltrowane) przed decymacją.



Rys.5.19 Przykład nieprawidłowej decymacji sygnału - bez filtra antyaliasingowego.



Rys.5.20 Przykład poprawnej, trzykrotnej decymacji sygnału - z filtrem antyaliasingowym.

Zwiększenie częstotliwości próbkowania o czynnik całkowity (interpolacja, *upsampling*) polega na określeniu wartości dodatkowych próbek pomiędzy próbkami ciągu interpolowanego:

$$x_i[n] = x[n/L] = x_c(nT/L), \quad n = 0, \pm L, \pm 2L, \dots \quad (5.25)$$

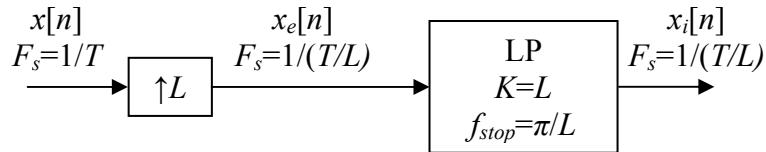
Dyskretna realizacja interpolacji składa się z ekspandera, tj. układu wstawiającego zero pomiędzy próbki sygnału i dolnoprzepustowego, cyfrowego filtra interpolującego o częstotliwości odcienia  $\pi/L$  i wzmacnieniu  $L$ .

Rys.5.21 przedstawia układ interpolacji sygnałów. Ekspander oznaczony jako  $\uparrow L$  jest opisany równaniem:

$$x_e[n] = \begin{cases} x[n/L], & n = 0, \pm L, \pm 2L, \dots \\ 0, & \text{inny} \end{cases} \quad (5.26)$$

lub równoważnie

$$x_e[n] = \sum_{k=-\infty}^{\infty} x[k] \delta[n - kL]. \quad (3.27)$$

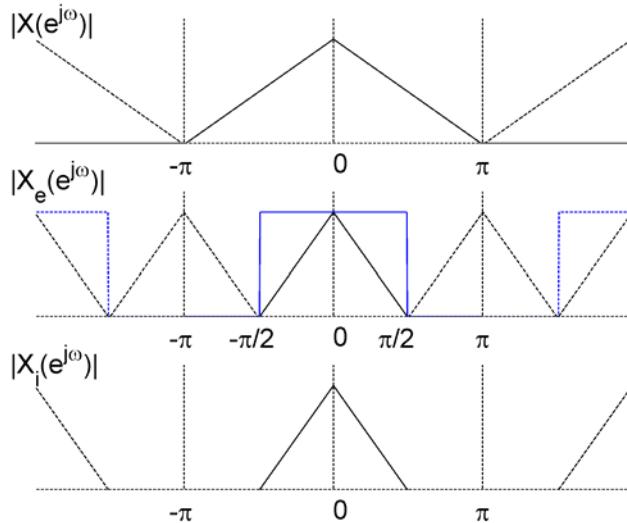


Rys.5.21 Układ interpolacji sygnałów: LP - cyfrowy filtr dolnoprzepustowy,  $K$  - wzmocnienie filtra,  $f_{stop}$  - krawędź pasma zaporowego.

Widmo sygnału  $x_e[n]$  jest następujące:

$$X_e(e^{j\omega}) = \sum_{n=-\infty}^{\infty} \left( \sum_{k=-\infty}^{\infty} x[k] \delta[n - kL] \right) e^{-j\omega n} = \sum_{k=-\infty}^{\infty} x[k] e^{-j\omega Lk} = X(e^{j\omega L}). \quad (5.28)$$

Widmo na wyjściu ekspandera jest przeskalowaną przez  $L$ , "ściśniętą" wersją widma na jego wejściu.

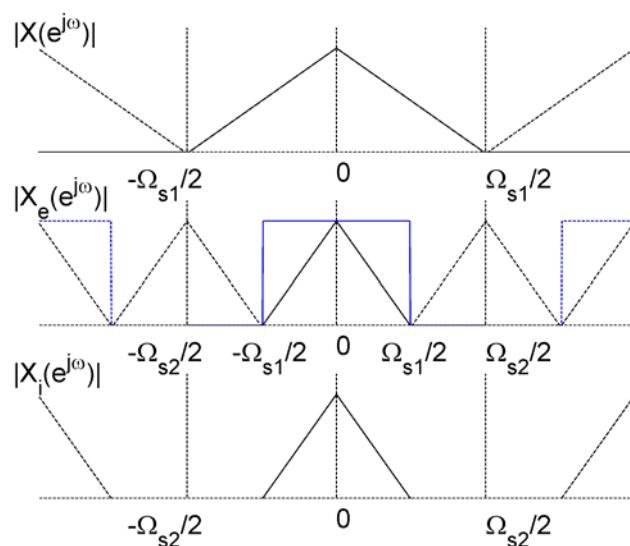


Rys.5.22 Ilustracja operacji interpolacji w dziedzinie częstotliwości - interpretacja dla pulsacji w [rad].

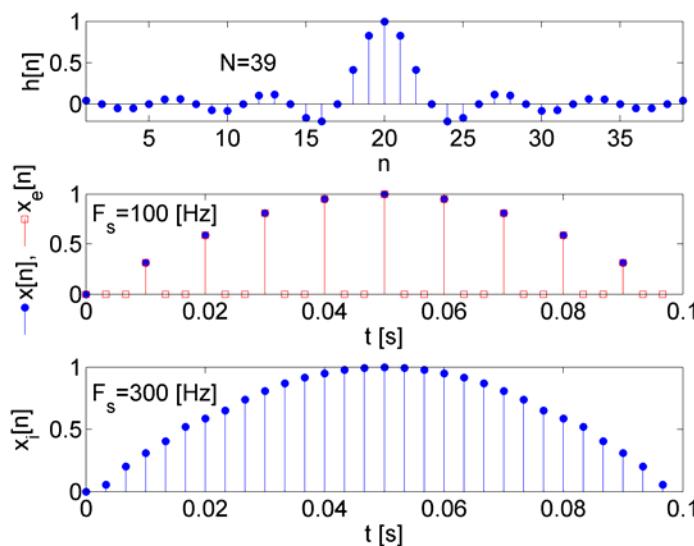
Rys.5.22 i rys.5.23 przedstawiają operację dwukrotnej interpolacji w dziedzinie częstotliwości, tj.  $L = 2$ ,  $X_e(e^{j\omega}) = X(e^{j2\omega})$ . Widmo sygnału po interpolacji  $X_i(e^{j\omega})$  uzyskuje się z widma sygnału na wyjściu ekspandera  $X_e(e^{j\omega})$  po usunięciu fragmentów widma, które nie są całkowitymi krotnościami  $2\pi$  oraz wzmocnieniu amplitud widma  $L$  razy. Do tego celu służy filtr LP.

Rys. 5.24 przedstawia przykład trzykrotniej interpolacji w dziedzinie czasu, a rys. 5.25 w dziedzinie częstotliwości.

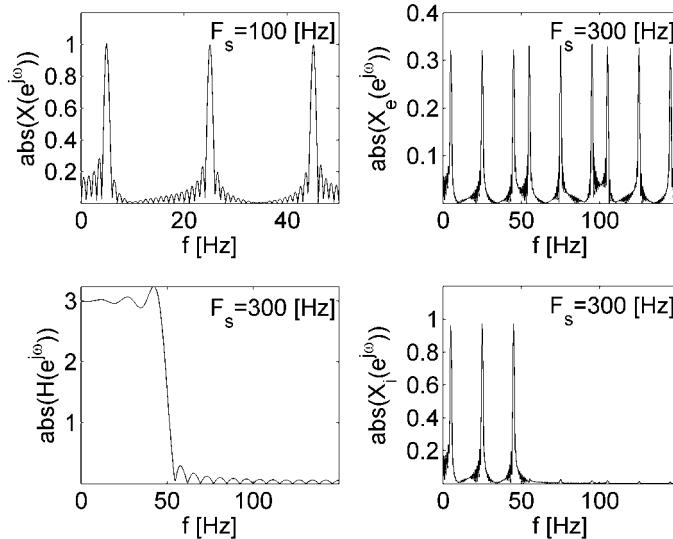
W przypadku, gdy zachodzi konieczność zmiany częstotliwości próbkowania o czynnik niecałkowitoliczbowy, stosuje się kaskadowe połączenie układu interpolatora i decymatora jak przedstawiono na rys.5.26. Ponieważ w układzie tym występuje kaskadowe połączenie dwóch filtrów LP, wystarczy zastosować jeden filtr o wzmacnieniu  $L$  i krawędzi pasma zaporowego odpowiadającej mniejszej częstotliwości. Częstotliwość próbkowania w układzie z rys.5.26 zostanie zmieniona  $L/M$  razy.



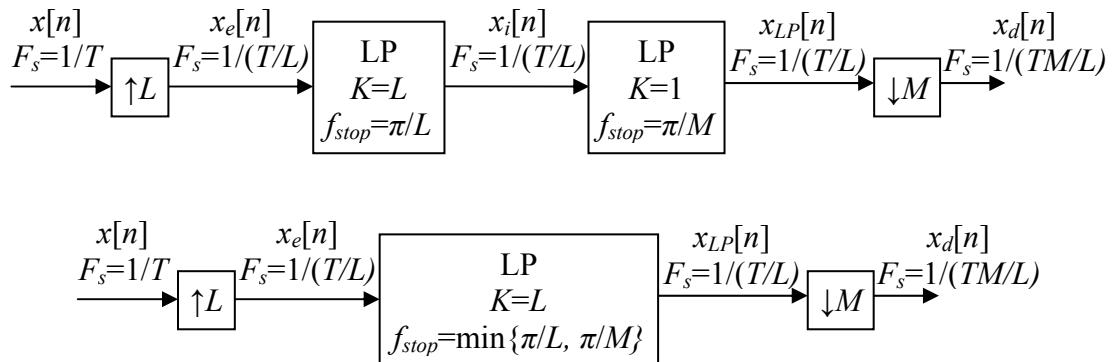
Rys.5.23 Ilustracja operacji interpolacji w dziedzinie częstotliwości - interpretacja dla częstotliwości próbkowania w [rad/s].



Rys.5.24 Trzykrotna interpolacja sygnału; od góry: odpowiedź impulsowa filtru interpolującego (dolnoprzepustowego), sygnał uzupełniony zerami, sygnał interpolowany.



Rys.5.25 Przykład trzykrotnej interpolacji sygnału; wykresy przedstawiają: widmo sygnału, widmo sygnału uzupełnionego zerami, charakterystyką amplitudową filtra LP i widmo sygnału interpolowanego.



Rys.5.26 Układ zmiany częstotliwości o czynniku niecałkowitoliczbowy.

Jednym z działań cyfrowego przetwarzania sygnałów jest przetwarzanie "wieloszybkościowe" (*multirate signal processing*), które zajmuje się tzw. reprezentacjami plolifazowymi sygnałów. Dekompozycja polifazowa ciągu dyskretnego polega na przedstawieniu go jako superpozycji  $M$  podciągów, z których każdy składa się z kolejnych  $M$ -tych wartości dekomponowanego ciągu. Dekompozycja taka zastosowana do odpowiedzi impulsowej filtra prowadzi do wydajnych obliczeniowo implementacji. Rozważmy odpowiedź impulsową  $h[n]$  zdekomponowaną na  $M$  podciągów  $h_k[n]$ ,  $k=0,1,2,\dots,M-1$ :

$$h_k[n] = \begin{cases} h[nM + k] \\ 0 \end{cases}. \quad (5.29)$$

Oryginalną odpowiedź impulsową można zrekonstruować w sposób następujący:

$$h[n] = \sum_{k=0}^{M-1} h_k[n - k]. \quad (5.30)$$

W dziedzinie transformacji  $Z$  polifazowa reprezentacja filtru ma postać:

$$H(z) = \sum_{k=0}^{M-1} H_k(z^M)z^{-k} \quad (5.31)$$

i jest ona przedstawiona na rys.5.27.

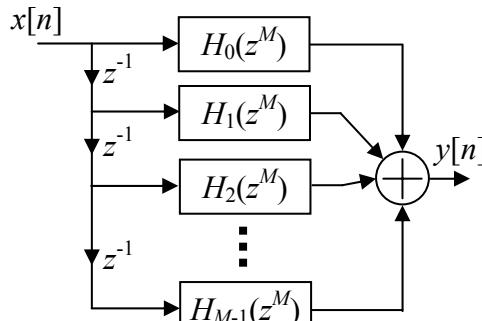
Jako przykład zastosowania dekompozycji polifazowej do implementacji filtrów rozważmy układ decymacji przedstawiony na rys.5.17. Postać polifazowa tego układu jest przedstawiona na rys.5.28. Można zauważyć, że najpierw wykonywana jest filtracja składowych polifazowych, a następnie wykonywana jest decymacja. Takie podejście jest nieefektywne obliczeniowo. Możliwa jest zamiana kolejności tych operacji, co prowadzi do wydajnej obliczeniowo struktury pokazanej na rys.5.31, w której dane są najpierw decymowane, a następnie filtrowane.

Rys.5.29 przedstawia dwa równoważne układy decymacji. Widma sygnałów z rys.5.29 są następujące

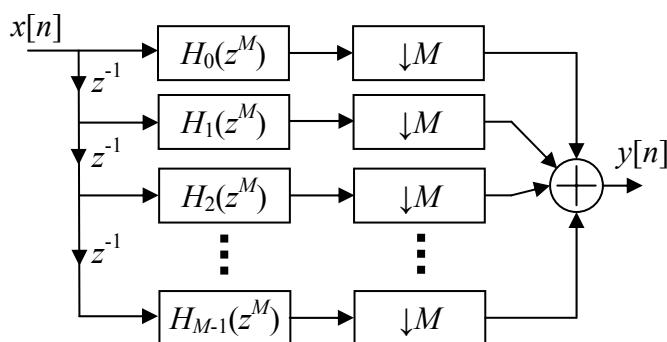
$$X_b(e^{j\omega}) = H(e^{j\omega M})X(e^{j\omega}), \quad (5.32)$$

i

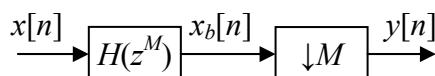
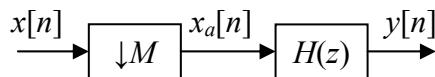
$$Y(e^{j\omega}) = \frac{1}{M} \sum_{i=0}^{M-1} X_b\left(e^{j(\omega/M - 2\pi i/M)}\right). \quad (5.33)$$



Rys.5.27 Reprezentacja polifazowa filtru.



Rys.5.28 Reprezentacja polifazowa układu decymacji.



Rys.5.29 Tożsamość decymacji.

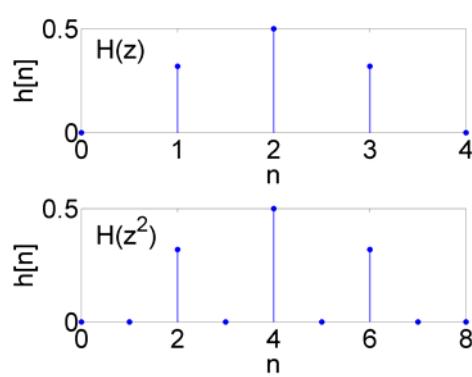
Podstawiając (5.32) do (5.33) otrzymujemy

$$Y(e^{j\omega}) = \frac{1}{M} \sum_{i=0}^{M-1} X\left(e^{j(\omega/M - 2\pi i/M)}\right) H\left(e^{j(\omega - 2\pi i)}\right). \quad (5.34)$$

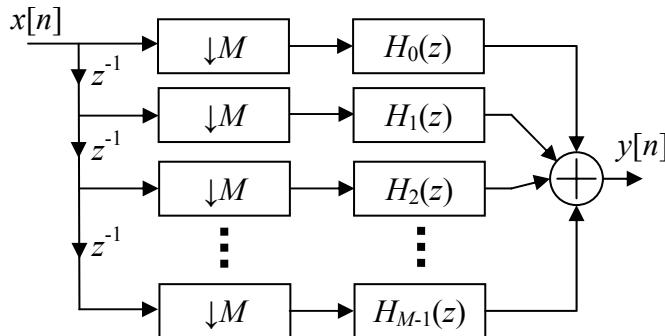
Ponieważ  $H(e^{j(\omega - 2\pi i)}) = H(e^{j\omega})$ ,

$$Y(e^{j\omega}) = H(e^{j\omega}) \frac{1}{M} \sum_{i=0}^{M-1} X\left(e^{j(\omega/M - 2\pi i/M)}\right) = H(e^{j\omega}) X_a(e^{j\omega}), \quad (5.35)$$

co oznacza, że oba układy z rys.5.29 są tożsame. Rys.5.30 przedstawia interpretację notacji  $H(z^M)$  dla  $M=2$ , a rys.5.32 przedstawia odpowiedź impulsową filtra układowej decymacji o długości 15 współczynników oraz jej trzy składowe polifazowe.

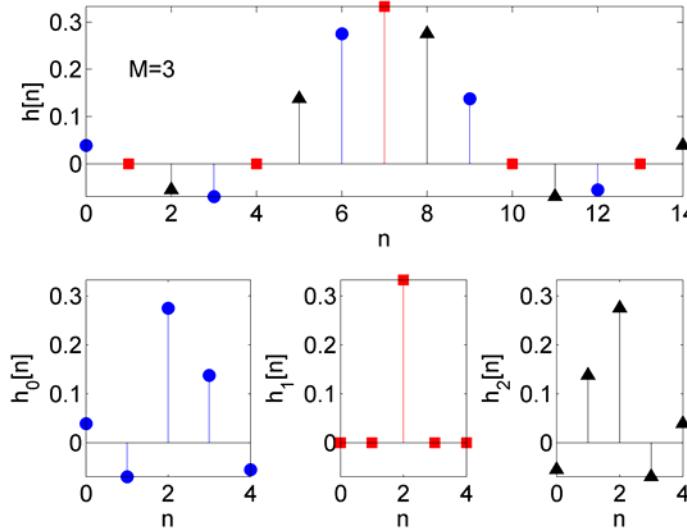


Rys.5.30 Interpretacja notacji  $H(z^M)$ , dla  $M=2$ ;  $H(z) = h_0z^0 + h_1z^1 + h_2z^2 + h_3z^3 + h_4z^4$ ,  
 $H(z^2) = h_0z^0 + 0 + h_1z^2 + 0 + h_2z^4 + 0 + h_3z^6 + 0 + h_4z^8$ .



Rys.5.31 Reprezentacja polifazowa układu decymacji wykorzystująca tożsamość decymacji.

W przypadku, pokazanej na rys.5.17, bezpośredniej implementacji układu decymacji obliczenie splotu wymaga wykonania  $N$  mnożeń i  $N-1$  dodawań, w jednym okresie próbkowania, gdzie  $N$  jest długością filtra. Natomiast w przypadku implementacji polifazowej z rys.5.31 sygnały na wejściach filtrów zmieniają się  $M$  razy wolniej, tzn. nowa próbka pojawia się po  $M$  okresach próbkowania, a filtry te są  $M$  razy krótsze. Dlatego implementacja polifazowa wymaga przeprowadzenia  $1/M \cdot (N/M)$  mnożeń i  $1/M \cdot (N/M-1)$  dodawań na każdy filtr w jednym okresie próbkowania, a cały układ decymacji wymaga  $(N/M)$  mnożeń i  $(N/M-1)+(M-1)$  dodawań w jednym okresie próbkowania, co dla właściwego doboru parametrów  $N$  i  $M$  może prowadzić do znaczającej redukcji wymaganej mocy obliczeniowej.



Rys.5.32 Filtr  $h[n]$  układu decymacji i jego dekompozycja na trzy składowe polifazowe  
 $h_k[n]=h[nM+k]$ .

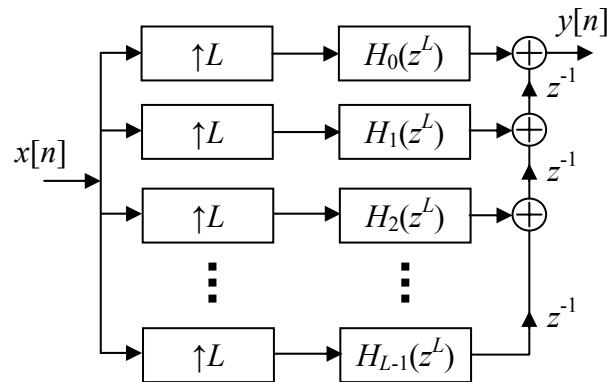
Reprezentacja polifazowa układu interpolacji przedstawiona jest na rys.5.33, a jej wersja wydajna obliczeniowo na rys.5.35. W przekształceniu rys.5.33 w rys.5.35 zastosowano tożsamości interpolacji zilustrowaną na rys.5.34. Z rys.5.34 otrzymujemy

$$Y(e^{j\omega}) = X_a(e^{j\omega L}) = X(e^{j\omega L})H(e^{j\omega L}), \quad (5.36)$$

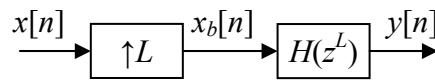
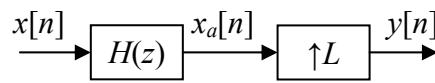
a po podstawieniu  $X_b(e^{j\omega}) = X(e^{j\omega L})$ :

$$Y(e^{j\omega}) = H(e^{j\omega L})X_b(e^{j\omega}), \quad (5.37)$$

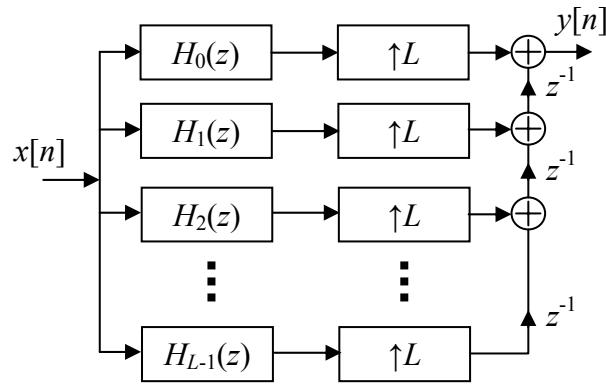
co potwierdza równoważność obu układów na rys.5.34.



Rys.5.33 Reprezentacja polifazowa układu interpolacji.



Rys.5.34 Tożsamość interpolacji.



Rys.5.35 Reprezentacja polifazowa układu interpolacji wykorzystująca tożsamość interpolacji.

#### 5.4 Cyfrowy tor analizy i przetwarzania sygnałów

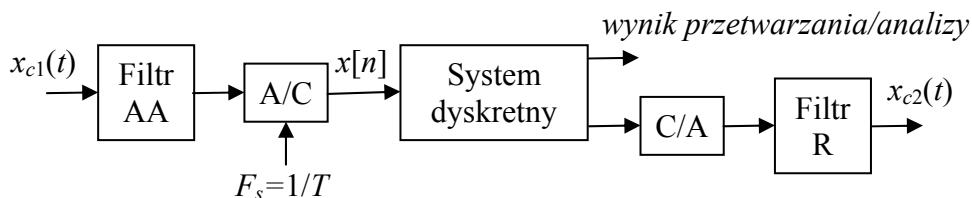
Rys.5.36 przedstawia schemat blokowy układu cyfrowego przeznaczonego do przetwarzania i analizy sygnałów analogowych. Źródłem sygnałów analogowych może być dowolny czujnik np. mikrofon rejestrujący fonokardiogram, czy elektrody rejestrujące elektrokardiogram.

Pierwszym krokiem przetwarzania sygnału analogowego jest ograniczenie jego pasma filtrem antyaliasingowym. Następnie sygnał analogowy jest próbkowany i kwantowany za pomocą przetworników analogowo-cyfrowych, w wyniku czego uzyskujemy sygnał cyfrowy.

Sygnał cyfrowy jest przetwarzany za pomocą układów dyskretnych. Celem przetwarzania może być wydobycie z sygnału interesującej informacji, np. detekcja i pomiar załamków w sygnale EKG lub też zmiana reprezentacji sygnału np. kompresja lub modulacja w celu transmisji.

Wyjściem z toru przetwarzania może być wynik analizy sygnału, np. określenie wybranych parametrów lub sygnał analogowy. W drugim przypadku stosowane są przetworniki cyfrowo-analogowe zawierające analogowy filtr rekonstrukcji. Przykładem sygnału przechodzącego cały tor przetwarzania z rys.5.36 jest muzyka CD, rejestrowana za pomocą mikrofonów analogowych, próbkowana i kwantowana, miksuowana cyfrowo za pomocą układów dyskretnych i zamieniana na postać analogową w odtwarzaczu CD.

Ze względu na koszt obliczeń (szybkość) działania pożąданie jest stosowanie niskich częstotliwości próbkowania. Wymaga to stosowania stromych, analogowych filtrów antyaliasingowych, które są trudne w realizacji, drogie i mają niekorzystne parametry (nieliniowa faza, możliwość odstrajania, wrażliwość temperaturowa). Rozwiązaniem tego problemu może być nadpróbkowanie sygnału umożliwiające stosowanie łagodnych filtrów analogowych niskiego rzędu, a następnie decymacja sygnału z cyfrowym filtrem antyaliasingowym, który dodatkowo może mieć liniową charakterystykę fazową.



Rys.5.36 Cyfrowy tor analizy i przetwarzania sygnałów: AA - analogowy filtr antyaliasingowy, A/C - przetwornik analogowo-cyfrowy, C/A - przetwornik cyfrowo - analogowy, R - analogowy filtr rekonstrukcji.

## 6. Projektowanie filtrów analogowych

Teoria filtrów analogowych została opracowana jako pierwsza, dlatego projektowanie rekursywnych filtrów cyfrowych najczęściej wykorzystuje prototypy analogowe. Bezpośrednie projektowanie filtrów analogowych jest łatwiejsze niż filtrów cyfrowych, ponieważ ich transmitancja jest funkcją  $j\Omega$ , a nie  $e^{j\omega}$ , jak ma to miejsce dla filtrów cyfrowych.

### 6.1 Filtre Butterwortha

Charakterystyka częstotliwościowa  $|H(j\Omega)|$  filtra Butterwortha jest maksymalnie płaska w paśmie przepustowym. Oznacza to, że dla filtra rzędu  $N$  pierwsze  $(2N-1)$  pochodnych funkcji  $|H(j\Omega)|^2$  ma wartość zero dla  $\Omega=0$ . Charakterystyka amplitudowa jest monotoniczna w paśmie przepustowym i zaporowym. Kwadrat charakterystyki amplitudowej definiuje się następująco:

$$|H_c(j\Omega)|^2 = \frac{1}{1 + (j\Omega / j\Omega_c)^{2N}}. \quad (6.1)$$

Podstawiając  $s=j\Omega$  do (6.1) otrzymujemy:

$$|H_c(s)|^2 = H_c(s)H_c(-s) = \frac{1}{1 + (s / j\Omega_c)^{2N}}. \quad (6.2)$$

Bieguny transmitancji spełniają zależność  $1 + (s / j\Omega_c)^{2N} = 0$ , z której, po wyznaczeniu pierwiastków wg wzoru  $C = Me^{j\varphi} = Me^{j(\varphi+k2\pi)}$ ,  $\sqrt[n]{C} = \sqrt[n]{M} e^{j(\varphi+k2\pi)/n}$ ,  $k = 0, 1, \dots, n-1$ , otrzymujemy wyrażenie na bieguny transmitancji filtra Butterwortha:

$$s_k = j\sqrt[2N]{-1}\Omega_c = \Omega_c e^{j(\pi/2N)(2k+N-1)}, \quad k = 0, 1, \dots, 2N-1. \quad (6.3)$$

Ponieważ (6.3) podaje bieguny iloczynu  $H_c(s)H_c(-s)$ , do realizacji filtra wybiera się bieguny z lewej półpłaszczyzny, tzn. z ujemną częścią rzeczywistą, co zapewnia stabilność filtra analogowego.

Rys.6.1 przedstawia rodzinę charakterystyk amplitudowych filtra Butterwortha dla różnych rzędów  $N$ . Wraz ze wzrostem rzędu filtra charakterystyki amplitudowe stają się coraz bardziej prostokątne, tzn. zbliżają się do charakterystyki idealnej. Wszystkie krzywe przecinają się w punkcie  $\Omega_c$ , w którym wzmacnienie wynosi:

$$|H_c(j\Omega_c)| = \sqrt{\frac{1}{1 + (j\Omega_c / j\Omega_c)^{2N}}} = \frac{1}{\sqrt{2}}, \quad (6.4)$$

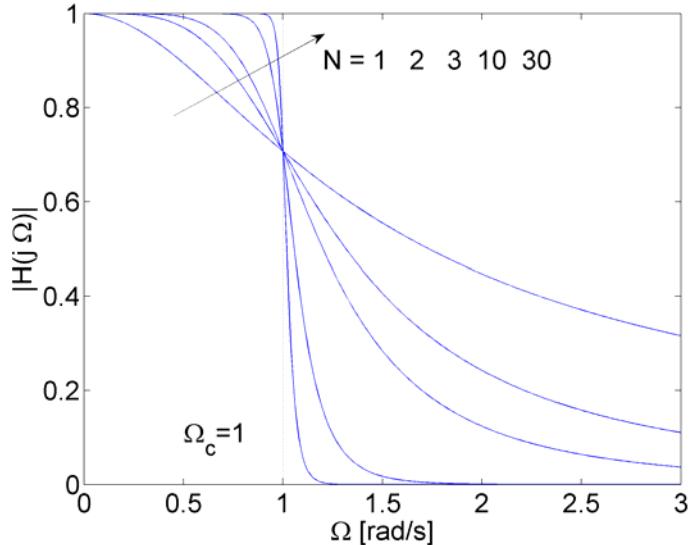
a w skali decybelowej  $|H_c(j\Omega_c)| = 20 \log_{10}(1/\sqrt{2}) = -3 \text{ dB}$ .  $\Omega_c$  jest to 3-decybowa częstotliwość filtra Butterwortha.

Rys.6.2 przedstawia przykłady położenia biegunków transmitancji filtra Butterwortha. W każdym przypadku ze wzoru (6.3) otrzymuje się  $2N$  biegunków, spośród których do transmitancji wybierane są tylko te leżące w lewej półpłaszczyźnie. Biegunki, oraz transmitancje dla filtrów z rys.6.2 są następujące:

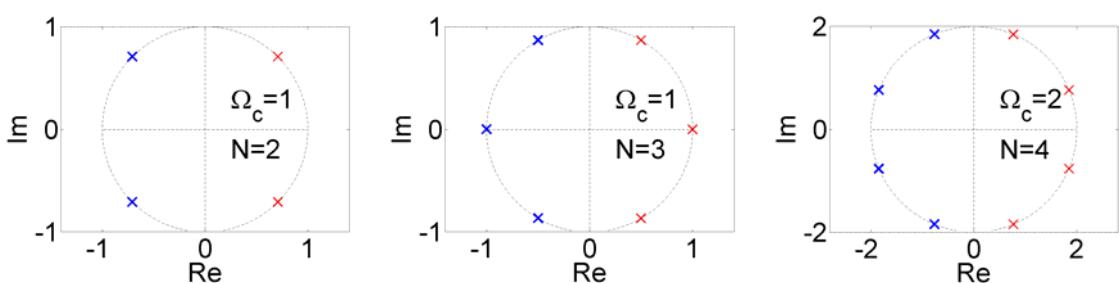
1.  $s_{1,2} = -0.7071 \pm j0.7071$ ,  $H_c(s) = \frac{1}{(s + 0.7071 + j0.7071)(s + 0.7071 - j0.7071)}$ ,  $H_c(s) = \frac{1}{s^2 + 1.4142s + 1}$
2.  $s_{1,3} = -0.5 \pm j0.866$ ,  $s_2 = 1$ ,  $H_c(s) = \frac{1}{s^3 + 2s^2 + 2s + 1}$

$$3. \ s_{1,4} = -0.7654 \pm j1.8478, \ s_{2,3} = -1.8478 \pm j0.7654, \ w = \Omega_c^N = 2^4 = 16$$

$$H_c(s) = \frac{16}{s^4 + 5.2263s^3 + 13.6569s^2 + 20.905s + 16}$$



Rys.6.1 Charakterystyki amplitudowe filtra Butterwortha.



Rys.6.2 Położenie biegunków transmitancji  $H(s)$  filtra Butterwortha.

W przypadku, kiedy  $\Omega_c$  jest różne od 1, należy uwzględnić wzmacnianie filtra w liczniku transmitancji  $w = \Omega_c^N$ .

Jako przykład zaprojektujemy filtr Butterwortha, tzn. określmy jego rzad  $N$  i częstotliwość 3 dB  $\Omega_c$ , spełniający następujące wymagania: krawędź pasma przepustowego  $\Omega_{pass}=100$  rad/s, krawędź pasma zaporowego  $\Omega_{stop}=160$  rad/s, maksymalne tłumienie w paśmie przepustowym  $r_p=1$  dB, minimalne tłumienie w paśmie zaporowym  $r_s=30$  dB.

Na podstawie wzoru (6.1) wymagania odnośnie krawędzi pasma przepustowego i zaporowego możemy zapisać w postaci układu równań:

$$\begin{cases} 20\log_{10}\left(1/\sqrt{1+(\Omega_{pass}/\Omega_c)^{2N}}\right) = -r_p [dB] \\ 20\log_{10}\left(1/\sqrt{1+(\Omega_{stop}/\Omega_c)^{2N}}\right) = -r_s [dB] \end{cases} \quad (6.5)$$

Wykonując kolejne przekształcenia otrzymujemy:

$$\begin{cases} 20\log_{10}\left(1+(\Omega_{pass}/\Omega_c)^{2N}\right)^{-1/2} = -r_p, \\ 20\log_{10}\left(1+(\Omega_{stop}/\Omega_c)^{2N}\right)^{-1/2} = -r_s \end{cases} \quad (6.6)$$

$$\begin{cases} \log_{10}(1 + (\Omega_{pass}/\Omega_c)^{2N}) = r_p/10 \\ \log_{10}(1 + (\Omega_{stop}/\Omega_c)^{2N}) = r_s/10 \end{cases}, \quad (6.7)$$

$$\begin{cases} (\Omega_{pass}/\Omega_c)^{2N} = 10^{r_p/10} - 1 \\ (\Omega_{stop}/\Omega_c)^{2N} = 10^{r_s/10} - 1 \end{cases}. \quad (6.8)$$

Dzieląc stronami i logarytmując:

$$\log_{10}\left(\frac{\Omega_{pass}}{\Omega_{stop}}\right)^{2N} = \log_{10}\left(\frac{10^{r_p/10} - 1}{10^{r_s/10} - 1}\right) \quad (6.9)$$

otrzymujemy zależność na rząd filtra:

$$N = \frac{1}{2} \log_{10}\left(\frac{10^{r_p/10} - 1}{10^{r_s/10} - 1}\right) / \log_{10}\left(\frac{\Omega_{pass}}{\Omega_{stop}}\right) = \frac{1}{2} \log_{10}\left(\frac{10^{1/10} - 1}{10^{30/10} - 1}\right) / \log_{10}\left(\frac{100}{160}\right) = 8.785 \quad (6.10)$$

Ponieważ rząd filtra musi być liczbą całkowitą,  $N$  zostaje zaokrąglone w górę  $N=9$ .  $\Omega_c$  można wyznaczyć np. z równań (6.8):

$$\log_{10}(\Omega_{pass}/\Omega_c) = \frac{1}{2N} \log_{10}(10^{r_p/10} - 1), \quad (6.11)$$

$$\log_{10}(\Omega_{pass}/\Omega_c) = \log_{10}(10^{r_p/10} - 1)^{\frac{1}{2N}}, \quad (6.12)$$

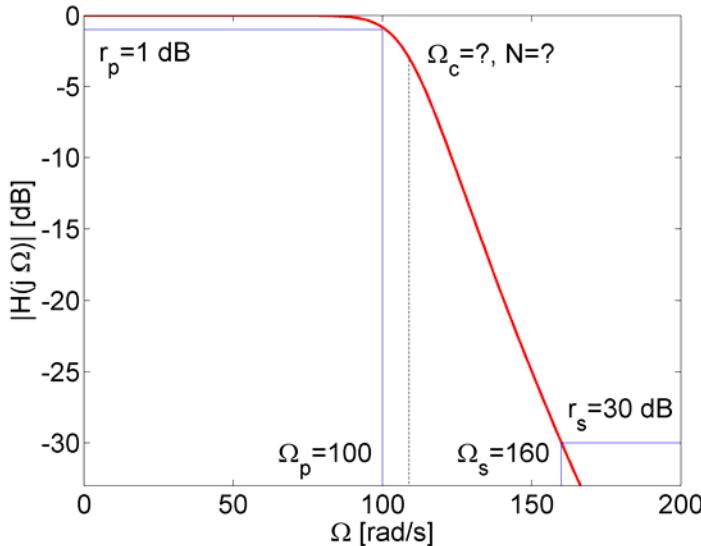
$$\Omega_{pass}/\Omega_c = (10^{r_p/10} - 1)^{\frac{1}{2N}}, \quad (6.13)$$

$$\Omega_c = \frac{\Omega_{pass}}{(10^{r_p/10} - 1)^{\frac{1}{2N}}} = \frac{100}{(10^{1/10} - 1)^{\frac{1}{2N}}} = 107.7957. \quad (6.14)$$

Rys.6.3 Przedstawia wymagania projektowe i charakterystykę amplitudową zaprojektowanego filtra Butterwortha. Jak widać, w paśmie przepustowym charakterystyka opada o 1 dB dla 100 rad/s, natomiast w paśmie zaporowym tłumienie dla 160 rad/s wynosi 30 dB, tak więc zaprojektowany filtr spełnia zadane wymagania.

Sprawdzenie poprawności projektu na podstawie charakterystyki amplitudowej jest niezbędnym etapem projektu.

Ponieważ filtr musi spełnić oba wymagania odnośnie krawędzi pasm, więc może się zdarzyć, że jedno z nich będzie spełnione z nadmiarem.



Rys.6.3 Wymagania projektowe i charakterystyka amplitudowa zaprojektowanego filtra Butterwortha.

## 6.2 Filtry Czebyszewa

Kwadrat charakterystyki amplitudowej filtra Czebyszewa typu I definiuje się następująco:

$$|H_c(j\Omega)|^2 = \frac{1}{1 + \varepsilon^2 V_N^2(j\Omega / j\Omega_c)}, \quad (6.15)$$

gdzie

$$V_N(x) = \begin{cases} \cos(N \arccos(x)), & |x| \leq 1 \\ \cosh(N \text{arccosh}(x)), & |x| > 1 \end{cases} \quad (6.16)$$

jest wielomianem Czebyszewa rzędu  $N$ . Wielomiany Czebyszewa można wyznaczać rekurencyjnie:

$$V_{N+1}(x) = 2xV_N(x) - V_{N-1}(x), \quad N \geq 1. \quad (6.17)$$

Pierwsze trzy wielomiany są następujące:

$$V_0(x) = \cos(0 \arccos(x)) = 1, \quad V_1(x) = \cos(\arccos(x)) = x, \quad V_2(x) = \cos(2 \arccos(x)) = 2x^2 - 1. \quad (6.18)$$

Dla  $|x| \leq 1 \quad 0 \leq V_N^2(x) \leq 1$ , dla  $|x| > 1 \quad V_N(x)$  rośnie monotonicznie,  $V_N(1) = 1$ , a  $V_N(0) = \pm 1$  dla  $N$  parzystego i  $V_N(0) = 0$  dla  $N$  nieparzystego.

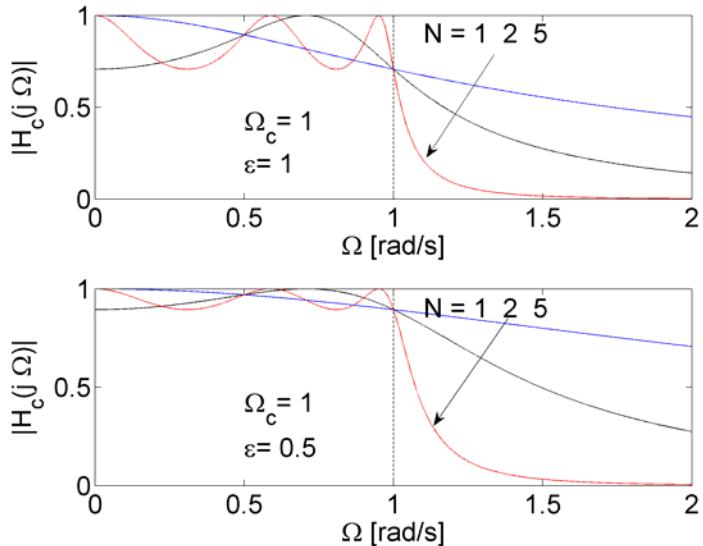
Charakterystyka filtra Czebyszewa typu I jest równomiernie falista w paśmie przepustowym, co pokazano na rys.6.4. Amplituda tego zafalowania jest regulowana parametrem  $\varepsilon$ . W paśmie zaporowym charakterystyka opada monotonicznie.

Biegunki analogowego filtra Czebyszewa typu I są położone na płaszczyźnie zespolonej na elipsie. Długość półosi małej wynosi  $a\Omega_c$  a półosi wielkiej  $b\Omega_c$ , przy czym:

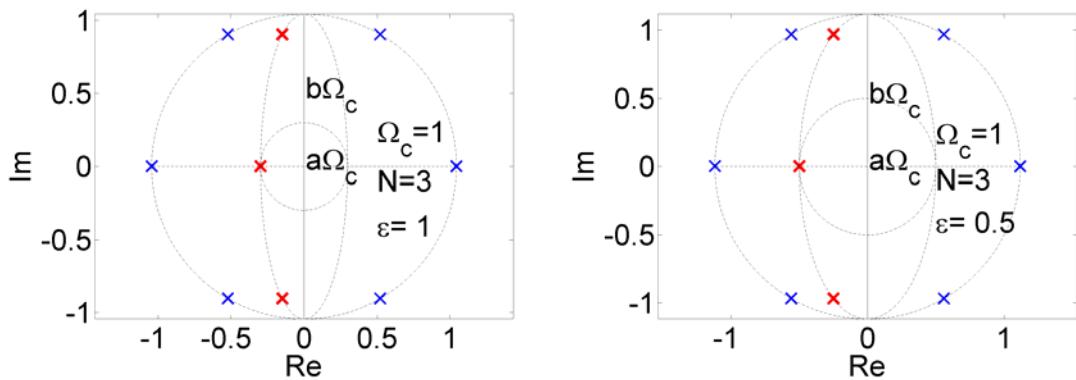
$$a = \frac{1}{2}(\alpha^{1/N} - \alpha^{-1/N}), \quad \alpha = \varepsilon^{-1} + \sqrt{1 + \varepsilon^{-2}}, \quad b = \frac{1}{2}(\alpha^{1/N} + \alpha^{-1/N}). \quad (6.19)$$

Biegunki transmitancji filtra Czebyszewa typu I można wyznaczyć na podstawie biegunków filtra Butterwortha. Na okręgu o promieniu równym wielkiej półosi elipsy wyznacza się biegunki filtra Butterwortha o tym samym rzędzie, co projektowany filtr

Czebyszewa, a następnie przesuwa się je na obwód elipsy, co zilustrowano na rys.6.5. Biegunki transmitancji w przykładzie na rys.6.5 wynoszą  $s_{1,2}=-0.1490\pm j0.9037$ ,  $s_3=-0.2980$  oraz  $s_{1,2}=-0.2500\pm j0.9682$ ,  $s_3=-0.5$ .



Rys.6.4 Charakterystyki amplitudowe filtru Czebyszewa typu I.



Rys.6.5 Położenie biegunków transmitancji  $H(s)$  filtru Czebyszewa typu I, wyznaczone poprzez biegunki transmitancji  $H(s)$  filtru Butterwortha.

Kwadrat charakterystyki amplitudowej filtru Czebyszewa typu II definiuje się następująco:

$$|H_c(j\Omega)|^2 = \frac{1}{1 + \frac{1}{\varepsilon^2 V_N^2(j\Omega_c / j\Omega)}}. \quad (6.20)$$

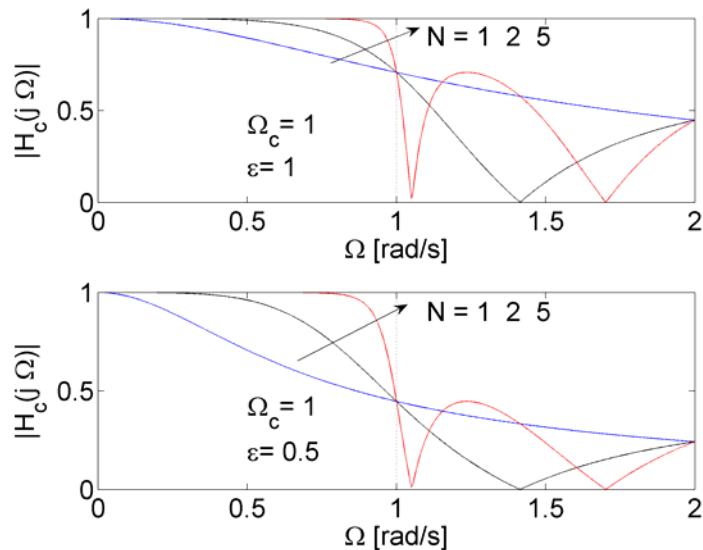
W porównaniu z filtrem typu I następuje odwrócenie czonu  $\varepsilon^2 V_N^2(j\Omega / j\Omega_c)$  w mianowniku oraz odwrócenie argumentu wielomianu Czebyszewa na  $j\Omega_c / j\Omega$ . Charakterystyka amplitudowa filtru Czebyszewa typu II, pokazana na rys.6.6, jest płaska w paśmie przepustowym i równomiernie falista w paśmie zaporowym. Parametr  $\varepsilon$  reguluje amplitudę zafalowania charakterystyki amplitudowej w paśmie zaporowym.

Biegunki filtru typu II można wyznaczyć znając biegunki filtru typu I na podstawie zależności:

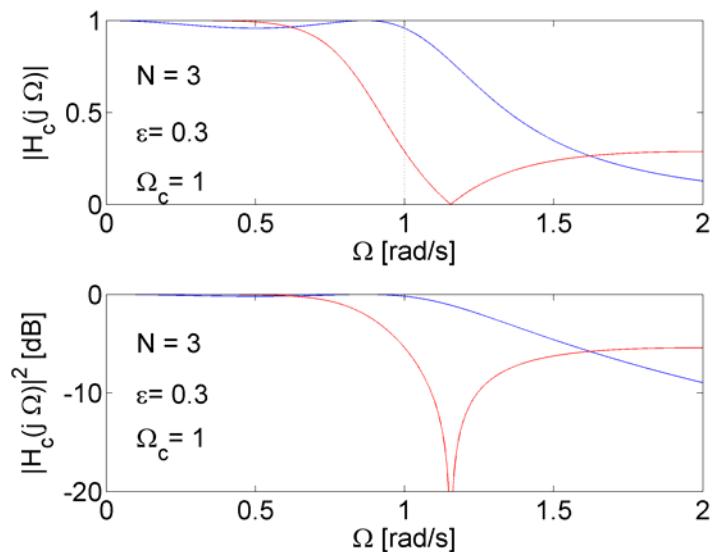
$$s_k^{II} = \frac{\Omega_c^2}{s_k^I}, \quad (6.21)$$

natomiast zera filtra Czebyszewa typu II leżą na osi urojonej w punktach  $V_N(\Omega_c/\Omega) = 0$ .

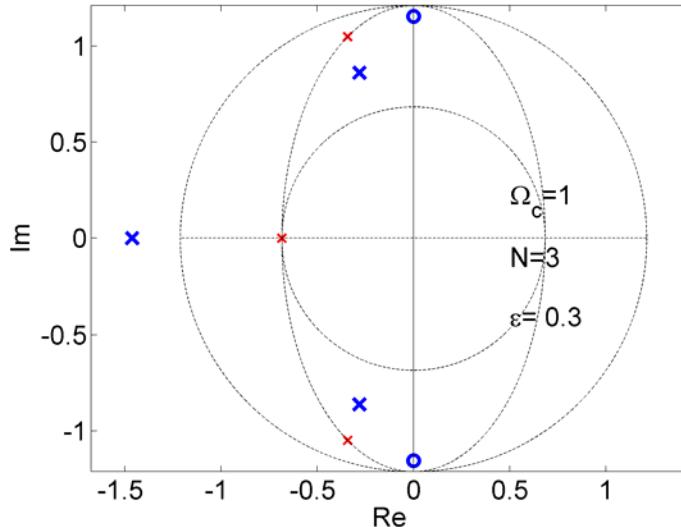
Rys.6.7 przedstawia charakterystyki amplitudowe filtra Czebyszewa typu I i utworzonego z niego filtra Czebyszewa typu II. Na rys.6.8 pokazane są biegunki obu tych filtrów oraz zera filtra Czebyszewa typu II. Biegunki transmitancji filtra Czebyszewa typu II wynoszą  $s_{1,2}=-0.2808 \pm 0.8615$ ,  $s_3=-1.4617$  a zera  $z_{1,2}=\pm j1.1547$ .



Rys.6.6 Charakterystyki amplitudowe filtra Czebyszewa typu II.



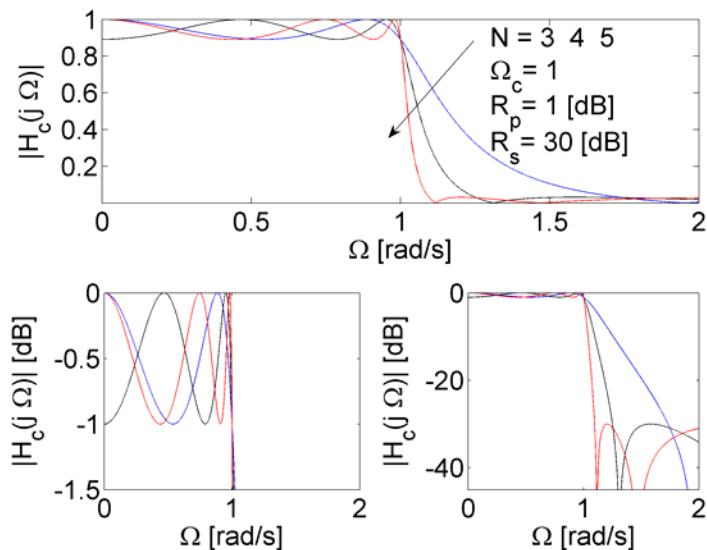
Rys.6.7 Charakterystyki amplitudowe filtra Czebyszewa typu I i utworzonego z niego filtra Czebyszewa typu II.



Rys.6.8 Położenie zer i biegunów transmitancji  $H(s)$  filtra Czebyszewa typu II, wyznaczone poprzez bieguny filtra Czebyszewa typu I wg (6.21).

### 6.3 Filtr eliptyczny

Charakterystyka amplitudowa filtru eliptycznego, przedstawiona na rys.6.9, jest równomiernie falista w paśmie przepustowym i zaporowym. Filtr eliptyczny opisują cztery parametry:  $N$  - rzad filtru,  $\Omega_c$  - krawędź pasma przepustowego oraz  $R_p$  - tłumienie (zafalowanie) w paśmie przepustowym i  $R_s$  - tłumienie w paśmie zaporowym.



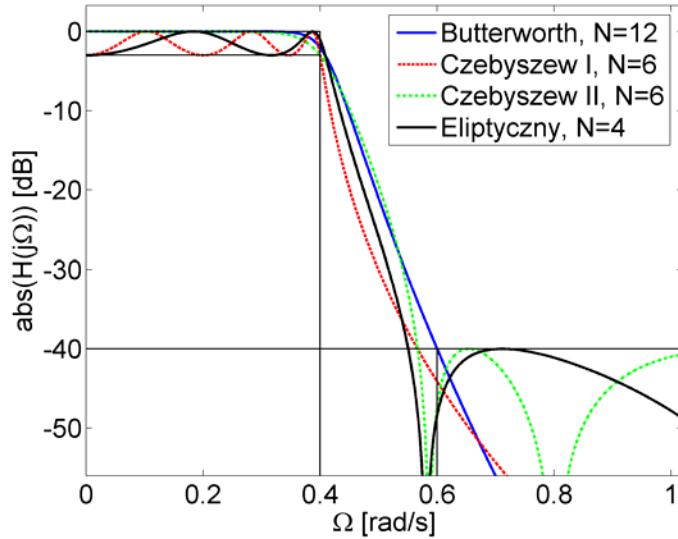
Rys.6.9 Charakterystyki amplitudowe filtru eliptycznego.

### 6.4 Porównanie charakterystyk filtrów analogowych

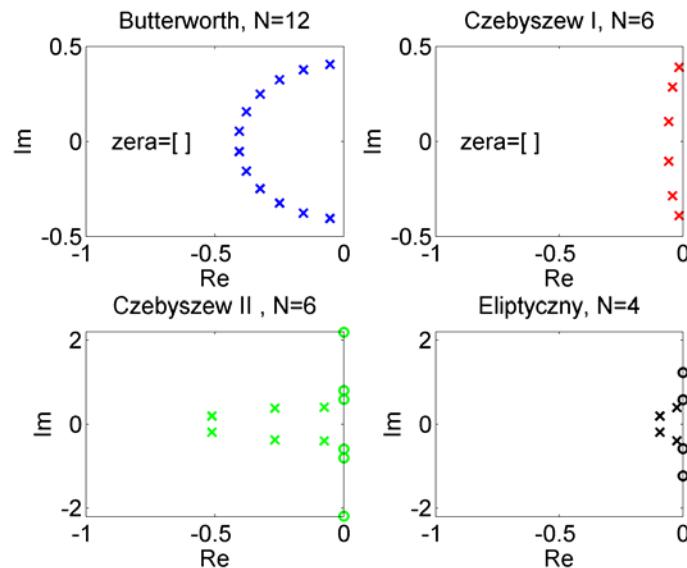
Rysunki 6.10-6.13 przedstawiają porównanie omówionych filtrów analogowych zaprojektowanych na te same wymagania odnośnie charakterystyki amplitudowej. Rys. 6.10 pokazuje, że dla tych samych wymagań rzędy filtrów są różne. Pod tym względem najlepszy jest filtr o najniższym rzędzie, czyli eliptyczny. Z drugiej strony charakterystyka filtra Butterwortha opada jednostajnie (jest płaska) zarówno w paśmie przepustowym jak i zaporowym, podczas gdy charakterystyka filtra eliptycznego faluje w obu tych pasmach na określonym poziomie, co jest jego wadą.

Rys.6.11 przedstawia położenie zer i biegunów transmitancji filtrów analogowych. Jak widać, filtry dolnoprzepustowe Butterwortha i Czebyszewa typu I nie mają zer transmitancji.

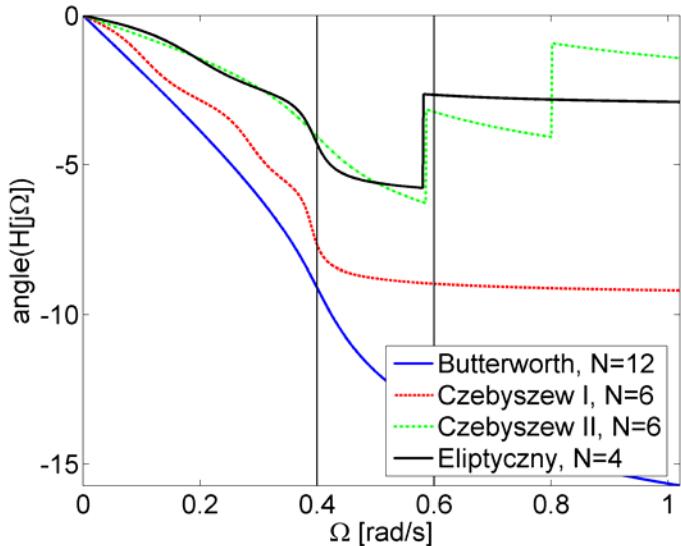
Rys.6.12 przedstawia charakterystyki fazowe filtrów, które niestety są nieliniowe. Charakterystyka opóźnienia grupowego na rys.6.12 jest najmniej pofaładowana dla filtrów Butterwortha i Czebyszewa typu II, co jest ich zaletą.



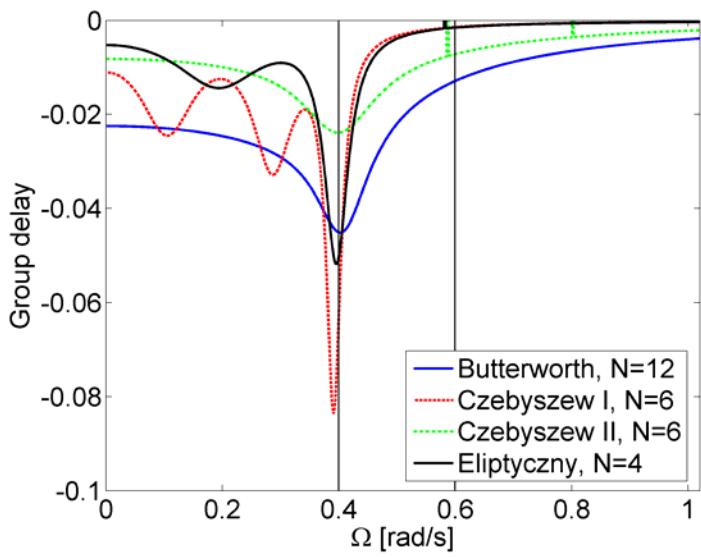
Rys.6.10 Porównanie charakterystyk amplitudowych.



Rys.6.11 Porównanie położenia zer i biegunów transmitancji  $H(s)$ .



Rys.6.12 Porównanie charakterystyk fazowych.



Rys.6.13 Porównanie opóźnienia grupowego.

## 6.5 Transformacja częstotliwości - filtry LP, HP, BP, BS

Na podstawie transmitancji unormowanych prototypów analogowych, tj. filtrów dolnoprzepustowych o częstotliwości odcięcia  $\Omega_c=1$  rad/s można otrzymać dowolne filtry pasmowe przez odpowiednie przeskalowanie argumentu transmitancji  $H(s)$ .

Filtr dolnoprzepustowy LP (*lowpass*) o dowolnej częstotliwości odcięcia  $\Omega_0$  otrzymuje się przez podstawienie:

$$s' = s / \Omega_0 . \quad (6.22)$$

Filtr górnoprzepustowy HP (*highpass*) o dowolnej częstotliwości odcięcia  $\Omega_0$  otrzymuje się przez podstawienie:

$$s' = \Omega_0 / s . \quad (6.23)$$

Filtr pasmowoprzepustowy BP (*bandpass*) o paśmie przepustowym od  $\Omega_1$  do  $\Omega_2$  otrzymuje się przez podstawienie:

$$s' = \frac{\Omega_0}{\Omega_{1,2}} \frac{(s/\Omega_0)^2 + 1}{s/\Omega_0}, \quad (6.24)$$

gdzie szerokość pasma wynosi  $\Omega_{1,2} = \Omega_2 - \Omega_1$ , a częstotliwość średnia  $\Omega_0 = \sqrt{\Omega_1 \Omega_2}$ .

Filtr pasmowozaporowy BS (*bandstop*) o paśmie zaporowym od  $\Omega_1$  do  $\Omega_2$  otrzymuje się przez podstawienie:

$$s' = \frac{\Omega_{1,2}}{\Omega_0} \frac{s/\Omega_0}{(s/\Omega_0)^2 + 1}, \quad (6.25)$$

znaczenie  $\Omega_{1,2}$  i  $\Omega_0$  jest takie jak poprzednio.

Jako przykład transformacji częstotliwości rozważmy analogowy prototyp filtra Butterwortha drugiego rzędu o transmitancji:

$$H(s) = \frac{1}{s^2 + \sqrt{2}s + 1}. \quad (6.26)$$

Przez podstawienie (6.22) możemy przesunąć częstotliwość odcięcia filtra LP z 1 rad/s do  $\Omega_0$ .

$$H(s/\Omega_0) = \frac{1}{(s/\Omega_0)^2 + \sqrt{2}(s/\Omega_0) + 1} = \frac{\Omega_0^2}{s^2 + \Omega_0 \sqrt{2}s + \Omega_0^2}. \quad (6.27)$$

Przez podstawienie (6.23) możemy uzyskać filtr HP:

$$H(\Omega_0 / s) = \frac{s^2}{s^2 + \Omega_0 \sqrt{2}s + \Omega_0^2}. \quad (6.28)$$

Przez podstawienie (6.24) możemy uzyskać filtr BP:

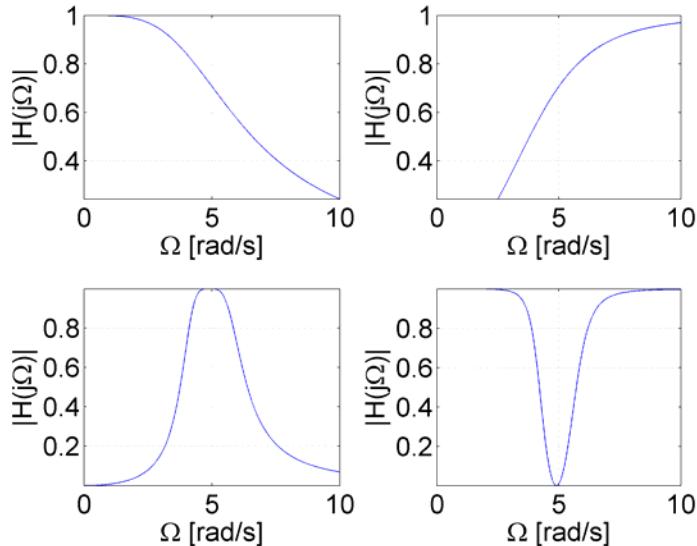
$$H\left(\frac{\Omega_0}{\Omega_{1,2}} \frac{(s/\Omega_0)^2 + 1}{s/\Omega_0}\right) = \frac{\Omega_{1,2}^2 s^2}{s^4 + \sqrt{2}\Omega_{1,2}s^3 + (2\Omega_0^2 + \Omega_{1,2}^2)s^2 + \sqrt{2}\Omega_{1,2}\Omega_0^2 s + \Omega_0^4}. \quad (6.29)$$

Przez podstawienie (6.25) możemy uzyskać filtr BS:

$$H\left(\frac{\Omega_{1,2}}{\Omega_0} \frac{s/\Omega_0}{(s/\Omega_0)^2 + 1}\right) = \frac{(s^2 + \Omega_0^2)^2}{s^4 + \sqrt{2}\Omega_{1,2}s^3 + (2\Omega_0^2 + \Omega_{1,2}^2)s^2 + \sqrt{2}\Omega_{1,2}\Omega_0^2 s + \Omega_0^4}. \quad (6.30)$$

Prototyp analogowy (6.26) jest drugiego rzędu i nie ma zer transmitancji. Filtr HP jest również drugiego rzędu, posiada jednak podwójne zero transmitancji. Filtry BP i BS są dwukrotnie większego rzędu niż prototyp.

Rys.6.14 przedstawia charakterystyki amplitudowe filtrów LP, HP, BP i BS uzyskane z prototypu unormowanego (6.26) dla  $\Omega_0=5$  rad/s (LP i HP) oraz dla  $\Omega_1=4$  rad/s i  $\Omega_2=6$  rad/s (BP i BS).



Rys.6.14 Charakterystyki filtrów LP, HP, BP i BS uzyskane z prototypu unormowanego (6.26).

Transformację częstotliwości można implementować dla transmitancji wyższych rzędów w sposób opisany poniżej. Jako przykład rozważmy podstawienie (6.24), czyli transformację transmitancji LP w BP. W implementacji wykorzystamy fakt, że transmitancja prototypów analogowych jest w postaci ilorazu wielomianów, oraz że stopień licznika transmitancji jest mniejszy od stopnia mianownika, dlatego w postaci iloczynowej transmitancję filtra można przedstawić za pomocą dwóch ułamków prostych tj.:  $H(s) = A/(s - B)$  i  $H(s) = (s - A)/(s - B)$ . Stosując do tych ułamków podstawienie (6.24) otrzymujemy:

$$H(s) = \frac{A}{s - B} \xrightarrow{(6.24)} H(s) = \frac{-A\Omega_{1,2}s}{s^2 - B\Omega_{1,2}s + \Omega_0^2}, \quad (6.31)$$

$$H(s) = \frac{s - A}{s - B} \xrightarrow{(6.24)} H(s) = \frac{s^2 - A\Omega_{1,2}s + \Omega_0^2}{s^2 - B\Omega_{1,2}s + \Omega_0^2}. \quad (6.32)$$

Jak widać z transmitancji stopnia pierwszego otrzymujemy transmitancję stopnia drugiego. Wielomiany drugiego stopnia możemy przedstawić w postaci iloczynowej przez wyliczenie pierwiastków, np. dla mianownika (6.31) mamy:  $\Delta = B\Omega_{1,2}^2 - 4\Omega_0^2$ ,  $s_{1,2} = (B\Omega_{1,2} \pm \sqrt{\Delta})/2$ . Transmitancja filtra BP w postaci iloczynowej składa się z zer i biegunów transmitancji w postaci (6.31) i (6.32).



## 7. Projektowanie filtrów cyfrowych

Filtry cyfrowe dzielą się na filtry o skończonej odpowiedzi impulsowej (nierekursywne) oznaczane jako FIR (*finite impulse response*) i o nieskończonej odpowiedzi impulsowej (rekursywne) oznaczane jako IIR (*infinite impulse response*).

Zaletą filtrów IIR są strome charakterystyki amplitudowe. Filtry te są projektowane najczęściej na podstawie prototypów analogowych z zastosowaniem transformacji biliniowej.

Filtry FIR mogą mieć liniową charakterystykę fazową, co jest ich zaletą. Są one projektowane najczęściej metodą okien lub metodami optymalizacyjnymi.

### 7.1 Transformacja biliniowa

Transformacja biliniowa służy do projektowania cyfrowych filtrów IIR na podstawie filtrów analogowych. Transformacja biliniowa jest przekształceniem algebraicznym odwzorowującym zespoloną płaszczyznę zmiennej  $s=j\Omega$  w zespoloną płaszczyznę zmiennej  $z=e^{j\omega}$ . Ponieważ  $-\infty \leq \Omega \leq \infty$ , a  $-\pi \leq \omega \leq \pi$  więc przekształcenie to jest nielinowe. Transmitancję dyskretną  $H(z)$  uzyskuje się z transmitancji analogowej  $H_c(s)$  przez podstawienie:

$$s = \frac{2}{T} \left( \frac{1-z^{-1}}{1+z^{-1}} \right), \text{ tzn.: } H(z) = H_c \left[ \frac{2}{T} \left( \frac{1-z^{-1}}{1+z^{-1}} \right) \right], \quad (7.1)$$

gdzie  $T$  jest okresem próbkowania w sekundach. Rozwiązuając (7.1) względem  $z$  i podstawiając  $s = \sigma + j\Omega$  otrzymujemy:

$$z = \frac{1 + (T_d/2)s}{1 - (T_d/2)s} = \frac{1 + \sigma T/2 + j\Omega T/2}{1 - \sigma T/2 - j\Omega T/2}. \quad (7.2)$$

Dla  $\sigma = 0$  otrzymujemy  $z = (1 + j\Omega T/2)/(1 - j\Omega T/2)$  co oznacza, że  $|z|=1$ . Oś urojona  $j\Omega$  jest więc przekształcana w okrąg jednostkowy  $e^{j\omega} = (1 + j\Omega T/2)/(1 - j\Omega T/2)$ .

Dla  $\sigma < 0$  mamy  $|z| < 1$ , a dla  $\sigma > 0$  mamy  $|z| > 1$ , co oznacza, że lewa półpłaszczyzna zespolonej jest przekształcana we wnętrze koła jednostkowego, a prawa półpłaszczyzna w obszar poza kołem. Interpretację geometryczną transformacji biliniowej na płaszczyźnie zespolonej przedstawia rys.7.1. Jeżeli biegun transmitancji analogowej  $H_c(s)$  leży w lewej półpłaszczyźnie to biegun transmitancji  $H(z)$ , po transformacji biliniowej, leży wewnątrz okręgu jednostkowego. Stabilny układ analogowy jest więc przekształcany w stabilny układ dyskretny.

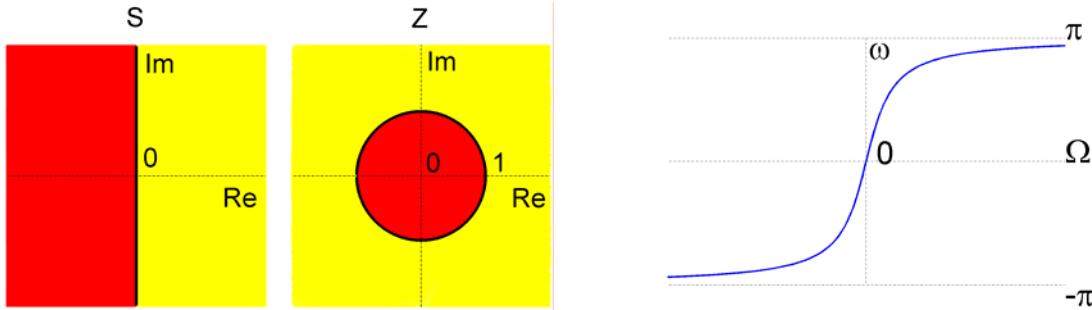
Zależność pomiędzy pulsacją sygnału analogowego  $\Omega$ , a pulsacją sygnału cyfrowego  $\omega$  wyznacza się z definicji przekształcenia:

$$\begin{aligned} j\Omega &= \frac{2}{T} \left( \frac{1-e^{-j\omega}}{1+e^{-j\omega}} \right) = \frac{2}{T} \left( \frac{1-e^{-j\omega/2}e^{-j\omega/2}}{1+e^{-j\omega/2}e^{-j\omega/2}} \right) = \frac{2}{T} \left( \frac{e^{-j\omega/2}(e^{j\omega/2}-e^{-j\omega/2})}{e^{-j\omega/2}(e^{j\omega/2}+e^{-j\omega/2})} \right) = \\ &= \frac{2}{T} \left( j \frac{\frac{e^{j\omega/2}-e^{-j\omega/2}}{2j}}{\frac{e^{j\omega/2}+e^{-j\omega/2}}{2}} \right) = \frac{j2}{T} \left( \frac{\sin(\omega/2)}{\cos(\omega/2)} \right) = j \frac{2}{T} \operatorname{tg}(\omega/2) \end{aligned} \quad (7.3)$$

ostatecznie:

$$\Omega = \frac{2}{T} \operatorname{tg}(\omega/2), \quad \omega = \arctg(\Omega T/2). \quad (7.4)$$

Zależność (7.4) jest pokazana na rys.7.1. Projekt prototypu filtra analogowego powinien uwzględnić nieliniowość transformacji biliniowej (7.4), tzn. filtr analogowy należy zaprojektować na  $\Omega$  skorygowaną wg (7.4).



Rys.7.1 Transformacja biliniowa. Odwzorowanie lewej półpłaszczyzny w okrąg jednostkowy i zależność pomiędzy częstotliwością sygnału analogowego  $\Omega$  i cyfrowego  $\omega$ .

Jako przykład rozważmy projekt dolnoprzepustowego, cyfrowego filtra eliptycznego spełniającego następujące wymagania: częstotliwość próbkowania  $F_s=100$  Hz, krawędź pasma przepustowego  $f_1=20$  Hz, krawędź pasma zaporowego  $f_2=25$  Hz, nieliniowość charakterystyki w paśmie przepustowym  $R_p=0.1$  dB, nieliniowość charakterystyki w paśmie zaporowym  $R_s=60$  dB.

Wyznaczenie transmitancji powyższego filtra cyfrowego przebiega w następującej kolejności:

1. Unormowanie częstotliwości filtra cyfrowego  $f_1$  i  $f_2$  podanych w Hz do pulsacji  $\omega$  w radianach:

$$\omega_1 = 2\pi \frac{f_1}{F_s} = 2\pi \frac{20}{100} = 1.2566 \text{ [rad]}, \quad \omega_2 = 2\pi \frac{f_2}{F_s} = 2\pi \frac{25}{100} = 1.5708 \text{ [rad]}. \quad (7.5)$$

2. Obliczenie krawędzi pasma  $\Omega$  rad/s filtra analogowego, uwzględniające nieliniowość transformacji biliniowej:

$$\Omega_1 = \frac{2}{T} \tan(\omega_1/2) = \frac{2}{1/100} \tan(1.2566/2) = 145.3085 \text{ [rad/s]}, \quad \Omega_2 = 200 \text{ [rad/s]}. \quad (7.6)$$

Rys.7.2 ilustruje konieczność uwzględnienia nieliniowości transformacji biliniowej w projekcie filtra cyfrowego. Jak widać z rys.7.2, filtr cyfrowy powinien mieć krawędzie pasmowe  $f_1=20$  Hz i  $f_2=25$  Hz, jednak prototyp analogowy jest projektowany na  $F_1=23.1266$  Hz i  $F_2=31.831$  Hz.

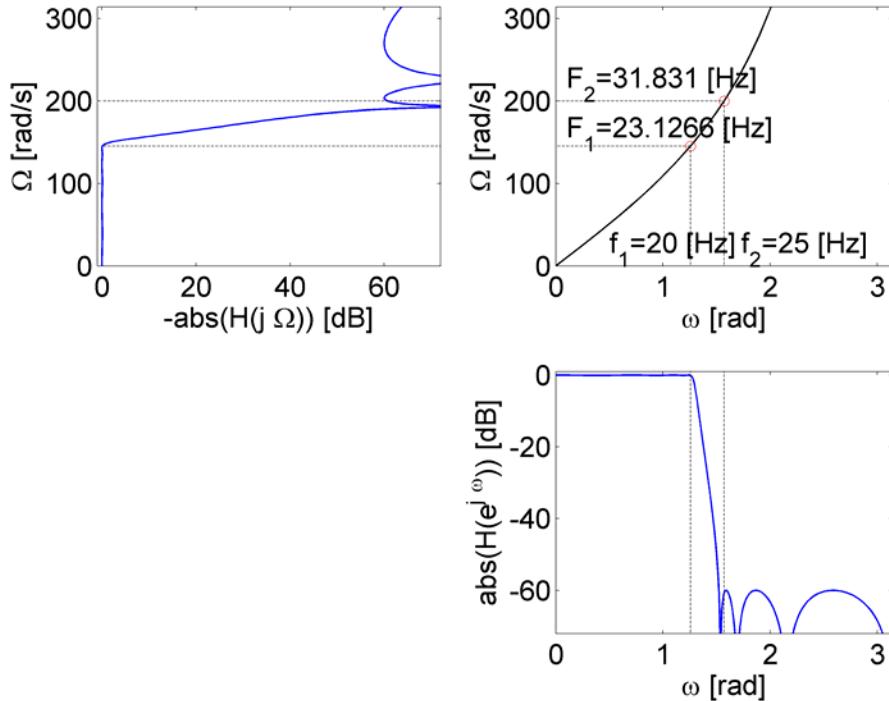
3. Określenie parametrów filtra analogowego na podstawie podanych wymagań odnośnie pasma oraz wyznaczenie jego transmitancji  $H(s)$  w postaci zer i biegunków.

Dla zadanych wymagań rzad analogowego filtra eliptycznego wynosi  $N=7$ , rozkład jego zer i biegunków przedstawia rys.7.3.

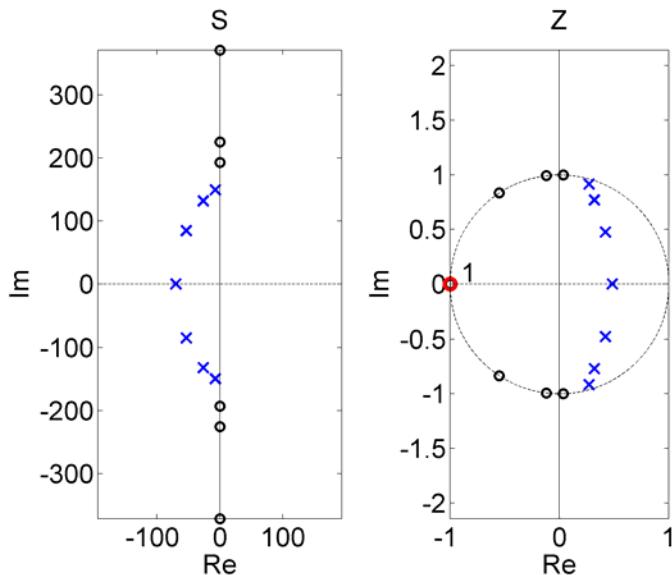
4. Transformacja biliniowa zer i biegunków filtra analogowego  $H(s)$  zgodnie ze wzorem (7.1). Otrzymujemy zera i biegunki filtra cyfrowego  $H(z)$ .

Rys.7.3 przedstawia położenie zer i biegunków transmitancji  $H(s)$  analogowego filtra eliptycznego, oraz położenie zer i biegunków transmitancji  $H(z)$  filtra cyfrowego po zastosowaniu transformacji biliniowej. Biegunki transmitancji  $H(s)$  z lewej półpłaszczyzny

zostały przetransformowane na bieguny transmitancji  $H(z)$  znajdujące się wewnątrz koła jednostkowego. Zera  $H(s)$ , które leżały na osi urojonej zostały przetransformowane na zera  $H(z)$  położone na okręgu jednostkowym.



Rys.7.2 Dobór krawędzi pasm filtra analogowego, uwzględniający nieliniowość transformacji biliniowej.

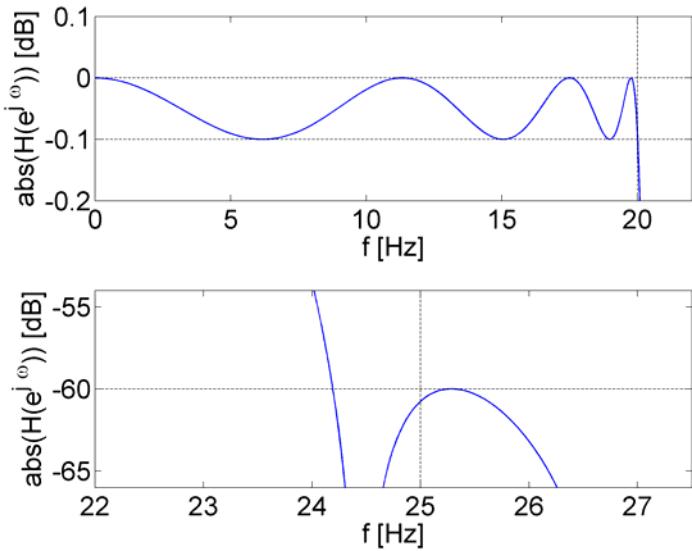


Rys.7.3 Położenie zer i biegunów transmitancji  $H(s)$  analogowego filtra eliptycznego oraz położenie zer i biegunów transmitancji  $H(z)$  filtra cyfrowego po zastosowaniu transformacji biliniowej.

##### 5. Weryfikacja otrzymanych charakterystyk częstotliwościowych.

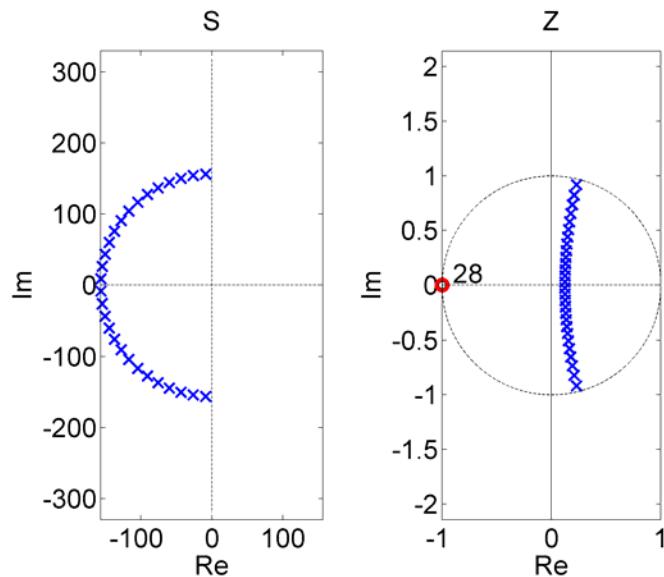
W celu określenia krawędzi pasma przepustowego i zaporowego zaprojektowanego filtra cyfrowego wyznacza się charakterystykę amplitudową w odpowiedniej skali, tak jak to pokazano na rys.7.4. Jak widać z rys.7.4 zaprojektowany, cyfrowy filtr eliptyczny spełnia

dokładnie wymagania w paśmie przepustowym i z małym nadmiarem w paśmie zaporowym, ponieważ tłumienie 60dB występuje już dla 24.2 Hz.



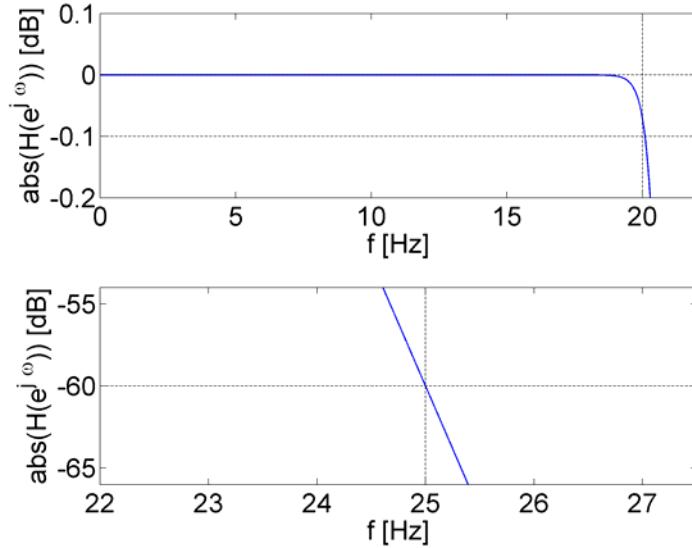
Rys.7.4 Weryfikacja charakterystyki amplitudowej filtra cyfrowego.

Transformacja biliniowa odwzorowuje całą osią  $j\Omega$  płaszczyzny  $S$  w okrąg jednostkowy na płaszczyźnie  $Z$ , a punkty  $\pm j\infty$  płaszczyzny  $S$  w -1 na płaszczyźnie  $Z$ , dlatego jeżeli transmitancja filtra analogowego  $H(s)$  ma wartość zero w  $j\Omega=\infty$ , to zera te przechodzą w zera transmitancji cyfrowej  $H(z)$  w -1. Należy zwrócić uwagę, że zera w  $j\Omega=\infty$  są generowane przez mianownik transmitancji wtedy, gdy jego stopień jest większy od stopnia licznika, np. transmitancja  $H(s)=1/(s^2+1.41s+1)$  ma podwójne zero w  $j\Omega=\infty$  i po zastosowaniu transformacji biliniowej transmitancja  $H(z)$  będzie miała podwójne zero w -1 (w liczniku). Cyfrowy filtr eliptyczny na rys.7.3 ma jedno zero w -1 ponieważ stopień licznika jego transmitancji  $H(s)$  jest o jeden mniejszy od stopnia mianownika. Dla porównania, dolnoprzepustowy, analogowy filtr Butterwortha nie ma zmiennej  $s$  w liczniku transmitancji  $H(s)$ , dlatego dolnoprzepustowy, cyfrowy filtr Butterwortha ma wielokrotne zero w -1.



Rys.7.5 Położenie zer i biegunów transmitancji  $H(s)$  analogowego filtra Butterwortha, oraz położenie zer i biegunów transmitancji  $H(z)$  filtra cyfrowego po zastosowaniu transformacji biliniowej.

Rys.7.5 przedstawia położenie zer i biegunów transmitancji  $H(s)$  analogowego filtra Butterwortha, oraz położenie zer i biegunów transmitancji  $H(z)$  filtra cyfrowego po zastosowaniu transformacji biliniowej. Filtr cyfrowy ma 28-krotne zero transmitancji  $H(z)$  w  $-1$ , ponieważ stopień mianownika jest o 28 większy od stopnia licznika. Filtr ten został zaprojektowany na te same wymagania, co filtr eliptyczny z rys.7.3, a jego charakterystykę amplitudową przedstawia rys.7.6.



Rys.7.6 Charakterystyka amplitudowa cyfrowego filtra Butterwortha.

Implementację transformacji biliniowej można wykonać analogicznie do implementacji transformacji częstotliwości. Ponownie wykorzystamy fakt, że transmitancja filtrów analogowych jest w postaci ilorazu wielomianów, oraz że stopień licznika transmitancji jest mniejszy od stopnia mianownika, dlatego w postaci iloczynowej transmitancję filtra analogowego można przedstawić za pomocą dwóch ułamków prostych tj.:  $H(s) = A/(s-B)$  i  $H(s) = (s-A)/(s-B)$ . Stosując do tych ułamków podstawienie (7.1) otrzymujemy (analogicznie do 6.31 i 6.32):

$$H(s) = \frac{A}{s-B} \quad \Rightarrow \quad H(z) = \frac{-AT(z+1)}{z(BT-2)+BT+2} = \frac{-AT}{BT-2} \frac{z+1}{z + \frac{BT+2}{BT-2}}, \quad (7.7)$$

$$H(s) = \frac{s-A}{s-B} \quad \Rightarrow \quad H(z) = \frac{z(AT-2)+AT+2}{z(BT-2)+BT+2} = \frac{AT-2}{BT-2} \frac{z + \frac{AT+2}{AT-2}}{z + \frac{BT+2}{BT-2}}. \quad (7.8)$$

Jak widać z (7.7),  $H(s)=0$  w  $j\Omega=\infty$ , generowane przez mianownik, przechodzi w zero licznika  $H(z)$  w  $e^{j\omega}=-1$ .

## 7.2 Filtry FIR

Charakterystyka amplitudowa idealnego filtra dolnoprzepustowego, przedstawiona na rys.7.7, jest następująca:

$$H_{LP}(e^{j\omega}) = \begin{cases} 1, & |\omega| < \omega_c \\ 0, & \omega_c < |\omega| \leq \pi \end{cases}. \quad (7.9)$$

Odpowiedź impulsową filtra LP można wyznaczyć za pomocą odwrotnej transformacji Fouriera:

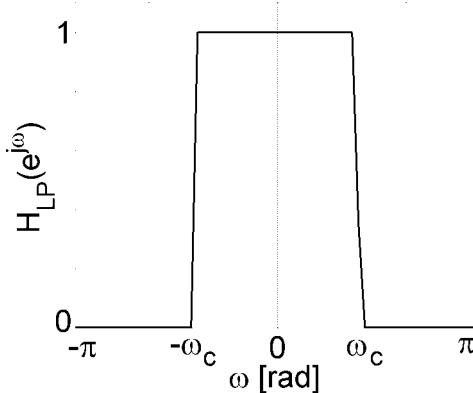
$$h_{LP}[n] = \frac{1}{2\pi} \int_{-\omega_c}^{\omega_c} e^{j\omega n} d\omega = \frac{1}{2\pi j n} [e^{j\omega n}]_{-\omega_c}^{\omega_c} = \frac{1}{2\pi j n} (e^{j\omega_c n} - e^{-j\omega_c n}) = \frac{\sin(\omega_c n)}{\pi n} = \begin{cases} \frac{\sin(\omega_c n)}{\pi n}, & n \neq 0 \\ \frac{\omega_c}{\pi}, & n = 0 \end{cases}. \quad (7.10)$$

Odpowiedź impulsowa (7.10) jest nieprzyczynowa, a zakres zmian indeksu  $n$  w (7.10) wynosi  $-\infty < n < \infty$ . W praktycznych realizacjach filtrów uwzględnia się tylko fragment odpowiedzi impulsowej (7.10) o długości  $M+1$  współczynników. Dodatkowo w celu uzyskania układów przyczynowych odpowiedź impulsowa filtra jest przesuwana o połowę jego długości:

$$H_{LP}(e^{j\omega}) = \begin{cases} e^{-j\omega M/2}, & |\omega| < \omega_c \\ 0, & \omega_c < |\omega| \leq \pi \end{cases},$$

$$h_{LP}[n] = \frac{1}{2\pi} \int_{-\omega_c}^{\omega_c} e^{-j\omega M/2} e^{j\omega n} d\omega = \frac{\sin(\omega_c(n - M/2))}{\pi(n - M/2)}, \quad 0 \leq n \leq M. \quad (7.11)$$

Odpowiedzi impulsowe (7.10) i (7.11) przedstawione są na rys.7.8. Ich kształty są takie same, różnią się one tylko indeksowaniem wyrazów, tzn. są względem siebie przesunięte o połowę długości filtru. Długość filtra przyczynowego na rys.7.8 wynosi  $M+1=31$  współczynników i jest ona przesunięta o  $M/2=15$  indeksów.



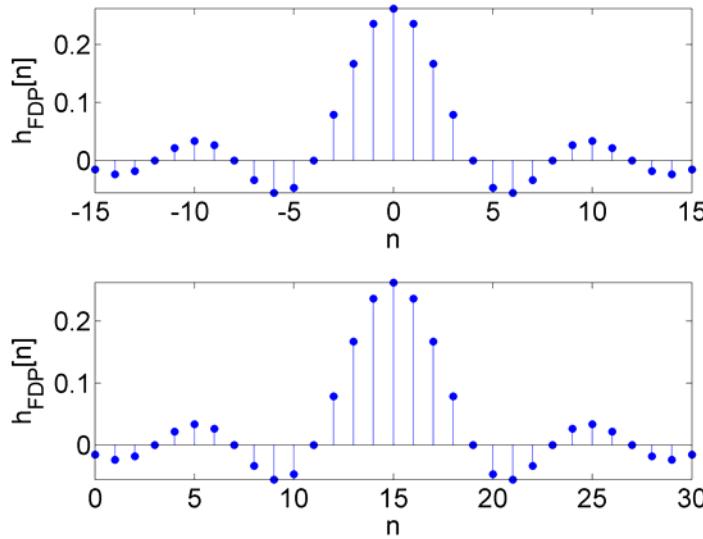
Rys.7.7 Charakterystyka amplitudowa idealnego filtra LP.

Rys.7.9 przedstawia charakterystyki amplitudowe dla różnych długości filtrów dolnoprzepustowych. Wraz ze wzrostem długości odpowiedzi impulsowej filtra charakterystyka amplitudowa coraz bardziej zbliża się do idealnego przebiegu prostokątnego, jednak amplituda oscylacji w punktach nieciągłości idealnej charakterystyki amplitudowej nie zmienia się. Odpowiedź impulsowa filtra LP nie jest bezwzględnie sumowalna, ponieważ nie jest spełniony warunek:

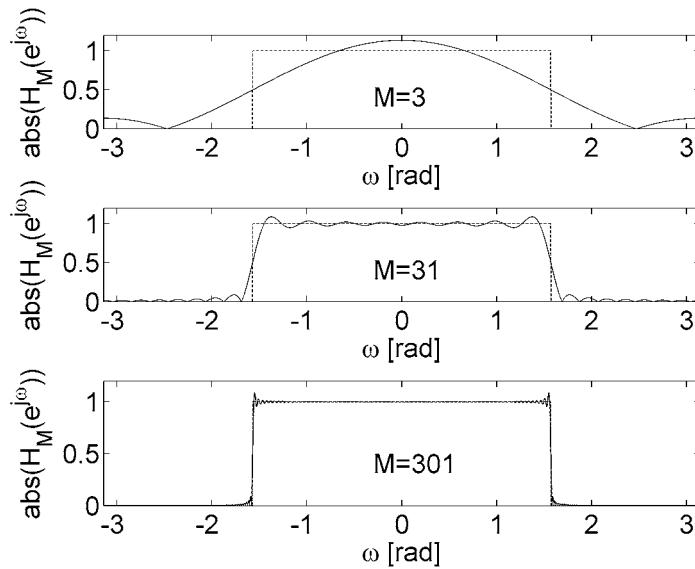
$$\lim_{M \rightarrow \infty} |H(e^{j\omega}) - H_M(e^{j\omega})| = 0, \quad (7.12)$$

tzn. wraz ze wzrostem długości filtra nie maleje błąd dopasowania do charakterystyki idealnej równocześnie dla wszystkich częstotliwości. Jest natomiast sumowalna w kwadracie, ponieważ maleje całkowita energia tego błędu:

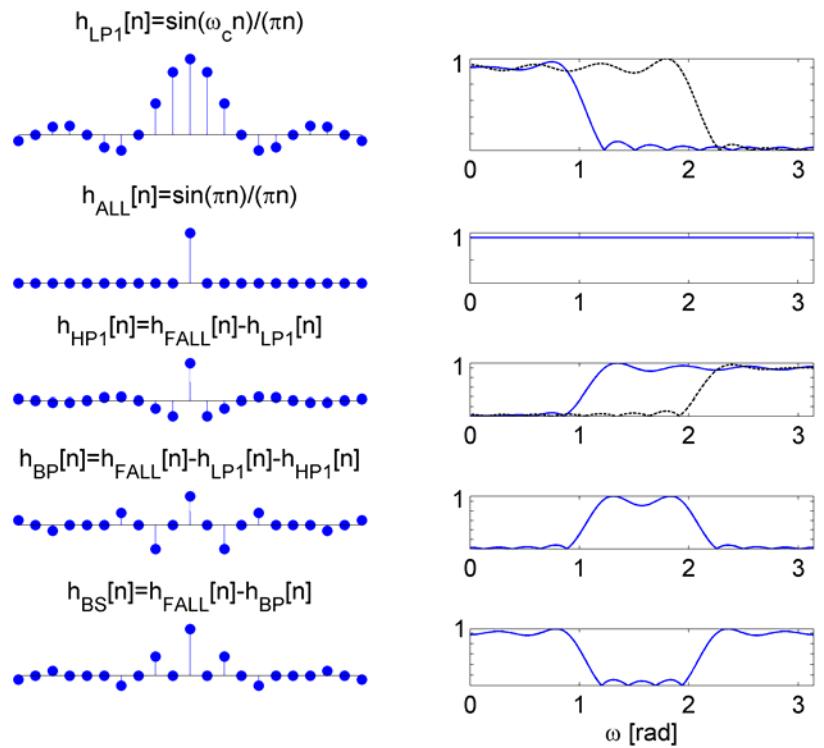
$$\lim_{M \rightarrow \infty} \int_{-\pi}^{\pi} |H(e^{j\omega}) - H_M(e^{j\omega})|^2 d\omega = 0. \quad (7.13)$$



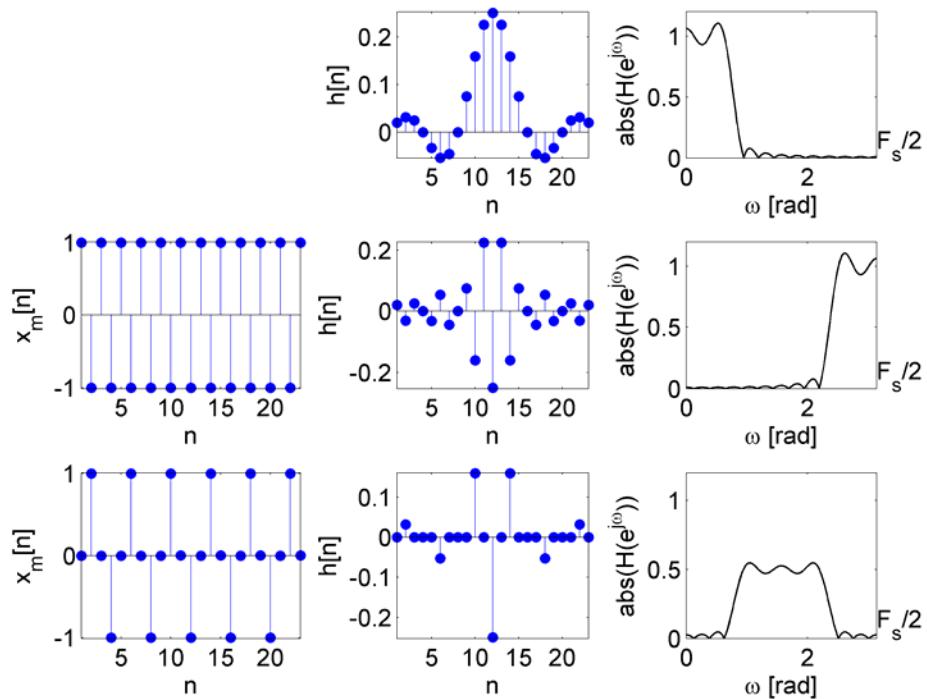
Rys.7.8 Fragment nieprzyczynowej, nieskończonej odpowiedzi impulsowej filtra LP (7.10) i przyczynowa, skończona odpowiedź impulsowa filtra LP (7.11).



Rys.7.9 Charakterystyki amplitudowe dla różnych długości filtrów LP.



Rys.7.10 Filtry HP, BP, BS jako sumy filtrów LP i filtrów wszechprzepustowego.



Rys.7.11 Zastosowanie twierdzenia o modulacji do przesunięcia charakterystyki filtru LP o  $F_s/2$  i  $F_s/4$ .

Odpowiedzi impulsowe pozostałych filtrów pasmowych, tj. HP, BP i BS można wyprowadzić analogicznie do (7.10) z zadanych charakterystyk amplitudowych. Filtry HP, BP, BS można też wyznaczyć jako sumy filtru LP i filtru wszechprzepustowego, co przedstawia rys.7.10. Jako przykład rozważmy filtr HP o częstotliwości granicznej  $\omega_c$ , jego odpowiedź impulsowa wynosi (wg rys.7.10):

$$h_{HP}[n] = h_{ALL}[n] - h_{LP}[n] = \frac{\sin(\pi n)}{\pi n} - \frac{\sin(\omega_c n)}{\pi n} = \begin{cases} \frac{\sin(\pi n) - \sin(\omega_c n)}{\pi n}, & n \neq 0 \\ 1 - \frac{\omega_c}{\pi}, & n = 0 \end{cases}. \quad (7.14)$$

Wydajną obliczeniowo procedurą zamiany filtra LP na filtr HP lub BP jest zastosowanie ciągów modulacyjnych pokazanych na rys.7.11. Ciągi modulacyjne na rys.7.11 zawierają tylko wartości -1, 0 i 1, więc mnożenie odpowiedzi impulsowej filtra LP przez te ciągi nie wymaga żadnych obliczeń. Rys.7.11 przedstawia odpowiedź impulsową filtra LP oraz jego charakterystykę amplitudową. Odpowiedzi impulsowe filtrów HP i BP zostały otrzymane przez wymnożenie odpowiedzi impulsowej LP przez ciąg modulacyjny. Jak widać z rys.7.11, podane ciągi modulacyjne przesuwają charakterystykę filtra LP o  $F_s/2$  i  $F_s/4$ . W przypadku filtra BP wynik filtracji powinien być przemnożony przez dwa.

### 7.2.1 Układy LTI z liniową charakterystyką fazową

Dzięki liniowej charakterystyce fazowej w paśmie przepustowym filtr zachowuje kształt sygnału (wszystkie częstotliwości składowe sygnału są tak samo opóźniane). Ogólnie odpowiedź częstotliwościowa układów z liniową fazą jest następująca:

$$H(e^{j\omega}) = |H(e^{j\omega})| e^{j(\beta - \omega\alpha)} = |H(e^{j\omega})| \cos(\beta - \omega\alpha) + j |H(e^{j\omega})| \sin(\beta - \omega\alpha), \quad (7.15)$$

gdzie charakterystyka fazowa jest dana zależnością  $faza\{H(e^{j\omega})\} = \beta - \omega\alpha$ , a opóźnienie grupowe, czyli pochodna charakterystyki fazowej po częstotliwości ze znakiem minus, wynosi  $grd\{H(e^{j\omega})\} = \alpha$  i jest stałe. Z (7.15) otrzymujemy:

$$\operatorname{tg}(\beta - \omega\alpha) = \frac{\sin(\beta - \omega\alpha)}{\cos(\beta - \omega\alpha)}. \quad (7.16)$$

Odpowiedź częstotliwościową można również wyliczyć z definicji transformacji Fouriera:

$$H(e^{j\omega}) = \sum_{n=-\infty}^{\infty} h[n] e^{-jn\omega} = \sum_{n=-\infty}^{\infty} h[n] \cos(\omega n) - j \sum_{n=-\infty}^{\infty} h[n] \sin(\omega n), \quad (7.17)$$

skąd:

$$\operatorname{tg}(\gamma) = \frac{- \sum_{n=-\infty}^{\infty} h[n] \sin(\omega n)}{\sum_{n=-\infty}^{\infty} h[n] \cos(\omega n)}. \quad (7.18)$$

Dla układu z liniową charakterystyką fazową zachodzi równość (7.16) i (7.18):

$$\begin{aligned} \operatorname{tg}(\beta - \omega\alpha) &= \operatorname{tg}(\gamma), \\ \frac{\sin(\beta - \omega\alpha)}{\cos(\beta - \omega\alpha)} &= \frac{- \sum_{n=-\infty}^{\infty} h[n] \sin(\omega n)}{\sum_{n=-\infty}^{\infty} h[n] \cos(\omega n)}. \end{aligned} \quad (7.19)$$

Mnożąc na krzyż (7.19), grupując i stosując tożsamość  $\sin \alpha \cos \beta \pm \cos \alpha \sin \beta = \sin(\alpha \pm \beta)$ :

$$\begin{aligned} \sin(\beta - \omega\alpha) \sum_{n=-\infty}^{n=\infty} h[n] \cos(\omega n) + \cos(\beta - \omega\alpha) \sum_{n=-\infty}^{n=\infty} h[n] \sin(\omega n) &= 0, \\ \sum_{n=-\infty}^{n=\infty} h[n] \sin(\beta - \omega\alpha) \cos(\omega n) + \sum_{n=-\infty}^{n=\infty} h[n] \cos(\beta - \omega\alpha) \sin(\omega n) &= 0, \end{aligned} \quad (7.20)$$

otrzymujemy warunek konieczny na  $h[n]$ ,  $\alpha$ ,  $\beta$ , aby system miał liniową fazę:

$$\sum_{n=-\infty}^{n=\infty} h[n] \sin(\omega(n-\alpha) + \beta) = 0. \quad (7.21)$$

Warunek (7.21) może być spełniony w dwóch przypadkach:

$$\begin{aligned} \beta = 0 \text{ lub } \beta = \pi \\ 1. \quad 2\alpha = M, \quad M \in C \quad , \quad \sum_{n=-\infty}^{n=\infty} h[n] \sin(\omega(n-\alpha)) = 0, \\ h[2\alpha-n] = h[n] \end{aligned} \quad (7.22)$$

$$\begin{aligned} \beta = -\pi/2 \text{ lub } \beta = \pi/2 \\ 2. \quad 2\alpha = M, \quad M \in C \quad , \quad \sum_{n=-\infty}^{n=\infty} h[n] \cos(\omega(n-\alpha)) = 0. \\ h[2\alpha-n] = -h[n] \end{aligned} \quad (7.23)$$

Warunki (7.22) i (7.23) określają cztery możliwe rodzaje symetrii odpowiedzi impulsowej układu, które zapewniają liniową charakterystykę fazową. Są one omówione poniżej dla filtra o skończonej odpowiedzi impulsowej o długości  $M+1$ , tzn.  $h[n]=0$  dla  $n<0$  i  $n>M$ .

**Filtr FIR typ I** - symetryczna odpowiedź impulsowa -  $M$  parzyste, opóźnienie  $M/2$ :

$$h[n] = h[M-n], \quad 0 \leq n \leq M. \quad (7.24)$$

Odpowiedź częstotliwościowa:

$$\begin{aligned} H(e^{j\omega}) &= \sum_{n=0}^M h[n] e^{-j\omega n} = \sum_{n=0}^{M/2-1} h[n] e^{-j\omega n} + h[M/2] e^{-j\omega M/2} + \sum_{n=M/2+1}^M h[n] e^{-j\omega n} = \\ &= \sum_{n=0}^{M/2-1} h[n] e^{-j\omega n} + h[M/2] e^{-j\omega M/2} + \sum_{n=0}^{M/2-1} h[n] e^{-j\omega(M-n)} = \sum_{n=0}^{M/2-1} h[n] (e^{-j\omega n} + e^{j\omega n} e^{-j\omega M}) + h[M/2] e^{-j\omega M/2} = \\ &= \sum_{n=0}^{M/2-1} h[n] e^{-j\omega M/2} (e^{-j\omega n} e^{j\omega M/2} + e^{j\omega n} e^{-j\omega M/2}) + h[M/2] e^{-j\omega M/2} = \\ &= 2e^{-j\omega M/2} \sum_{n=0}^{M/2-1} h[n] \frac{e^{-j[\omega(n-M/2)]} + e^{j[\omega(n-\omega M/2)]}}{2} + h[M/2] e^{-j\omega M/2} = \\ &= e^{-j\omega M/2} \left( 2 \sum_{n=0}^{M/2-1} h[n] \cos(\omega(n-M/2)) + h[M/2] \right) = e^{-j\omega M/2} \left( 2 \sum_{n=0}^{M/2} h[n] \cos(\omega(n-M/2)) \right). \end{aligned}$$

Wyrażenie w nawiasie ma wartość rzeczywistą (a nie zespoloną) i nie wpływa na charakterystykę fazową układu.

**Filtr FIR typ II** - symetryczna odpowiedź impulsowa -  $M$  nieparzyste, opóźnienie  $M/2$ :

$$h[n] = h[M-n], \quad 0 \leq n \leq M. \quad (7.25)$$

Odpowiedź częstotliwościowa:

$$\begin{aligned}
H(e^{j\omega}) &= \sum_{n=0}^M h[n]e^{-j\omega n} = \sum_{n=0}^{(M-1)/2} h[n]e^{-j\omega n} + \sum_{n=(M+1)/2}^M h[n]e^{-j\omega n} = \\
&= \sum_{n=0}^{(M-1)/2} h[n]e^{-j\omega n} + \sum_{n=0}^{(M-1)/2} h[n]e^{-j\omega(M-n)} = = \sum_{n=0}^{(M-1)/2} h[n] \left( e^{-j\omega n} + e^{j\omega n} e^{-j\omega M} \right) = \\
&= \sum_{n=0}^{(M-1)/2} h[n] 2e^{-j\omega M/2} \left( \frac{e^{-j\omega n} e^{j\omega M/2} + e^{j\omega n} e^{-j\omega M/2}}{2} \right) = = e^{-j\omega M/2} \sum_{n=0}^{(M-1)/2} 2h[n] \left( \frac{e^{-j[\omega(n-M/2)]} + e^{j[\omega(n-M/2)]}}{2} \right) = \\
&= e^{-j\omega M/2} \left( 2 \sum_{n=0}^{(M-1)/2} h[n] \cos(\omega(n-M/2)) \right).
\end{aligned}$$

**Filtr FIR typ III** - antysymetryczna odpowiedź impulsowa -  $M$  parzyste, opóźnienie  $M/2$ :

$$h[n] = -h[M-n], \quad 0 \leq n \leq M. \quad (7.26)$$

Odpowiedź częstotliwościowa:

$$\begin{aligned}
H(e^{j\omega}) &= \sum_{n=0}^M h[n]e^{-j\omega n} = \sum_{n=0}^{M/2-1} h[n]e^{-j\omega n} + h[M/2]e^{-j\omega M/2} + \sum_{n=M/2+1}^M h[n]e^{-j\omega n} = \\
&= \sum_{n=0}^{M/2-1} h[n]e^{-j\omega n} + h[M/2]e^{-j\omega M/2} - \sum_{n=0}^{M/2-1} h[n]e^{-j\omega(M-n)} = \\
&= 2je^{-j\omega M/2} \sum_{n=0}^{M/2-1} h[n] \frac{e^{-j[\omega(n-M/2)]} - e^{j[\omega(n-\omega M/2)]}}{2j} + h[M/2]e^{-j\omega M/2} = \\
&= je^{-j\omega M/2} \left( 2 \sum_{n=0}^{M/2-1} h[n] \sin(\omega(n-M/2)) + h[M/2] \right) = e^{-j\omega M/2+\pi} \left( 2 \sum_{n=0}^{M/2} h[n] \sin(\omega(n-M/2)) \right).
\end{aligned}$$

**Filtr FIR typ IV** - antysymetryczna odpowiedź impulsowa -  $M$  nieparzyste, opóźnienie  $M/2$ :

$$h[n] = -h[M-n], \quad 0 \leq n \leq M. \quad (7.27)$$

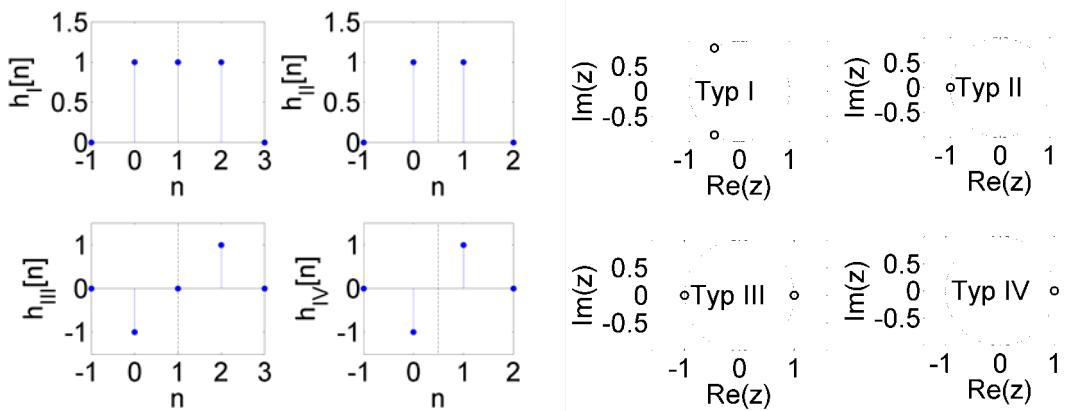
Odpowiedź częstotliwościowa:

$$\begin{aligned}
H(e^{j\omega}) &= \sum_{n=0}^M h[n]e^{-j\omega n} = \sum_{n=0}^{(M-1)/2} h[n]e^{-j\omega n} + \sum_{n=(M+1)/2}^M h[n]e^{-j\omega n} = \\
&= \sum_{n=0}^{M/2-1} h[n]e^{-j\omega n} - \sum_{n=0}^{M/2-1} h[n]e^{-j\omega(M-n)} = 2je^{-j\omega M/2} \sum_{n=0}^{M/2-1} h[n] \frac{e^{-j[\omega(n-M/2)]} - e^{j[\omega(n-\omega M/2)]}}{2j} = \\
&= e^{-j\omega M/2+\pi} \left( 2 \sum_{n=0}^{M/2-1} h[n] \sin(\omega(n-M/2)) \right).
\end{aligned}$$

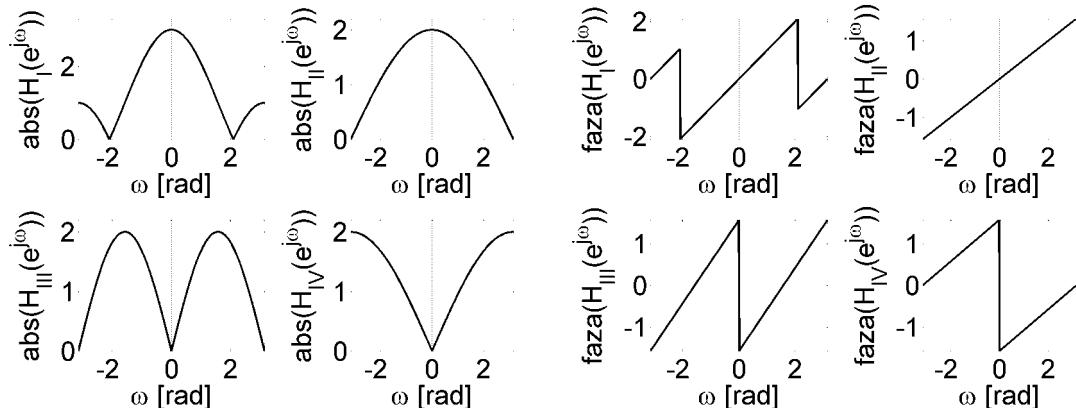
Opisane powyżej symetrie odpowiedzi impulsowych są zilustrowane poniższymi przykładami:

$$\begin{aligned}
h_I[n] &= [1 \quad 1 \quad 1], \quad H_I(z) = 1 + z^{-1} + z^{-2}, \quad H_I(z) = (z - e^{j2.09})(z - e^{-j2.09}) \quad - \text{wszystkie typy}, \\
h_{II}[n] &= [1 \quad 1], \quad H_{II}(z) = 1 + z^{-1}, \quad H_{II}(z) = z + 1 \quad \text{zero dla } z=-1 \quad - \text{HP}, \\
h_{III}[n] &= [-1 \quad 0 \quad 1], \quad H_{III}(z) = -1 + z^{-2}, \quad H_{III}(z) = -(z-1)(z+1) \quad \text{zero dla } z=\pm 1 \quad \text{tylko BP}, \\
h_{IV}[n] &= [-1 \quad 1], \quad H_{IV}(z) = -1 + z^{-1}, \quad H_{IV}(z) = -(z-1) \quad \text{zero dla } z=1 \quad - \text{LP}.
\end{aligned}$$

Symetria typu I umożliwia realizację wszystkich typów filtrów pasmowych. Pozostałe symetrie odpowiedzi impulsowej posiadają zero transmitancji  $H(z)$  w -1 lub/i w 1, co ogranicza ich zastosowanie do niektórych typów filtrów pasmowych. Rys.7.12 przedstawia symetrie odpowiedzi impulsowych układów z liniową fazą oraz położenie zer transmitancji tych układów, a rys.7.13 przedstawia ich charakterystyki amplitudowe. Położenie zer transmitancji dla filtra  $h_I$  w  $\omega = \pm 2.09$  rad wynika z doboru jego współczynników. W przypadku pozostałych filtrów zera w -1 i 1 są następstwem symetrii odpowiedzi impulsowej i występują zawsze niezależnie od wartości współczynników.



Rys.7.12 Symetrie odpowiedzi impulsowych układów z liniową fazą oraz położenie zer transmitancji tych układów.



Rys.7.13 Charakterystyki amplitudowe i fazowe dla różnych symetrii odpowiedzi impulsowej.

## 7.2.2 Projektowanie filtrów FIR metodą okien

W metodzie okien zadaje się wymaganą charakterystykę częstotliwościową filtra:

$$H_d(e^{j\omega}) = \sum_{n=-\infty}^{\infty} h_d[n] e^{-j\omega n}, \quad (7.28)$$

na jej podstawie, za pomocą odwrotnej transformacji Fouriera, wyznacza się nieskończoną odpowiedź impulsową filtra:

$$h_d[n] = \frac{1}{2\pi} \int_{-\pi}^{\pi} H_d(e^{j\omega}) e^{j\omega n} d\omega, \quad (7.29)$$

a następnie mnoży się tę odpowiedź impulsową przez wybrane okno o skończonej długości (np. okno prostokątne):

$$h[n] = h_d[n]w[n], \quad w[n] = \begin{cases} 1, & 0 \leq n \leq M, \\ 0, & n < 0 \text{ i } n > M. \end{cases} \quad (7.30)$$

Widmo iloczynu sygnałów (7.30) jest splotem widm tych sygnałów. Odpowiedź częstotliwościowa filtra  $H(e^{j\omega})$  jest więc splotem widma filtra idealnego z widmem okna:

$$H(e^{j\omega}) = \frac{1}{2\pi} \int_{-\pi}^{\pi} H_d(e^{j\Theta})W(e^{j(\omega-\Theta)})d\Theta. \quad (7.31)$$

Rys.7.14 ilustruje zależności (7.30) i (7.31), odpowiedź impulsowa filtra o skończonej długości jest fragmentem odpowiedzi impulsowej filtra idealnego wybranym przez okno. Charakterystyka amplitudowa filtra o skończonej długości jest splotem widma charakterystyki filtra idealnego, tj. prostokąta, z widmem okna. W trakcie liczenia splotu oba te widma "przejeżdżają" przez siebie, co powoduje, że zafalowanie charakterystyki filtra o skończonej długości w paśmie przepustowym i zaporowym jest takie samo. Dla filtra FIR z oknem prostokątnym tłumienie charakterystyki w paśmie zaporowym wynosi 21 dB i nie zależy od długości filtra, co pokazano na rys.7.15. Wraz ze wzrostem długości filtra maleje jego pasmo przejściowe, tzn. charakterystyka amplitudowa coraz bardziej zbliża się do prostokąta. Filtr FIR z oknem prostokątnym jest przypadkiem skrajnym, charakteryzuje się on największą stromością w paśmie przejściowym, co jest jego zaletą i najmniejszym tłumieniem pierwszego listka bocznego, co jest jego wadą.

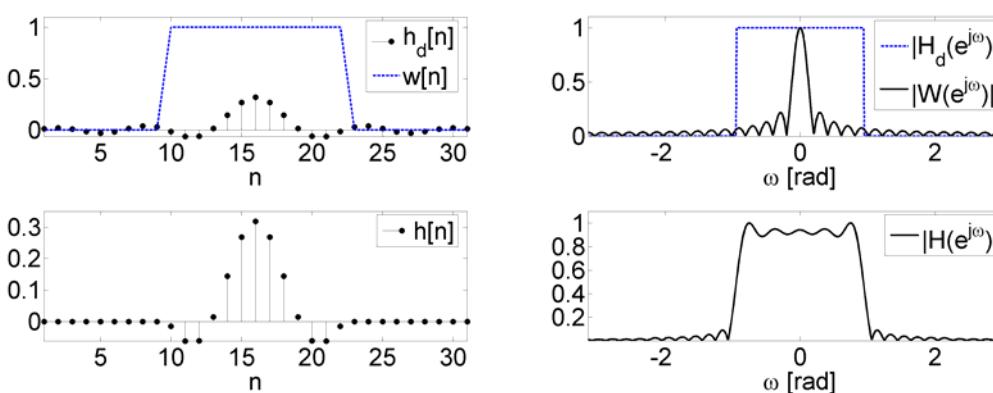
W literaturze oprócz okna prostokątnego funkcjonuje wiele innych okien czasowych. W porównaniu z oknem prostokątnym wszystkie pozostałe okna mają niżej położony pierwszy listek boczny i szersze pasmo przejściowe.

Jednym z bardziej popularnych okien jest okno Hamminga zdefiniowane zależnością:

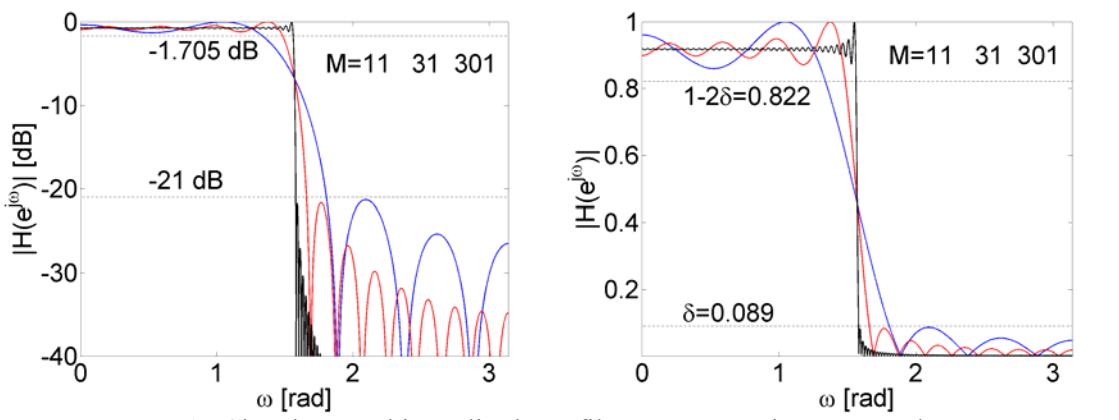
$$w[n] = 0.54 - 0.46 \cos(2\pi n/M), \quad 0 \leq n \leq M. \quad (7.32)$$

Innym przykładem okna czasowego jest okno Bartletta (trójkątne):

$$w[n] = \begin{cases} 2n/M, & 0 \leq n \leq M/2 \\ 2 - 2n/M, & M/2 < n \leq M. \end{cases} \quad (7.33)$$



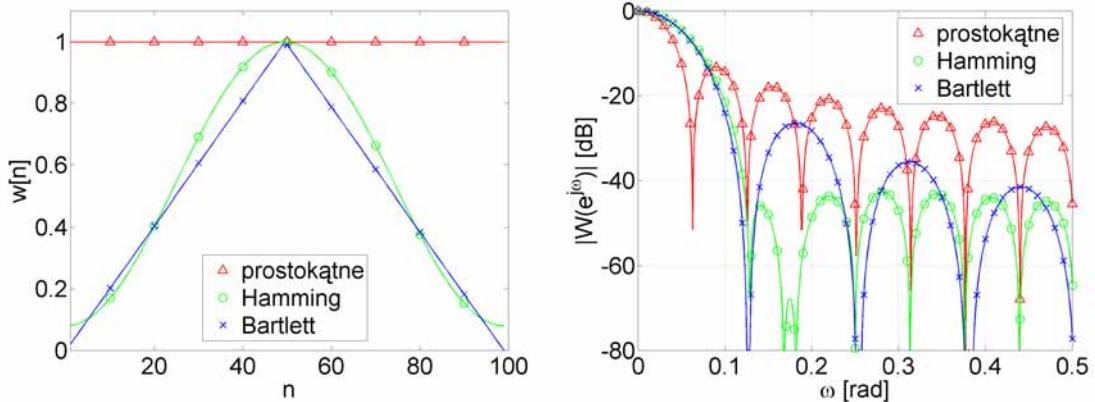
Rys.7.14 Nieskończona odpowiedź impulsowa idealnego filtra LP oraz fragment wybranej przez okno prostokątne. Widma filtra idealnego i okna prostokątnego oraz widmo filtra o skończonej odpowiedzi impulsowej.



Rys. 7.15 Charakterystyki amplitudowe filtru LP FIR z oknem prostokątnym.

Rys. 7.16 przedstawia trzy okna czasowe oraz ich widma. Listki boczne okna prostokątnego są położone na wysokości ok. -13 dB, okna Hamminga ok. -41 dB, a okna Bartletta ok. -25 dB.

Dla wszystkich okien czasowych zwiększenie długości okna  $M$  powoduje zmniejszenie szerokości listka głównego i nie wpływa na tłumienie pierwszego listka bocznego.



Rys. 7.16 Okno prostokątne, okno Hamminga i okno Bartletta oraz ich widma.

### 7.2.2.1 Okno Kaiser - okno parametryczne

Położenie listków bocznych w oknie prostokątnym czy np. Hamminga jest stałe, oznacza to, że tłumienie filtru FIR z takimi oknami jest również stałe, tj. nie może być regulowane. Okna parametryczne umożliwiają zmianę położenia listków bocznych do żądanej wartości poprzez zmianę parametru.

Parametryczne okno Kaiser zdefiniowane jest następująco:

$$w[n] = \begin{cases} \frac{I_0[\beta(1-(n-\alpha)/\alpha)^2]}{I_0(\beta)}, & 0 \leq n \leq M \\ 0 & \text{inny} \end{cases}, \quad (7.34)$$

gdzie  $\alpha=M/2$ , a  $I_0(\cdot)$  jest zmodyfikowaną funkcją Bessela zerowego rzędu pierwszego rodzaju. Okno Kaiser ma dwa parametry: 1)  $M$ -długość (liczba niezerowych współczynników); 2)  $\beta$ -parametr kształtu. Wzrost  $\beta$  powoduje obniżanie pierwszego listka bocznego (i niestety zwiększenie szerokości listka głównego, tj. pasma przejściowego). Natomiast wzrost  $M$  powoduje zawężanie listka głównego i nie wpływa na położenie pierwszego listka bocznego.

Na podstawie doświadczeń numerycznych Kaiser określił wymagania odnośnie  $M$  i  $\beta$  pozwalające spełnić zadane wymagania charakterystyki amplitudowej filtru FIR tj. tłumienie

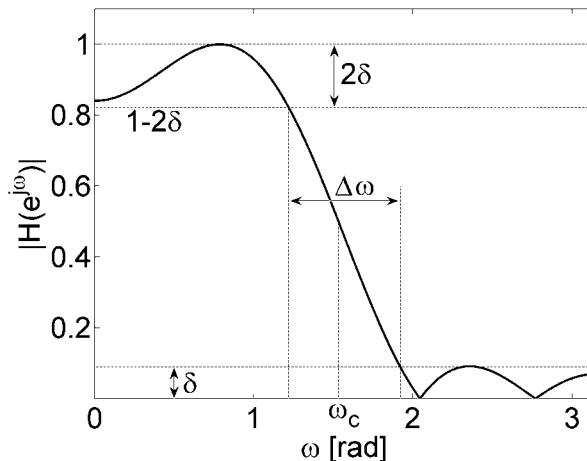
w paśmie przepustowym/zaporowym  $\delta$  i szerokość pasma przejściowego  $\Delta\omega$ , pokazane na rys.7.17. Parametr  $\beta$  wyznaczany jest z zależności:

$$\beta = \begin{cases} 0.1102(A - 8.7), & A > 50 \\ 0.5842(A - 21)^{0.4} + 0.07886(A - 21), & 21 \leq A \leq 50, \\ 0, & A < 21 \end{cases}, \quad (7.35)$$

gdzie  $A = -20 \log_{10} \delta$ ,  $\Delta\omega = \omega_s - \omega_p$ . Znaczenie  $\delta$  i  $\Delta\omega$  przedstawione jest na rys.7.17. Długość filtra z dokładnością do  $\pm 2$  wynosi:

$$M = \frac{A - 8}{2.285\Delta\omega}, \quad (7.36)$$

przy czym wynik jest zaokrąglany w góre do najbliższej liczby całkowitej.



Rys.7.17 Charakterystyka amplitudowa filtru projektowanego metodą okien.

Dla wybranych wartości  $\beta$  okno Kaiser ma kształt taki, jak niektóre okna nieparametryczne, np. dla  $\beta=0$  okno Kaiser jest oknem prostokątnym, a dla  $\beta=4.86$  okno Kaiser jest oknem Hamminga.

Jako przykład, wyznaczmy parametry okna Kaiser dla filtru LP, który ma spełniać następujące wymagania: krawędź pasma przepustowego  $\omega_p=0.4\pi$  rad, krawędź pasma zaporowego  $\omega_s=0.6\pi$  rad, maksymalna nieliniowość w paśmie przepustowym  $r_p=1$  dB, minimalne tłumienie w paśmie zaporowym  $r_s=60$  dB. Projekt powyższego filtru przebiega w następujących etapach:

1. Obliczenie  $\delta$  dla obu pasm:

$$\text{pasmo przepustowe } \delta_p = (1 - 10^{-r_p/20})/2 = 0.0544,$$

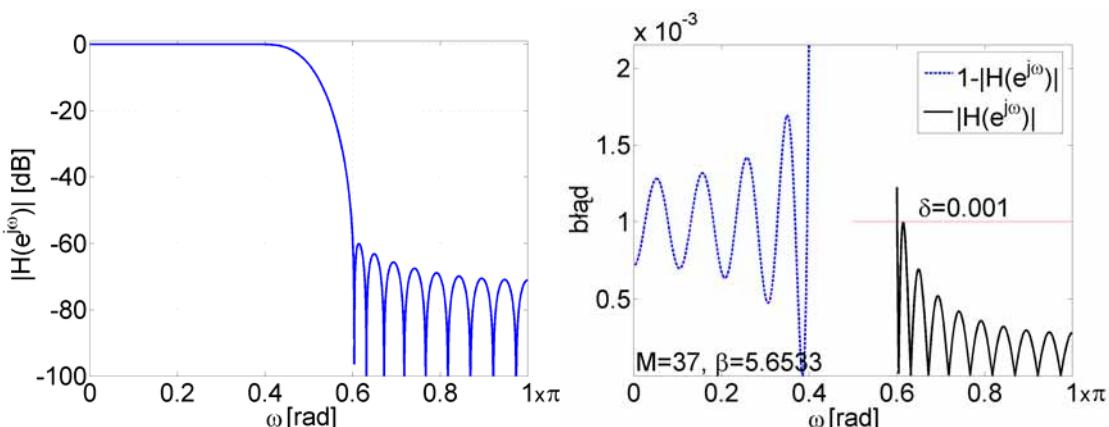
$$\text{pasmo zaporowe } \delta_s = 10^{-r_s/20} = 0.001.$$

Ponieważ dla FIR projektowanego metodą okien  $\delta_p=\delta_s$ , więc wybiera się wymaganie ostrzejsze:  $\delta = \min\{\delta_p, \delta_s\} = 0.001$ .

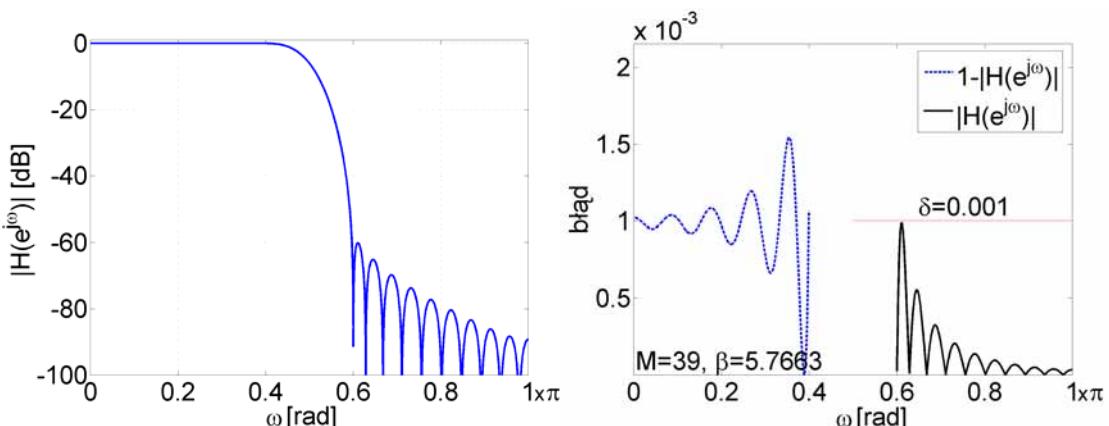
2. Korzystając ze wzorów projektowych (7.35) i (7.36) wyznacza się  $\beta = 5.6533$  i  $M = 37$ , a następnie okno o tych parametrach (7.34).

3. Odpowiedź impulsową FIR wylicza się dla  $\omega_c = (\omega_p + \omega_s)/2$  wg (7.10), a następnie wymnaża się ją przez współczynniki okna.

4. Ostatnim etapem projektu jest weryfikacja otrzymanych charakterystyk filtra i ewentualna ich korekcja (przez zmianę parametrów okna metodą prób i błędów). Rys. 7.18 przedstawia charakterystykę amplitudową i błąd aproksymacji dla filtra z oknem Kaisera bezpośrednio po zastosowaniu wzorów projektowych, tj. dla  $\beta=5.6333$  i  $M=37$ . Jak widać zaprojektowany filtr nieznacznie nie spełnia zadanych wymagań, ponieważ tłumienie w paśmie zaporowym (ostrzejsze wymaganie) jest zbyt małe. Rys. 7.19 przedstawia analogiczne wykresy dla  $\beta=5.7663$  i  $M=39$ , które zostały dobrane na drodze prób i błędów. Jak widać z rys. 7.19 wymagania projektowe są spełnione dokładnie.



Rys. 7.18 Charakterystyka amplitudowa i błąd aproksymacji dla filtra z oknem Kaisera bezpośrednio po zastosowaniu wzorów projektowych.



Rys. 7.19 Charakterystyka amplitudowa i błąd aproksymacji dla filtra z oknem Kaisera po korekcji parametrów  $M$  i  $\beta$ .

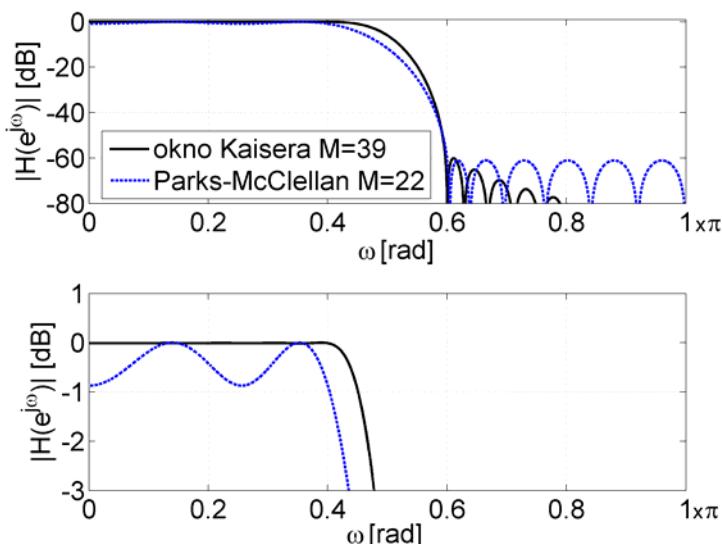
### 7.2.3 Optymalny filtr FIR - algorytm Parks-McClellana

Przy projektowaniu filtrów FIR metodą okien, okno prostokątne jest najlepszą, średniokwadratową aproksymacją zadanej charakterystyki częstotliwościowej dla danego rzędu  $M$ , tzn. minimalizuje błąd:

$$\varepsilon^2 = \frac{1}{2\pi} \int_{-\pi}^{\pi} |H_d(e^{j\omega}) - H(e^{j\omega})|^2 d\omega. \quad (7.37)$$

Niestety kryterium (7.37) nie uwzględnia oscylacji w punktach nieciągłości i uniemożliwia osobne traktowanie pasma przepustowego i zaporowego. Dlatego stosuje się optymalizację *min-max* (minimalizację błędów maksymalnych) oraz ważone częstotliwościowo kryteria błędów. W algorytmie Parks-McClellana rząd filtra  $M$ , krawędzie pasma  $\omega_p$  (przepustowego) i  $\omega_s$  (zaporowego) oraz stosunek  $\delta_p/\delta_s$  są stałe, natomiast  $\delta_p$  (tłumienie w paśmie przepustowym) lub  $\delta_s$  (tłumienie w paśmie zaporowym) jest zmienne. W porównaniu do metody okien, gdzie  $\delta_p=\delta_s$ , metodą Parks-McClellana można zaprojektować filtry FIR niższego rzędu spełniające te same wymagania zwłaszcza, jeżeli  $\delta_p$  znacznie różni się od  $\delta_s$ .

Rys.7.20 przedstawia charakterystyki amplitudowe filtra zaprojektowanego metodą okien z oknem Kaisera i filtra zaprojektowanego metodą Parks-McClellana na te same wymagania projektowe, tj.:  $\omega_p=0.4\pi$  rad,  $\omega_s=0.6\pi$  rad,  $r_p=1$  dB,  $r_s=60$  dB. Dla tych wymagań  $\delta_p=0.05$  a  $\delta_s=0.001$ . W metodzie Parks-McClellana zastosowano ważone kryterium częstotliwości z wagą  $\delta_p/\delta_s$  w paśmie zaporowym. Jak widać z rys.7.20 oba filtry spełniają zadane wymagania. Długość filtra wyznaczonego metodą optymalizacyjną wynosi  $M=22$ , a metodą okien  $M=39$ . Filtr wyznaczony metodą optymalizacyjną jest nazywany filtrem równomiernie falistym (*equiripple filter*) ze względu na kształt charakterystyki amplitudowej.



Rys.7.20 Charakterystyki amplitudowe filtra FIR zaprojektowanego metodą okien z oknem Kaisera i filtra FIR zaprojektowanego metodą Parks-McClellana na te same wymagania projektowe.

## 7.2.4 Filtr Hilberta

Odpowiedź częstotliwościowa filtru Hilberta jest zdefiniowana w następujący sposób:

$$H(e^{j\omega}) = \begin{cases} -j, & 0 \leq \omega < \pi \\ j, & -\pi \leq \omega < 0 \end{cases}. \quad (7.38)$$

Filtr (7.38) jest nazywany idealnym 90 stopniowym przesuwniukiem fazowym, filtrem kwadraturowym lub idealnym filtrem Hilberta. Odpowiedź impulsowa filtru Hilberta jest następująca:

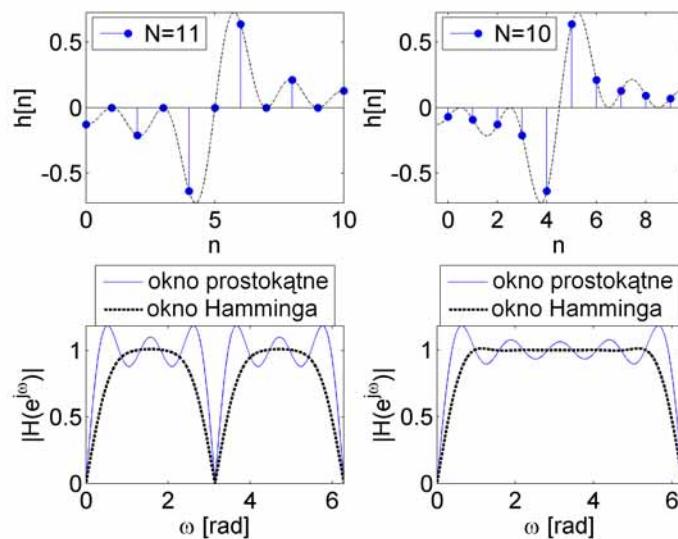
$$h[n] = \frac{1}{2\pi} \int_{-\pi}^0 j e^{j\omega n} d\omega - \frac{1}{2\pi} \int_0^\pi j e^{j\omega n} d\omega = \begin{cases} \frac{2 \sin^2(\pi n/2)}{\pi n}, & n \neq 0 \\ 0, & n = 0 \end{cases}. \quad (7.39)$$

Idealny filtr Hilberta (podobnie jak idealny filtr dolnoprzepustowy) jest nieprzyczynowy i ma nieskończoną odpowiedź impulsową. Aproxymację filtru Hilberta

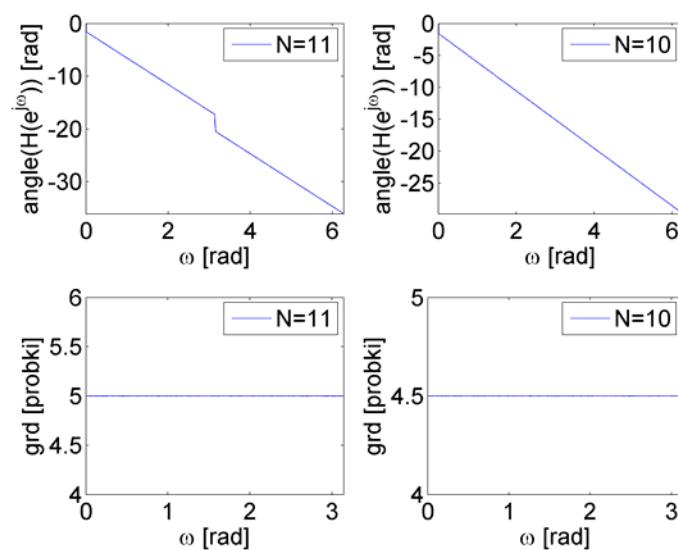
uzyskuje się metodą okien lub metodą Parks-McClellana. Dla filtrów przyczynowych opóźnienie o 90 stopni posiada dodatkową składową liniową. Do aproksymacji filtra Hilberta można również stosować filtry IIR w układzie *phase-splitter* złożonym z dwóch filtrów wszechprzepustowych, których charakterystyki fazowe różnią się w przybliżeniu o 90 stopni w przedziale  $0 < |\omega| < \pi$ .

Rys.7.21 przedstawia odpowiedzi impulsowe oraz charakterystyki amplitudowe filtrów Hilberta zaprojektowanych metodą okien dla symetrii typu III i IV. Odpowiedzi impulsowe wyznaczono z (7.39), tj. dla filtra nieprzyczynowego, a następnie przesunięto je w indeksie  $n$  tak, aby otrzymać, pokazane na rys.7.21, filtry przyczynowe. Symetria typu IV daje lepszą aproksymację charakterystyki amplitudowej, jednak wprowadza opóźnienie grupowe o pół próbki, co pokazano na rys.7.22.

Filtr Hilberta realizuje transformację Hilberta, tzn. tłumski składową stałą i opóźnia o 90 stopni wszystkie częstotliwości. Tak więc, transformatą Hilberta sygnału sinusoidalnego  $x[n] = \cos(\omega n)$  jest  $x_H[n] = \sin(\omega n)$ , a dla  $x[n] = \sin(\omega n)$  jest nią  $x_H[n] = -\cos(\omega n)$ .



Rys.7.21 Odpowiedzi impulsowe oraz charakterystyki amplitudowe filtrów Hilberta zaprojektowanych metodą okien dla symetrii typu III i IV.

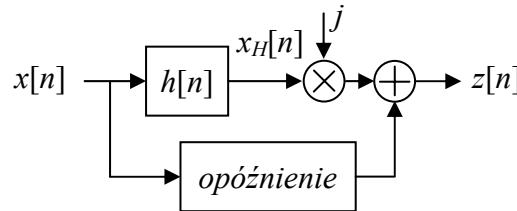


Rys.7.22 Charakterystyki fazowe i opóźnienie grupowe filtrów Hilberta z rys.7.21.

Filtr Hilberta jest stosowany do wyznaczania dyskretnego sygnału analitycznego (SA) zdefiniowanego jako:

$$z[n] = x[n] + jx_H[n] = A[n]e^{j\varphi[n]}, \quad (7.40)$$

gdzie  $x[n]$  jest sygnałem o wartościach rzeczywistych,  $x[n]=\text{Re}\{z[n]\}$ , a  $x_H[n]$  jest transformatą Hilberta sygnału  $x[n]$ . Zależność (7.40) jest zilustrowana na rys.7.23 z użyciem filtra Hilberta. Przy dodawaniu części rzeczywistej i urojonej należy uwzględnić opóźnienie wnoszone przez filtr Hilberta.



Rys.7.23 Schemat blokowy obliczania SA za pomocą filtra Hilberta.

W postaci wykładniczej SA definiuje obwiednię (amplitudę chwilową) sygnału  $x[n]$ :

$$A[n] = \sqrt{x^2[n] + x_H^2[n]}, \quad (7.41)$$

oraz fazę uogólnioną sygnału  $x[n]$ :

$$\varphi[n] = \arctan\left(\frac{x_H[n]}{x[n]}\right). \quad (7.42)$$

Faza uogólniona jest to argument funkcji  $\sin(\omega n + \varphi)$  i oprócz fazy sygnału  $\varphi$  zawiera ona rosnącą wraz z indeksem czasu składową  $\omega n$ . Przesunięcie fazowe pomiędzy dwoma przebiegami sinusoidalnymi jest różnicą ich faz uogólnionych, a pochodna fazy uogólnionej jest pulsacją chwilową.

Jako przykład zastosowania SA rozważmy demodulację amplitudy i częstotliwości. Sygnał sinusoidalny zmodulowany amplitudowo ma postać:

$$x[n] = (1 + k m[n]) \cos(\omega_0 n), \quad 0 < k \leq 1, \quad (7.43)$$

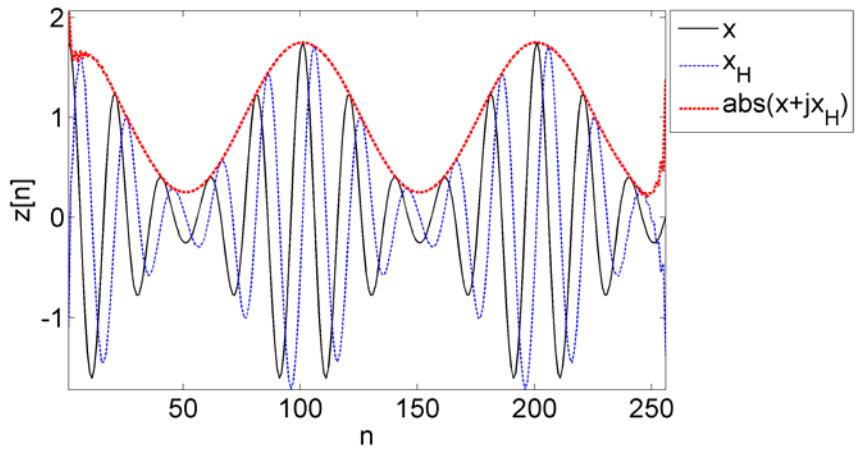
gdzie  $k$  jest głębokością modulacji, a sygnał modulujący  $m[n]$  może być np. przebiegiem sinusoidalnym  $m[n] = \cos(\omega_m n)$ ,  $\omega_m \ll \omega_0$ .

Częstotliwość chwilowa jest to pochodna argumentu funkcji kosinus, tak więc dla sygnału w postaci:

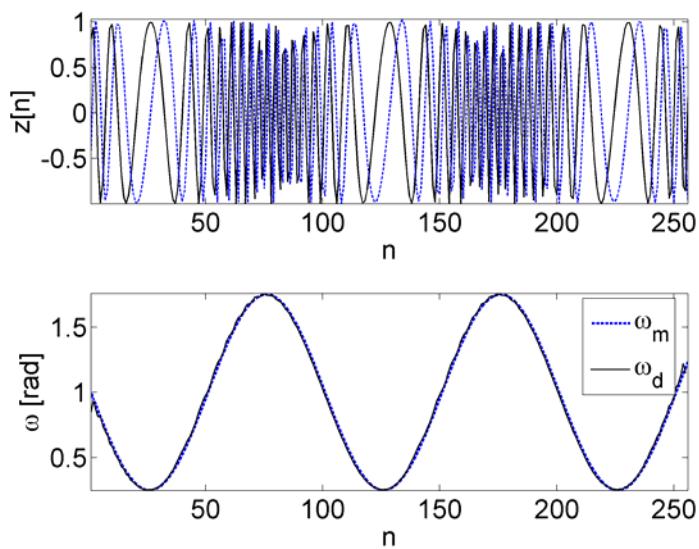
$$x[n] = \cos(\omega_0 n + k \cos(\omega_m n) / \omega_m), \quad (7.44)$$

częstotliwość chwilowa wynosi  $m[n] = \omega_0 - k \sin(\omega_m n)$ , czyli oscyluje wokół  $\omega_0$ .

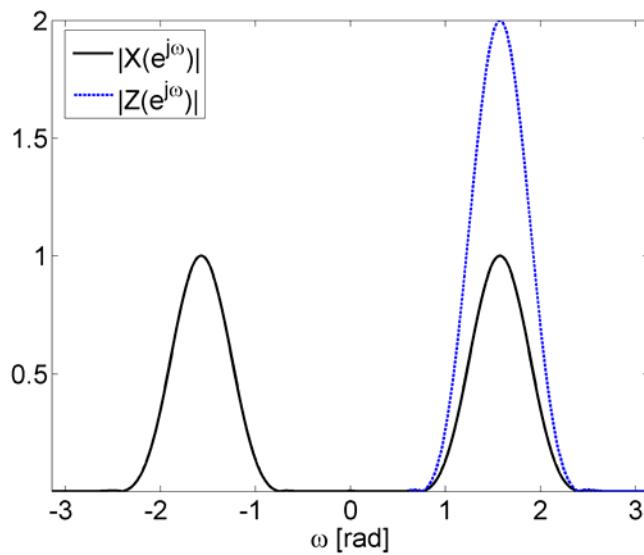
Rys.7.24 przedstawia sygnał zmodulowany amplitudowo, odpowiadający mu SA oraz obwiednię, czyli sygnał modulujący wyznaczoną jako moduł SA. Rys.7.25 przedstawia SA sygnału zmodulowanego w częstotliwości oraz zadaną i zdemodulowaną, tj. policzoną jako pochodną (7.42), częstotliwość modulacji. W obu przypadkach widoczna jest zgodność sygnału zdemodulowanego za pomocą SA z sygnałem modulującym.



Rys.7.24 Zastosowanie SA do demodulacji amplitudy.



Rys.7.25 Zastosowanie SA do demodulacji częstotliwości  $\omega_m$  - pulsacja zadana  $\omega_d$  - pulsacja zdemodulowana.



Rys.7.26 Widmo sygnału  $x[n]$  i SA  $z[n]=x[n]+jx_H[n]$ .

Widmo SA jest prawostronne, co pokazano na rys.7.26. Dla pulsacji ujemnych przyjmuje ono wartość zero, natomiast dla pulsacji dodatnich wartości dwukrotnie większe

niż widmo sygnału  $x[n]$ . Własność ta wykorzystywana jest w telekomunikacji, SA zajmuje o połowę mniejsze pasmo częstotliwości, dzięki czemu dwukrotnie wzrasta przepustowość kanału.

SA można również wyznaczyć metodą modyfikacji widma. Dla sygnału  $x[n]$  wyznacza się dyskretną transformatę Fouriera (opisaną w rozdziale 8)  $X[k]$ , a następnie modyfikuje widmo według reguły:

$$Z[k] = \begin{cases} X[k], & k=0, \\ 2X[k], & k=1, 2, \dots, \frac{N}{2}-1, \\ X[k], & k=\frac{N}{2}, \\ 0, & k=\frac{N}{2}+1, \dots, N-1. \end{cases} \quad (7.45)$$

Sygnal analityczny  $z[n]$  jest obliczany jako odwrotna, dyskretna transformata Fouriera sygnału  $Z[k]$ .

### 7.2.5 Dyskretne układy różniczkujące

Dla sygnałów ciągłych widmo pochodnej sygnału można wyznaczyć licząc widmo tego sygnału, które następnie należy przemnożyć przez  $j\Omega$ . Dla sygnałów dyskretnych idealny filtr różniczkujący (o liniowej charakterystyce fazowej) definiuje się w sposób następujący:

$$H(e^{j\omega}) = \frac{1}{T}(j\omega)e^{-j\omega M/2}, \quad -\pi < \omega < \pi. \quad (7.46)$$

Odpowiedź impulsowa filtra różniczkującego wynosi:

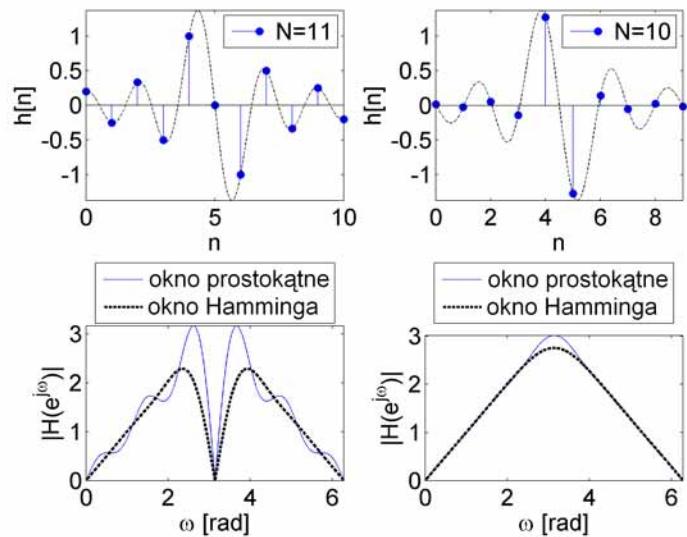
$$h[n] = \begin{cases} \frac{\cos(\pi n)}{n} - \frac{\sin(\pi n)}{\pi n^2}, & n \neq 0 \\ 0, & n = 0 \end{cases}. \quad (7.47)$$

Dla indeksów  $n$  będących liczbami całkowitymi (7.47) upraszcza się do:

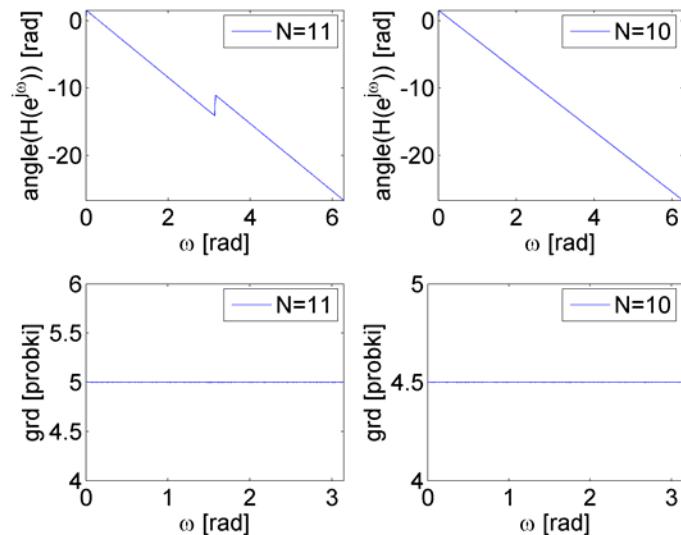
$$h[n] = \frac{\cos(\pi n)}{n} = \frac{-1^n}{n}, \quad n \neq 0. \quad (7.48)$$

Rys.7.27 przedstawia odpowiedzi impulsowe oraz charakterystyki amplitudowe filtrów różniczkujących zaprojektowanych metodą okien dla symetrii typu III i IV. Symetria typu IV daje lepszą aproksymację charakterystyki amplitudowej, jednak wprowadza opóźnienie grupowe o pół próbki, co pokazano na rys.7.28.

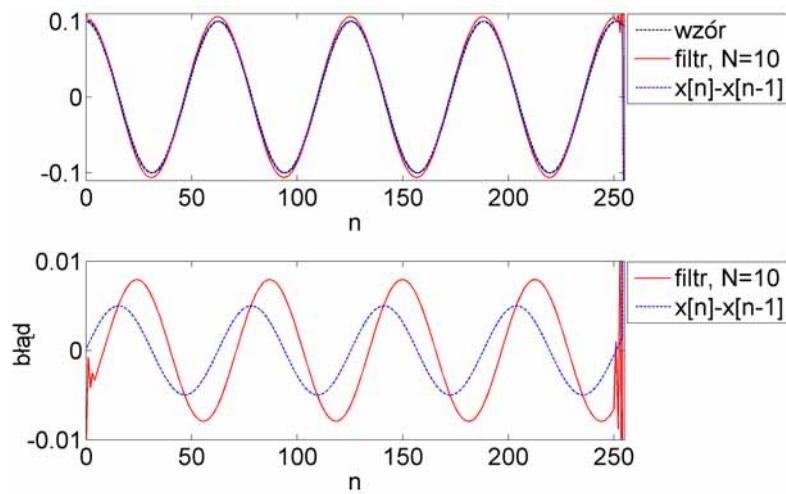
Jako przykład zastosowania filtra różniczkującego rozważmy filtrację sygnału testowego w postaci  $x[n]=\sin(\omega n)$ . Pochodna tego sygnału wynosi  $y[n]=\omega \cos(\omega n)$ . Rys.7.29 przedstawia wyniki wyznaczenia pochodnej za pomocą filtra różniczkującego oraz za pomocą różnicy wstecznej (*backward difference*). Dla porównania pokazano również teoretyczny przebieg pochodnej sygnału. Błąd wyznaczenia pochodnej pokazany na dolnym wykresie na rys.7.29 został wyznaczony jako różnica pomiędzy wartością teoretyczną a wartością obliczoną obiema metodami.



Rys.7.27 Odpowiedzi impulsowe oraz charakterystyki amplitudowe filtrów różniczkujących zaprojektowanych metodą okien dla symetrii typu III i IV.



Rys.7.28 Charakterystyki fazowe i opóźnienie grupowe filtrów różniczkujących z rys.7.27.



Rys.7.29 Obliczenie pochodnej sygnału  $x[n]=\sin(\omega n)$  i błąd wyznaczony jako różnica pomiędzy wartością teoretyczną a obliczoną.

## 8. Dyskretne przekształcenie Fouriera

### 8.1 Dyskretny Szereg Fouriera

Rozważmy nieskończony ciąg okresowy  $\tilde{x}[n] = \tilde{x}[n+rN]$ ,  $n, r$  - liczby całkowite. Taki ciąg może być reprezentowany przez dyskretny szereg Fouriera DFS (*Discrete Fourier Series*) jako suma harmonicznych, zespolonych sygnałów eksponencjalnych tj. sygnałów eksponencjalnych z częstotliwościami będącymi krotnościami częstotliwości podstawowej  $\omega=2\pi/N$  ciągu  $\tilde{x}[n]$  o następującej postaci:

$$e_k[n] = e^{j(2\pi/N)kn} = e_k[n+rN], k - \text{liczba całkowita.} \quad (8.1)$$

Dyskretny szereg Fouriera ma wówczas następującą postać:

$$\tilde{x}[n] = \frac{1}{N} \sum_k \tilde{X}[k] e^{j(2\pi/N)kn}. \quad (8.2)$$

Ponieważ ciągi  $e_k[n]$  są okresowe względem  $k$  z okresem  $N$ :

$$e_{k+lN}[n] = e^{j(2\pi/N)(k+lN)n} = e^{j(2\pi/N)kn} e^{j2\pi ln} = e^{j(2\pi/N)kn} = e_k[n], l - \text{liczba naturalna,} \quad (8.3)$$

więc sumowanie w (8.2) wystarczy przeprowadzić dla  $N$  wartości:

$$\tilde{x}[n] = \frac{1}{N} \sum_{k=0}^{N-1} \tilde{X}[k] e^{j(2\pi/N)kn}. \quad (8.4)$$

W celu wyznaczenia współczynników  $\tilde{X}[k]$  DFS pomóżmy (8.4) przez  $e^{-j(2\pi/N)rn}$  i zsumujmy od  $n=0$  do  $n=N-1$ :

$$\sum_{n=0}^{N-1} \tilde{x}[n] e^{-j(2\pi/N)rn} = \sum_{n=0}^{N-1} \frac{1}{N} \sum_{k=0}^{N-1} \tilde{X}[k] e^{j(2\pi/N)(k-r)n}. \quad (8.5)$$

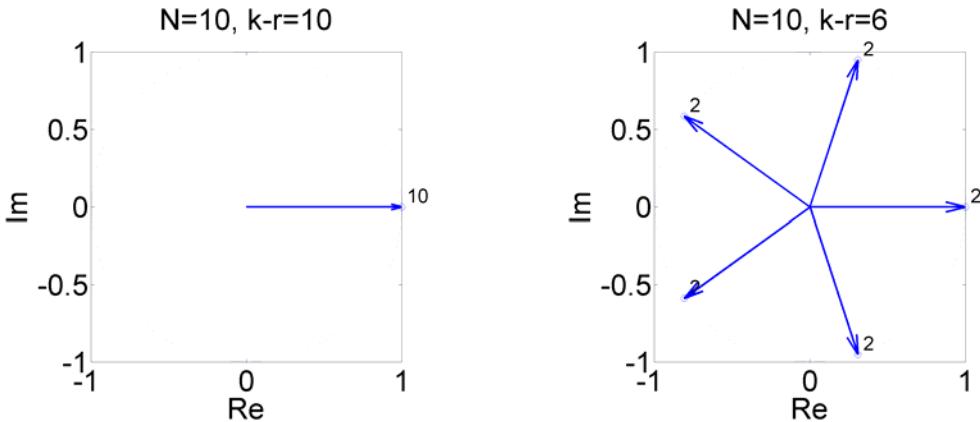
Po zamianie kolejności sumowania po prawej stronie równania otrzymujemy:

$$\sum_{n=0}^{N-1} \tilde{x}[n] e^{-j(2\pi/N)rn} = \sum_{k=0}^{N-1} \tilde{X}[k] \left( \frac{1}{N} \sum_{n=0}^{N-1} e^{j(2\pi/N)(k-r)n} \right). \quad (8.6)$$

Dla ciągów  $e_k[n]$  zachodzi:

$$\frac{1}{N} \sum_{n=0}^{N-1} e^{j(2\pi/N)(k-r)n} = \begin{cases} 1, & k-r = mN, \quad m - \text{liczba całkowita} \\ 0, & \text{w przeciwnym wypadku} \end{cases}. \quad (8.7)$$

Własność (8.7) jest zilustrowana na rys.8.1. Na płaszczyźnie zespolonej wyrazy ciągu  $e_k[n]$  są wektorami o długości 1, poczatku w punkcie  $(0, j0)$  i koncu na okregu jednostkowym. W przypadku, gdy  $k-r=mN$ , wszystkie wektory leżą na osi rzeczywistej, a ich suma jest równa ich liczbie, w przeciwnym przypadku suma wektorów wynosi zero.



Rys.8.1 Ilustracja własności (8.7).

Równanie (8.6) można więc przedstawić w postaci:

$$\sum_{n=0}^{N-1} \tilde{x}[n] e^{-j(2\pi/N)rn} = \tilde{X}[r]. \quad (8.8)$$

Współczynniki DFS wyznacza się z zależności:

$$\tilde{X}[k] = \sum_{n=0}^{N-1} \tilde{x}[n] e^{-j(2\pi/N)kn}. \quad (8.9)$$

Ciąg  $\tilde{X}[k]$  jest okresowy z okresem  $N$ :

$$\tilde{X}[k+N] = \sum_{n=0}^{N-1} \tilde{x}[n] e^{-j(2\pi/N)(k+N)n} = \sum_{n=0}^{N-1} \tilde{x}[n] e^{-j(2\pi/N)kn} e^{-j2\pi n} = \tilde{X}[k]. \quad (8.10)$$

## 8.2 Dyskretnie przekształcenie Fouriera

Rozważmy ciąg  $x[n]$  o skończonej długości  $N$ , tzn.  $x[n]=0$  dla  $0 < n < N-1$ . Z ciągu  $x[n]$  można zawsze utworzyć ciąg okresowy w ten sposób, że  $x[n]$  będzie jednym okresem:

$$\tilde{x}[n] = \sum_{r=-\infty}^{\infty} x[n-rN], \text{ wówczas } x[n] = \begin{cases} \tilde{x}[n], & 0 \leq n \leq N-1, \\ 0, & \text{w przeciwnym wypadku.} \end{cases} \quad (8.11)$$

Z okresowego DFS  $\tilde{X}[k]$  można wybrać ciąg  $X[k]$  o skończonej długości równej jednemu okresowi:

$$X[k] = \begin{cases} \tilde{X}[k], & 0 \leq k \leq N-1 \\ 0, & \text{w przeciwnym wypadku} \end{cases} \quad (8.12)$$

Podstawiając (8.11) i (8.12) do (8.4) i (8.9) otrzymujemy dyskretnie przekształcenie Fouriera oznaczane jako DFT (*Discrete Fourier Transform*).

DFT analiza (przekształcenie proste):

$$X[k] = \sum_{n=0}^{N-1} x[n] e^{-j(2\pi/N)kn}, \quad 0 \leq n \leq N-1, \quad 0 \leq k \leq N-1$$

(8.13)

Dla porównania transformacja Fouriera jest zdefiniowana jako  $X(e^{j\omega}) = \sum_{n=-\infty}^{\infty} x[n]e^{-j\omega n}$ .

DFT synteza (przekształcenie odwrotne, IDFT - *inverse DFT*):

$$x[n] = \frac{1}{N} \sum_{k=0}^{N-1} X[k] e^{j(2\pi/N)kn}, \quad 0 \leq n \leq N-1, \quad 0 \leq k \leq N-1. \quad (8.14)$$

Dla porównania odwrotna transformacja Fouriera jest zdefiniowana jako  $x[n] = \frac{1}{2\pi} \int_{-\pi}^{\pi} X(e^{j\omega}) e^{j\omega n} d\omega$ .

DFT zdefiniowane równaniami (8.13) i (8.14) jest jednym z najbardziej popularnych i najpowszechniej stosowanych narzędzi przetwarzania sygnałów cyfrowych.

Jako przykład obliczymy DFT sygnału dyskretnego  $x[n]=[1,2,3,4]$ . We wzorze (8.13) sumowanie po  $n$  przebiega dla ustalonej wartości  $k$ . Obliczenia mają więc następującą postać:

$$\begin{aligned} X[0] &= \sum_{n=0}^{4-1} x[n] e^{-j(2\pi/4)0n} = x[0] + x[1] + x[2] + x[3] = 10, \\ X[1] &= \sum_{n=0}^3 x[n] e^{-j(2\pi/4)1n} = x[0]e^0 + x[1]e^{-j(2\pi/4)} + x[2]e^{-j(2\pi/4)2} + x[3]e^{-j(2\pi/4)3} = 1 - j2 - 3 + j4 = -2 + j2, \\ X[2] &= \sum_{n=0}^3 x[n] e^{-j(2\pi/4)2n} = x[0]e^0 + x[1]e^{-j(2\pi/4)2} + x[2]e^{-j(2\pi/4)2 \cdot 2} + x[3]e^{-j(2\pi/4)2 \cdot 3} = 1 - 2 + 3 - 4 = -2, \\ X[3] &= \sum_{n=0}^3 x[n] e^{-j(2\pi/4)3n} = x[0]e^0 + x[1]e^{-j(2\pi/4)3} + x[2]e^{-j(2\pi/4)3 \cdot 2} + x[3]e^{-j(2\pi/4)3 \cdot 3} = 1 + j2 - 3 - j4 = -2 - j2. \end{aligned}$$

Tak, więc widmo DFT sygnału  $x[n]=[1, 2, 3, 4]$  wynosi  $X[k]=[10, -2+j2, -2, -2-j2]$ . Jak widać z powyższych obliczeń, dla  $k=0$  otrzymujemy sumę wartości wszystkich wyrazów ciągu  $x[n]$ , jest to składowa stała sygnału.

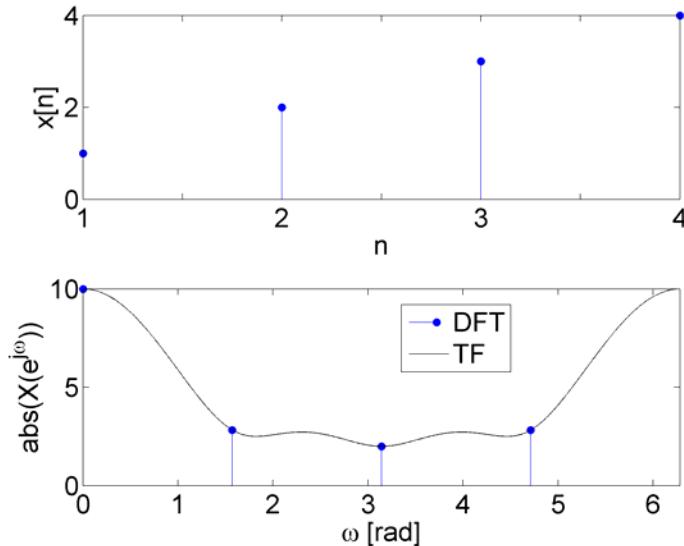
Zależność (8.13) można przedstawić w postaci macierzowej. Dla sygnału  $x[n]=[1,2,3,4]$  otrzymujemy:

$$\begin{bmatrix} X[0] \\ X[1] \\ X[2] \\ X[3] \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -j & -1 & j \\ 1 & -1 & 1 & -1 \\ 1 & j & -1 & -j \end{bmatrix} \begin{bmatrix} x[0] \\ x[1] \\ x[2] \\ x[3] \end{bmatrix} = \begin{bmatrix} 10 \\ -2+j2 \\ -2 \\ -2-j2 \end{bmatrix}.$$

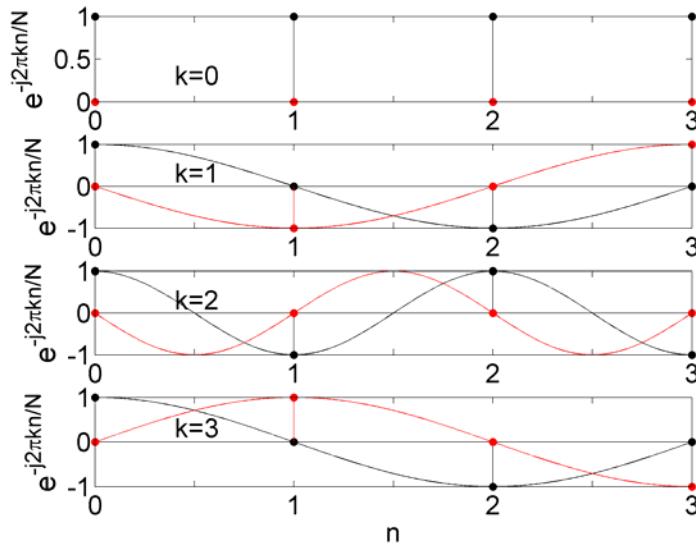
Bezpośrednie wyznaczenie DFT ze wzoru definicyjnego (8.13) wymaga  $N^2$  zespolonych mnożeń i  $(N-1)$  zespolonych dodawań.

Porównując ze sobą wzory na DFT i transformację Fouriera widzimy, że DFT wyznacza widmo sygnału dyskretnego dla częstotliwości  $\omega=(2\pi/N)k$ , tzn. dla  $N$  równoodległych prążków częstotliwościowych na okręgu jednostkowym od  $\omega=0$  z krokiem  $2\pi/N$ . Mówimy, więc że DFT próbkuje ciągłe widmo sygnału dyskretnego.

Rys.8.2 przedstawia przykładowy ciąg  $x[n]$  oraz jego widmo ciągłe i widmo DFT. Rys.8.3 przedstawia zespolone funkcje bazowe DFT dla sygnału o długości  $N=4$ . Pierwsza funkcja bazowa, dla  $k=0$ , jest to funkcja stała o wartościach rzeczywistych, iloczyn wektorowy tej funkcji bazowej z sygnałem powoduje zsumowanie wyrazów ciągu. Druga funkcja bazowa, dla  $k=1$ , określa krok próbkowania w osi częstotliwości, iloczyn wektorowy tej funkcji bazowej z sygnałem określa podobieństwo sygnału do funkcji bazowej. Kolejne funkcje bazowe są harmonicznymi drugiej funkcji bazowej, tj. dla  $k=1$ , a ich iloczyny wektorowe z sygnałem określają zawartość tych częstotliwości w sygnale.



Rys.8.2 Ciąg  $x[n]$  oraz jego widmo ciągłe i widmo DFT.



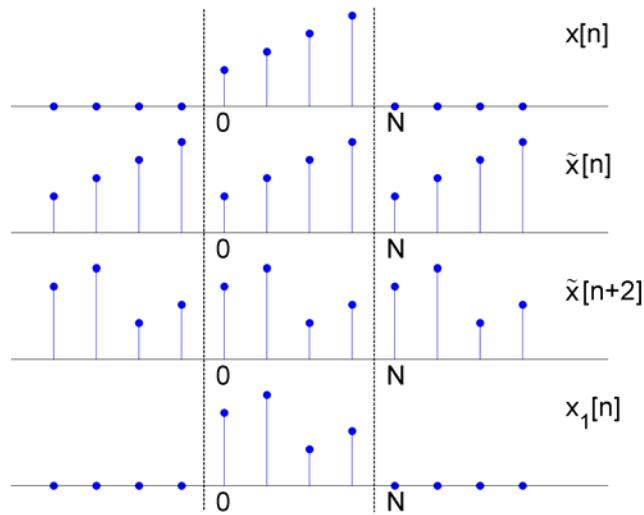
Rys.8.3 Zespolone funkcje bazowe DFT dla sygnału o długości  $N=4$ .

### 8.2.1 Splot kołowy

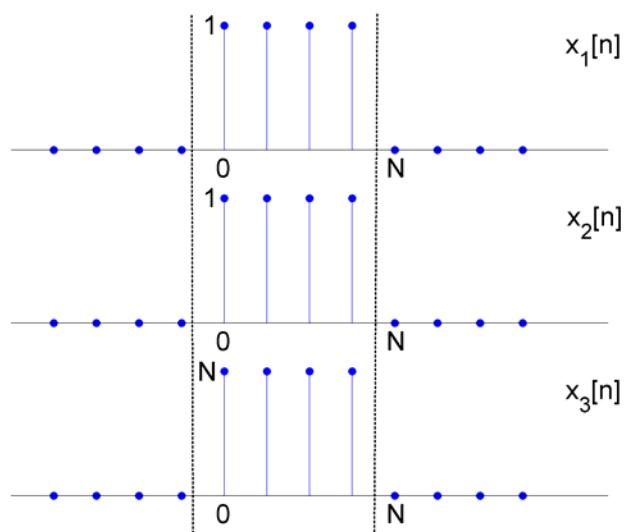
DFT ma własność splotu kołowego, tzn. dla ciągów i ich widm DFT  $x_1[n] \leftrightarrow X_1[k]$  i  $x_2[n] \leftrightarrow X_2[k]$ , jeżeli zachodzi  $X_3[k] = X_1[k]X_2[k]$ , to:

$$x_3[n] = \sum_{m=0}^{N-1} x_1[m]x_2[(n-m)_N], \quad 0 \leq n < N-1, \quad (8.15)$$

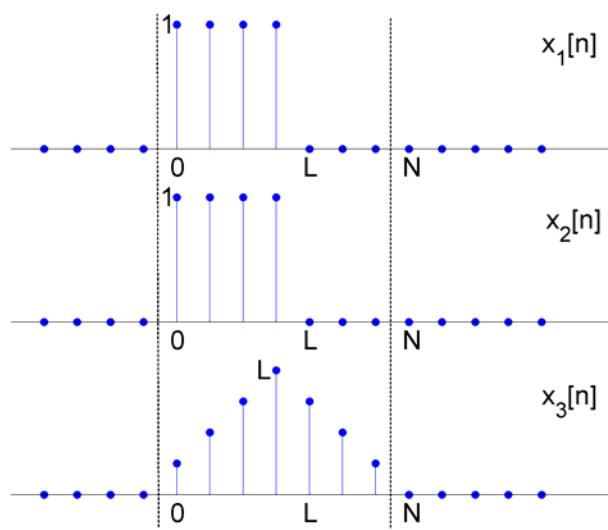
gdzie zapis  $x_2[(n-m)_N]$  oznacza przesunięcie kołowe (modulo  $N$ ); dla porównania splot liniowy opisany jest zależnością  $x_3[n] = \sum_{m=-\infty}^{\infty} x_1[m]x_2[n-m]$ . Zależność (8.15) oznacza, że iloczyn widm DFT sygnałów jest równoważny widmu splotu kołowego tych sygnałów.



Rys.8.4 Przesunięcie kołowe ciągu.



Rys.8.5 Splot kołowy.



Rys.8.6 Splot kołowy równoważny za okres splotowi liniowemu.

Rys.8.4 ilustruje przesunięcie kołowe ciągu  $x[n]$ . Z ciągu  $x[n]$  o  $N$  niezerowych elementach jest tworzony nieskończony ciąg okresowy  $\tilde{x}[n]$ . Przesunięcie ciągu okresowego powoduje, że w ciągu  $x_1[n]$  zamiast zer pojawiają się wyrazy z sąsiedniego okresu.

Splot kołowy dwóch ciągów stałych o długości  $N$  i amplitudzie 1 jest przedstawiony na rys.8.5, wynik tego splotu jest również ciągiem stałym, którego elementy mają wartość  $N$ .

Rys.8.6 Przedstawia splot kołowy dwóch ciągów stałych o długości  $N$ , które wartości niezerowe mają na pozycjach od 0 do  $L-1$ . W tym przypadku wynik splotu kołowego za jeden okres jest taki sam jak wynik splotu liniowego, tak więc splot liniowy można liczyć za pomocą DFT pod warunkiem, że sygnały zostaną uzupełnione zerami do odpowiedniej długości.

## 8.2.2 Obliczanie splotu liniowego za pomocą splotu kołowego

Wynik splotu liniowego  $x_3[n]$  sygnałów  $x_1[n]$  i  $x_2[n]$  o długościach  $N_1$  i  $N_2$  (tzn.  $x_1[n]=0$ ,  $0 < n > N_1 - 1$ ,  $x_2[n]=0$ ,  $0 < n > N_2 - 1$ ) może mieć, co najwyżej  $N_1+N_2-1$  niezerowych elementów:  $x_3[n]=0$ ,  $0 < n > N_2 + N_1 - 2$ .

Splot kołowy odpowiada splotowi liniowemu sygnałów o skończonej długości, jeżeli sygnały  $x_1[n]$  i  $x_2[n]$  zostaną uzupełnione zerami do długości  $N_1+N_2-1$  (lub większej) przed policzeniem DFT. Algorytm liczenia splotu liniowego sygnałów o skończonej długości poprzez DFT jest następujący:

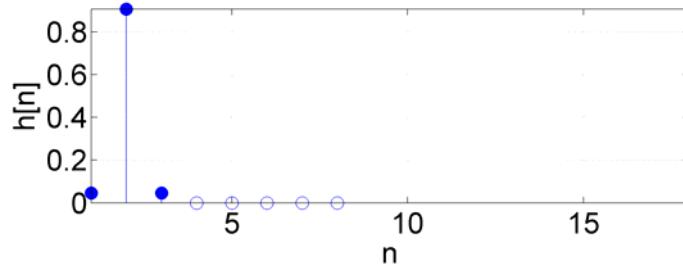
1. Określić długości  $N_1$  i  $N_2$  sygnałów  $x_1[n]$  i  $x_2[n]$ ,
2. Uzupełnić zerami sygnały  $x_1[n]$  i  $x_2[n]$  do długości  $N_1+N_2-1$ ,
3. Obliczyć DFT obu sygnałów,
4. Obliczyć odwrotne DFT iloczynu widm.

## 8.2.3 Splot sekcjonowany

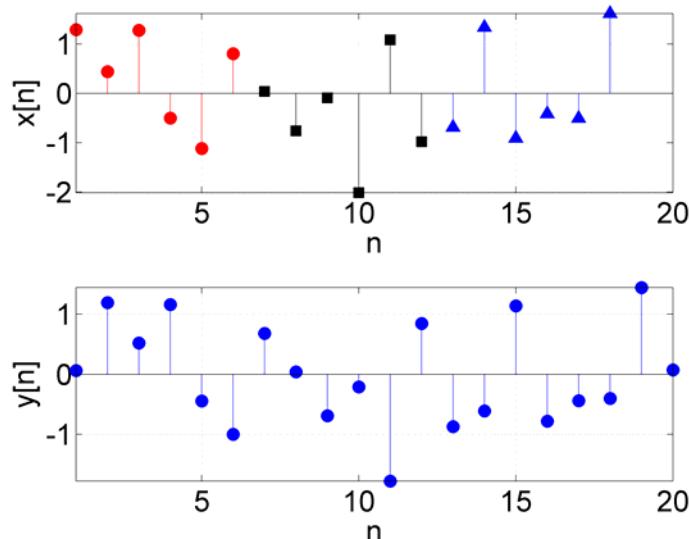
Splot sekcjonowany jest algorytmem liczenia splotu liniowego poprzez DFT. Stosuje się go, gdy jeden z sygnałów jest zdecydowanie dłuższy. Dzięki temu unika się konieczności liczenia DFT dla długich sygnałów oraz redukuje opóźnienia związane z rejestracją (można liczyć splot na bieżąco, tj. w miarę napływających próbek rejestrowanego sygnału, np. mowy, bez konieczności zapisania całości sygnału). Splot sekcjonowany może być realizowany dwiema metodami: jako sekwencja splotów liniowych - metoda *overlap-add* lub sekwencja splotów kołowych - metoda *overlap-save*.

Rys.8.7-8.9 ilustrują metodę *overlap-add*. Sygnał  $x[n]$  jest dzielony na niezachodzące na siebie fragmenty o takiej samej długości. Rys.8.8 przedstawia podział na sekcje o długości 6 próbek. Następnie odpowiedź impulsowa filtra pokazana na rys.8.7 i każda bieżąca sekcja sygnału  $x[n]$  są uzupełniane zerami tak, aby splot kołowy był równoważny splotowi liniowemu. W omawianym przykładzie oba sygnały są uzupełniane zerami do długości  $6+3-1=8$  próbek, co pokazano na rys.8.7 dla filtru i rys.8.9 dla sygnału. DFT filtru  $h[n]$  uzupełnionego zerami wystarczy policzyć tylko raz. Dla kolejnych, uzupełnionych zerami, sekcji sygnału wyznacza się splot liniowy poprzez DFT. Wyniki przykładowych obliczeń przedstawione są na rys.8.9 (prawa kolumna). Końcowy wynik splotu  $y[n]$  pokazany na rys.8.8 otrzymuje się przez sumowanie wyników splotów dla kolejnych sekcji.

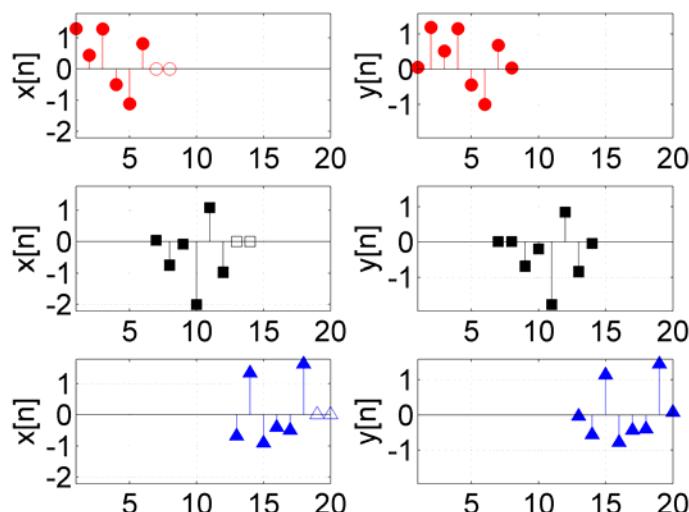
Rys.8.10-8.12 przedstawiają metodę *overlap-save*. Sygnał  $x[n]$  pokazany na rys.8.10 dzielony jest na zachodzące na siebie fragmenty pokazane na rys.8.11. Odpowiedź impulsowa filtra jest uzupełniana zerami do długości sekcji. Poprzez DFT wyznaczane są sploty kołowe dla poszczególnych sekcji, dlatego w wyniku należy odrzucić  $M-1$  pierwszych próbek, gdzie  $M$  jest długością filtru, co pokazano na rys.8.11 (prawa kolumna). Wynik splotu przedstawiony na rys.8.12 składa się z wyników splotów dla poszczególnych sekcji.



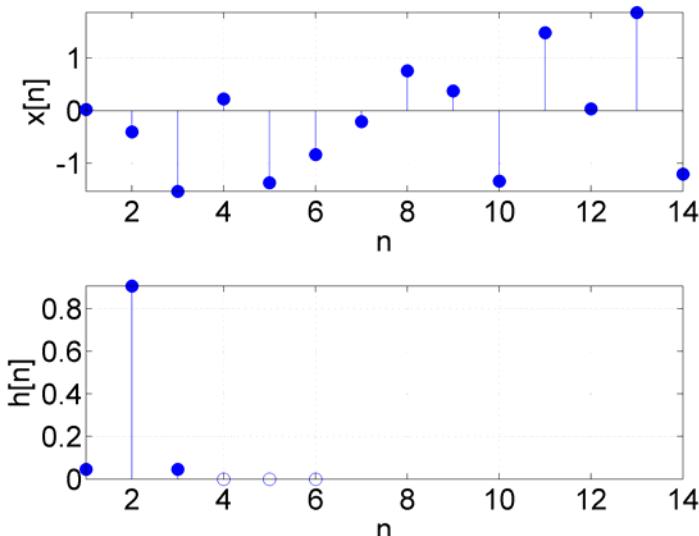
Rys.8.7 Odpowiedź impulsowa filtra uzupełniona zerami.



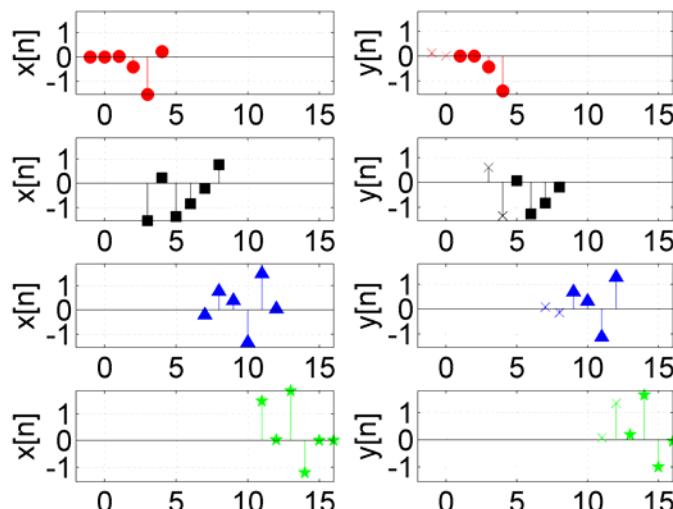
Rys.8.8 Podział sygnału na sekcje i wynik splotu liniowego.



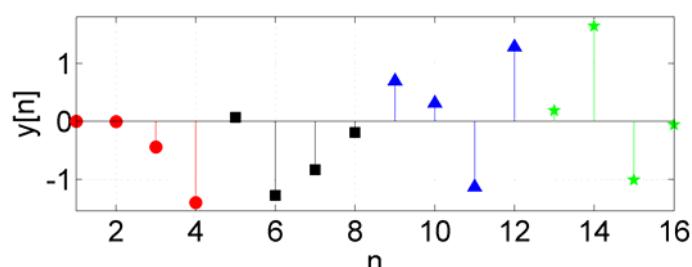
Rys.8.9 Metoda *overlap-add* - podział sygnału na niezachodzące na siebie sekcje i wyniki splotów liniowych dla sekcji.



Rys.8.10 Sygnał i odpowiedź impulsowa filtra uzupełniona zerami.



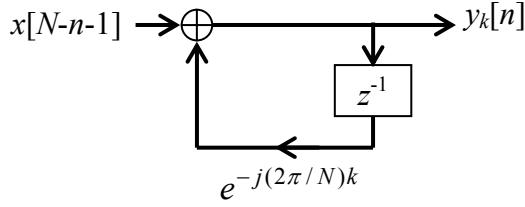
Rys.8.11 Metoda *overlap-save* - podział sygnału na zachodzące na siebie sekcje i wyniki splotów kołowych dla sekcji.



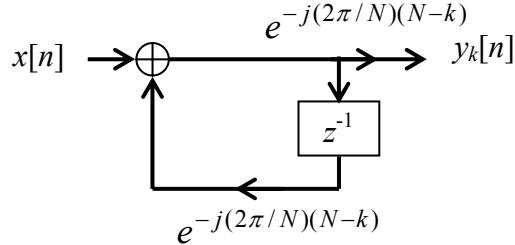
Rys.8.12 Wynik splotu liniowego.

#### 8.2.4 Obliczanie DFT algorytmem Goertzla

Algorytm Goertzla umożliwia rekurencyjne obliczenie DFT. Algorytmy Goertzla w wersji z odwróconą kolejnością próbek na wejściu i z naturalną kolejnością próbek na wejściu są przedstawione na rys.8.13 i rys.8.14. Naturalna kolejność próbek na wejściu umożliwia wyznaczanie DFT sygnału na bieżąco, tj. w miarę rejestracji sygnału.



Rys.8.13 Algorytm Goertzla - odwrócona kolejność próbek na wejściu.



Rys.8.14 Algorytm Goertzla - naturalna kolejność próbek na wejściu.

Rozważmy przepływ sygnału w układzie z rys.8.13 dla wektora  $x[n]$  o długości  $N=4$ ,  $n=0, \dots, 3$ ,  $k=0, \dots, 3$ :

dla  $k=0$ , otrzymujemy:

$$\begin{aligned} y_0[0] &= 0 + x[3] \\ y_0[1] &= x[2] + y_0[0]e^{-j(2\pi/4)0} = x[2] + x[3] \\ y_0[2] &= x[1] + x[2] + x[3] \\ y_0[3] &= x[0] + x[1] + x[2] + x[3] \end{aligned}$$

dla  $k=1$ , otrzymujemy:

$$\begin{aligned} y_1[0] &= 0 + x[3] \\ y_1[1] &= x[2] + y_1[0]e^{-j(2\pi/4)1} = x[2] + x[3]e^{-j(2\pi/4)} \\ y_1[2] &= x[1] + x[2]e^{-j(2\pi/4)} + x[3]e^{-j2(2\pi/4)} \\ y_1[3] &= x[0] + x[1]e^{-j(2\pi/4)} + x[2]e^{-j2(2\pi/4)} + x[3]e^{-j3(2\pi/4)} \end{aligned}$$

Dla  $n=N-1$  wartości  $y_k[n]$  z rys.8.13 odpowiadają prążkom DFT  $X[k]$ . Ten sposób obliczeń nie wymaga przechowywania w pamięci baz DFT, ponieważ są one wyznaczane rekurencyjnie. Złożoność obliczeniowa jest zbliżona do bezpośredniego wyliczania DFT ze wzoru definicyjnego.

Rozważmy przepływ sygnału w układzie z rys.8.14 dla wektora  $x[n]$  o długości  $N=4$ ,  $n=0, \dots, 3$ ,  $k=0, \dots, 3$ :

$$k = 0$$

$$y_0[3] = (x[3] + x[2] + x[1] + x[0])e^{-j(2\pi/4)4} = x[3] + x[2] + x[1] + x[0]$$

$$k = 1$$

$$y_1[3] = (x[3] + x[2]e^{-j3(2\pi/4)} + x[1]e^{-j6(2\pi/4)} + x[0]e^{-j9(2\pi/4)})e^{-j3(2\pi/4)}$$

$$y_1[3] = x[0]e^{-j12(2\pi/4)} + x[1]e^{-j9(2\pi/4)} + x[2]e^{-j6(2\pi/4)} + x[3]e^{-j3(2\pi/4)}$$

$$y_1[3] = x[0]e^{-j0} + x[1]e^{-j(2\pi/4)} + x[2]e^{-j2(2\pi/4)} + x[3]e^{-j3(2\pi/4)}$$

Dla  $n=N-1$  wartości  $y_k[n]$  z rys.8.14 odpowiadają prążkom DFT  $X[k]$ .

## 8.2.5 Transformacja Chirp-Z

W transformacji Chirp-Z, oznaczanej jako CZT, współczynniki Fouriera sygnału dyskretnego wyznacza się jako splot sygnału z predefiniowanymi funkcjami bazowymi. Możliwe jest

wyznaczenie dowolnego ciągu równoodległych współczynników Fouriera na okręgu jednostkowym.

W celu wyprowadzenia transformacji CZT założmy, że chcemy wyznaczyć współczynniki Fouriera  $X(e^{j\omega})$  ciągu dyskretnego  $x[n]$ ,  $n = 0, 1, \dots, N-1$  dla zbioru wartości  $\omega_k = \omega_0 + k\Delta\omega$ ,  $k = 0, 1, \dots, M-1$  równomiernie rozmieszczonego na okręgu jednostkowym. Pulsacja początkowa  $\omega_0$ , przyrost pulsacji  $\Delta\omega$  oraz liczba współczynników Fouriera  $M$  mogą być dobrane arbitralnie. W szczególnym przypadku dla  $\omega_0 = 0$ ,  $M = N$  i  $\Delta\omega = 2\pi/N$  otrzymujemy DFT. Transformacja Fouriera w wybranych punktach  $\omega_k$  ma postać:

$$X(e^{j\omega_k}) = \sum_{n=0}^{N-1} x[n] e^{-j\omega_k n}, \quad k = 0, 1, \dots, M-1, \quad (8.16)$$

podstawiając  $W = e^{-j\Delta\omega}$  oraz  $\omega_k = \omega_0 + k\Delta\omega$ :

$$X(e^{j\omega_k}) = \sum_{n=0}^{N-1} x[n] e^{-j\omega_0 n} W^{nk}. \quad (8.17)$$

Korzystając z tożsamości  $nk = \frac{1}{2} [n^2 + k^2 - (k-n)^2]$ :

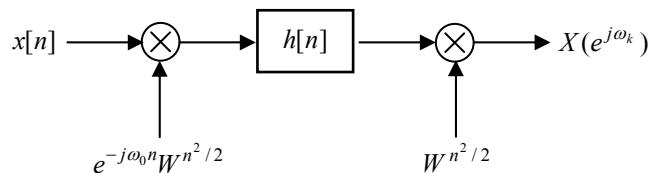
$$X(e^{j\omega_k}) = \sum_{n=0}^{N-1} x[n] e^{-j\omega_0 n} W^{n^2/2} W^{k^2/2} W^{-(k-n)^2/2} \quad (8.18)$$

i podstawiając  $g[n] = x[n] e^{-j\omega_0 n} W^{n^2/2}$  otrzymujemy:

$$X(e^{j\omega_k}) = W^{k^2/2} \left( \sum_{n=0}^{N-1} g[n] W^{-(k-n)^2/2} \right), \quad k = 0, 1, \dots, M-1. \quad (8.19)$$

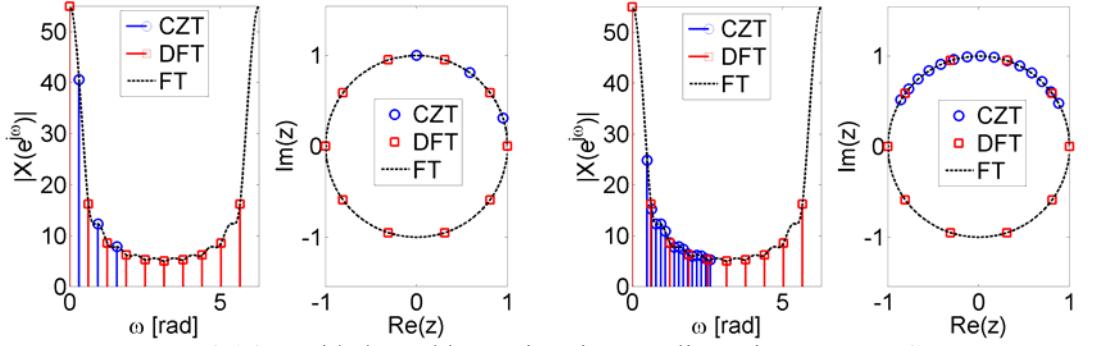
Zgodnie z (8.19),  $X(e^{j\omega_k})$  jest splotem ciągu  $g[n]$  z ciągiem  $W^{-k^2/2}$ . Splot ten może zostać zrealizowany przez trzy FFT (algorytmy FFT są omówione w następnym podrozdziale): 1) FFT funkcji  $W^{-k^2/2}$  (wykonywane tylko raz), 2) FFT sygnału  $g[n]$  oraz 3) IFFT iloczynu widm. Schemat blokowy obliczeń CZT przedstawia rys.8.15.

Rys.8.16 przedstawia przykłady próbkowania osi częstotliwości za pomocą CZT, dla porównania wykresy przedstawiają również widmo DFT i widmo ciągle sygnału dyskretnego. Wybór pulsacji początkowej, kroku pulsacji i liczby prążków widma jest w CZT dowolny, można np. wyznaczyć tylko częstotliwości harmoniczne wybranej częstotliwości podstawowej.



Rys.8.15 Schemat blokowy obliczeń CZT,  $\omega_k = \omega_0 + k\Delta\omega$ ,  $k = 0, 1, \dots, M-1$ ,  $W = e^{-j\Delta\omega}$ ,

$$h[n] = \begin{cases} W^{-n^2/2}, & -(N-1) \leq n \leq M-1 \\ 0 & \text{w przeciwnym przypadku} \end{cases}.$$



Rys.8.16 Przykłady próbkowania osi częstotliwości za pomocą CZT.

### 8.2.6 Algorytmy FFT

Szybka transformata Fouriera, oznaczana jako FFT (*Fast Fourier Transform*), jest to metoda obliczania współczynników DFT. Wyznaczanie DFT ze wzoru definycyjnego ma złożoność obliczeniową rzędu  $N^2$ . Algorytmy FFT pozwalają zredukować złożoność obliczeniową wyznaczania współczynników DFT do  $N \log_2 N$ . Zysk obliczeniowy stosowania FFT rośnie wraz ze wzrostem długości sygnału  $N$ , np. dla sygnału o długości  $N=2^{10}=1024$ , policzenie DFT ze wzoru definycyjnego wymaga wykonania  $N^2=2^{20}=1048576$  zespolonych mnożeń, a przy zastosowaniu algorytmu FFT tylko  $N \log_2 N=10240$  zespolonych mnożeń. Zysk obliczeniowy w tym przypadku wynosi więc dwa rzędy wielkości, co oznacza, że obliczenia zostaną wykonane 100 razy szybciej i np. zużyją 100 razy mniej energii w urządzeniu przenośnym.

Wprowadzając  $W_N = e^{-j(2\pi/N)}$ , równania DFT (8.13) i (8.14) można zapisać w postaci:

$$X[k] = \sum_{n=0}^{N-1} x[n] W_N^{kn}, \quad 0 \leq n \leq N-1, \quad 0 \leq k \leq N-1, \quad (8.20)$$

$$x[n] = \frac{1}{N} \sum_{k=0}^{N-1} X[k] W_N^{-kn}, \quad 0 \leq n \leq N-1, \quad 0 \leq k \leq N-1. \quad (8.21)$$

Algorytmy redukujące złożoność obliczeniową wyznaczania współczynników DFT wykorzystują symetrię i okresowość funkcji bazowych  $W_N^{kn}$ :

$$W_N^{k[N-n]} = W_N^{-kn} = (W_N^{kn})^* - \text{zespolona, sprzężona symetria}, \quad (8.22)$$

$$W_N^{kn} = W_N^{k(n+N)} = W_N^{(k+N)n} - \text{okresowość z } k \text{ i } n. \quad (8.23)$$

#### 8.2.6.1 FFT z podziałem z czasie (DIT, decimation in time)

Rozważmy ciąg  $x[n]$  o długości  $N$  równej całkowitej potędze dwójki, tzn.  $N=2^v$ ,  $v=1,2,3\dots$  (jeżeli warunek ten nie jest spełniony, ciąg  $x[n]$  może zostać uzupełniony zerami do wymaganej długości). Dla ciągu  $x[n]$  DFT jest zdefiniowane jako:

$$X[k] = \sum_{n=0}^{N-1} x[n] W_N^{kn}, \quad 0 \leq n \leq N-1, \quad 0 \leq k \leq N-1. \quad (8.24)$$

Rozdzielimy obliczenia dla wyrazów o indeksach parzystych (*even*) i nieparzystych (*odd*):

$$X[k] = \sum_{even} x[n] W_N^{kn} + \sum_{odd} x[n] W_N^{kn}, \quad (8.25)$$

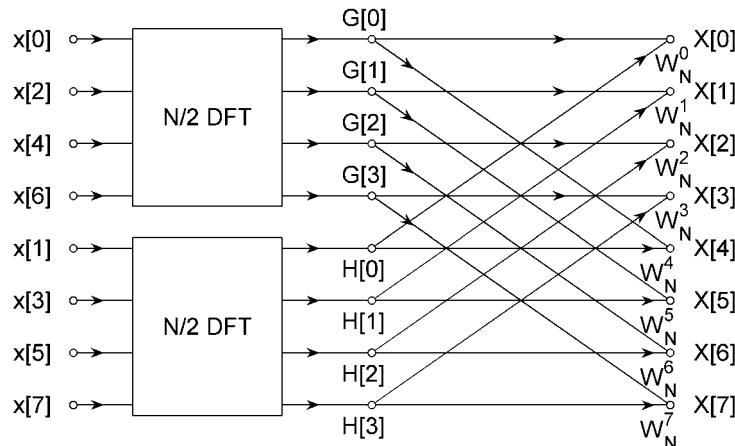
co można również zapisać (wykorzystując podstawienie  $n=2r$  dla próbek o indeksach parzystych i  $n=2r+1$  dla próbek o indeksach nieparzystych) następująco:

$$X[k] = \sum_{r=0}^{N/2-1} x[2r] W_N^{2rk} + \sum_{r=0}^{N/2-1} x[2r+1] W_N^{(2r+1)k} = \sum_{r=0}^{N/2-1} x[2r] (W_N^2)^r W_N^{rk} + W_N^k \sum_{r=0}^{N/2-1} x[2r+1] (W_N^2)^r W_N^{rk}. \quad (8.26)$$

Ponieważ  $W_N^2 = W_{N/2}$  ( $W_N^2 = e^{-2j(2\pi/N)} = e^{-j2\pi/(N/2)} = W_{N/2}$ ), otrzymujemy:

$$X[k] = \sum_{r=0}^{N/2-1} x[2r] W_{N/2}^{rk} + W_N^k \sum_{r=0}^{N/2-1} x[2r+1] W_{N/2}^{rk} = G[k] + W_N^k H[k], \quad 0 \leq k \leq N-1. \quad (8.27)$$

Zgodnie z (8.27),  $N$  punktowa DFT została przedstawiona jako suma dwóch  $N/2$  punktowych DFT (dla próbek o indeksach parzystych i nieparzystych). Chociaż zakres  $k$  wynosi  $0 \leq k \leq N-1$ , obliczenia wystarczy przeprowadzić dla  $0 \leq k \leq N/2-1$  ze względu na okresowość  $G[k]$  i  $H[k]$  z okresem  $N/2$ , a następnie odpowiednio złożyć wyniki, co pokazano na rys.8.17 dla sygnału o długości  $N=8$  próbek.



Rys.8.17 Dekompozycja  $N$  punktowej DFT na sumę dwóch  $N/2$  punktowych DFT wg (8.27), dla  $N=8$ .

Taki sam podział obliczeń można zastosować do wyznaczenia  $G[k]$  i  $H[k]$  w (8.27), tzn. wyznaczyć DFT  $N/2$  punktowe przez dwa DFT  $N/4$  punktowe itd. aż do DFT 2 punktowych.

Rys.8.18 przedstawia graf przepływowego obliczeń FFT z podziałem w czasie, dla  $N=8$ . Ze względu na podział obliczeń osobno dla próbek o indeksach parzystych i nieparzystych, sygnał wejściowy  $x[n]$  nie jest w kolejności naturalnej tylko w tzw. porządku o odwróconej kolejności bitów, natomiast jego widmo  $X[k]$  jest w porządku naturalnym.

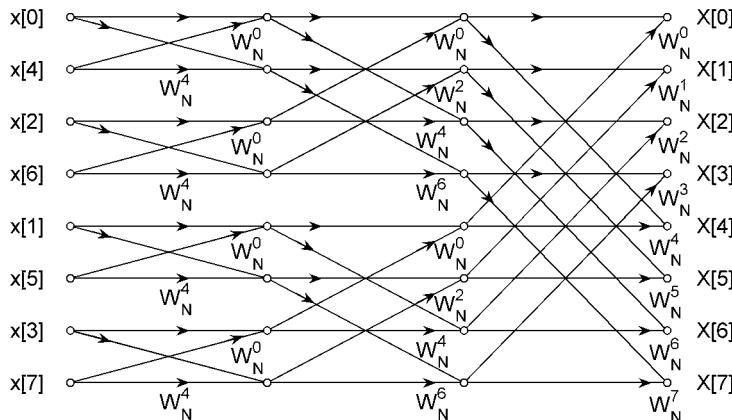
Porządek o odwróconej kolejności bitów uzyskuje się zapisując indeks próbki w postaci binarnej, a następnie odwracając go (odbijając lustrzanie) i odczytując nowy indeks. Jako przykład rozważmy sygnał o długości  $N=8$ , do reprezentacji indeksów tego sygnału potrzeba  $\log_2 N = 3$  bitów. Poniżej przedstawione są sygnały w porządku naturalnym i o odwróconej kolejności bitów:

Porządek naturalny:

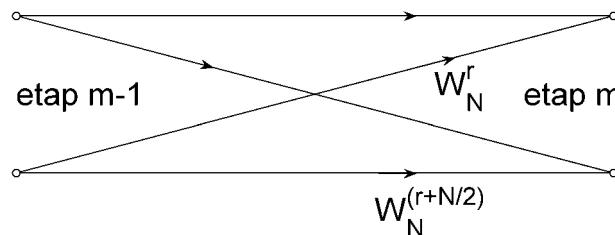
$$\begin{aligned}x[0] &= x[(0\ 0\ 0)] \\x[1] &= x[(0\ 0\ 1)] \\x[2] &= x[(0\ 1\ 0)] \\x[3] &= x[(0\ 1\ 1)] \\x[4] &= x[(1\ 0\ 0)] \\x[5] &= x[(1\ 0\ 1)] \\x[6] &= x[(1\ 1\ 0)] \\x[7] &= x[(1\ 1\ 1)]\end{aligned}$$

Porządek o odwróconej kolejności bitów:

$$\begin{aligned}x[(0\ 0\ 0)] &= x[0] \\x[(1\ 0\ 0)] &= x[4] \\x[(0\ 1\ 0)] &= x[2] \\x[(1\ 1\ 0)] &= x[6] \\x[(0\ 0\ 1)] &= x[1] \\x[(1\ 0\ 1)] &= x[5] \\x[(0\ 1\ 1)] &= x[3] \\x[(1\ 1\ 1)] &= x[7]\end{aligned}$$



Rys.8.18 Graf przepływowego obliczeń FFT z podziałem w czasie, dla  $N=8$ .



Rys.8.19 Obliczenia motylkowe.

Obliczenia w grafie z rys.8.18 wykonywane są w podstawowej strukturze pokazanej na rys.8.19, są to tzw. obliczenia motylkowe.

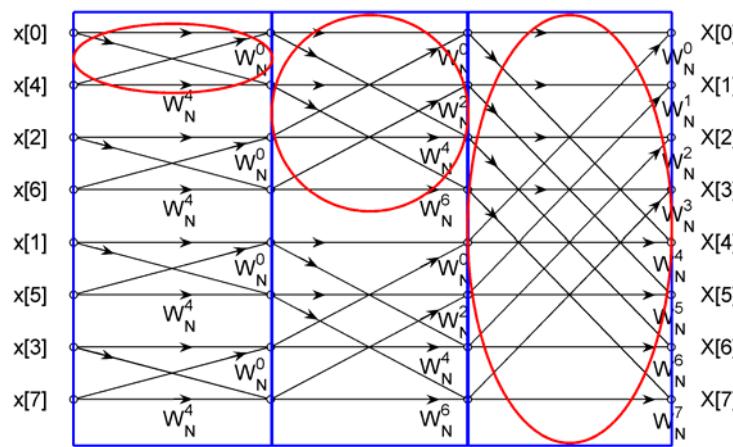
Obliczenia FFT przebiegają w etapach zaznaczonych prostokątami na rys.8.20. Na każdym etapie występują bloki motylków zaznaczone elipsami na rys.8.20. Na kolejnych etapach obliczeń maleje liczba bloków, a liczba motylków pozostaje taka sama, chociaż zmieniają się ich rozmiary.

Na kolejnych etapach obliczeń  $N$  liczb zespolonych jest, poprzez obliczenia motylkowe, zamieniane na  $N$  liczb zespolonych. Są to tzw. obliczenia z podstawianiem, które nie wymagają dodatkowej pamięci, tzn. po ostatnim etapie obliczeń współczynniki  $X[k]$  zajmują komórki pamięci, w których znajdował się sygnał wejściowy  $x[n]$ .

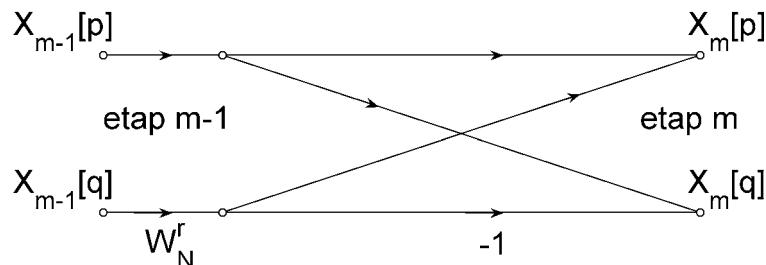
Ponieważ zachodzi  $W_N^{N/2} = e^{-j(2\pi/N)N/2} = e^{-j\pi} = -1$ , więc mnożnik  $W_N^{(r+N/2)}$  w grafie na rys.8.19 można zapisać w postaci  $W_N^{r+N/2} = W_N^{N/2}W_N^r = -W_N^r$ , co prowadzi do uproszczonych obliczeń motylkowych pokazanych na rys.8.21, w których wykonywane jest tylko jedno mnożenie zespolone (zamiast dwóch). Graf z rys.8.21 opisują równania:

$$\begin{aligned} X_m[p] &= X_{m-1}[p] + W_N^r X_{m-1}[q] \\ X_m[q] &= X_{m-1}[p] - W_N^r X_{m-1}[q], \end{aligned} \quad (8.28)$$

gdzie:  $m=1, 2, \dots, v$  oznacza etap obliczeń ( $N=2^v$ ,  $v=1, 2, 3, \dots$ ), a  $p, q, r$  są parametrami motylka.

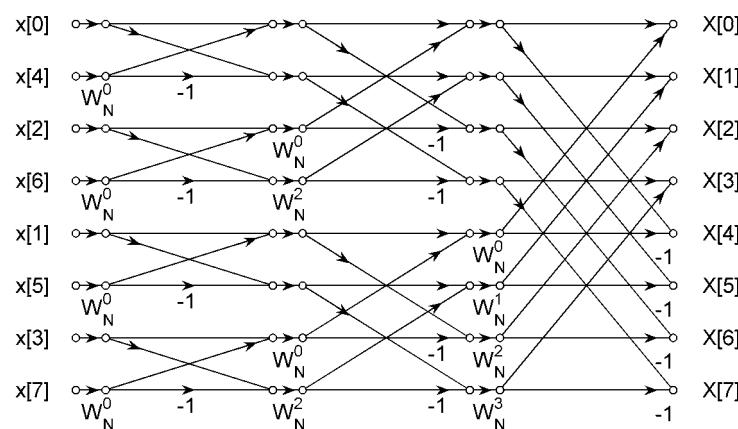


Rys. 8.20 Graf przepływowego obliczeń FFT z podziałem w czasie, dla  $N=8$ , z zaznaczonymi etapami obliczeń (prostokąty) i blokami motylków (elipsy).



Rys. 8.21 Uproszczone obliczenia motylkowe, FFT z podziałem w czasie.

Rys. 8.22 przedstawia graf przepływowego obliczeń FFT z podziałem w czasie, dla  $N=8$ , z zastosowaniem uproszczonych obliczeń motylkowych. Implementacja algorytmu FFT z podziałem w czasie wymaga zastosowania równań (8.28) wyliczanych wewnętrz pętli, iterowanych po liczbie motylków w bloku, po liczbie bloków na etapie i po liczbie etapów.



Rys. 8.22 Graf przepływowego obliczeń FFT z podziałem w czasie, dla  $N=8$ , z zastosowaniem uproszczonych obliczeń motylkowych.

### 8.2.6.2 FFT z podziałem z częstotliwości (DIF, *decimation in frequency*)

Rozważmy ciąg  $x[n]$  o długości  $N$  równej całkowitej potędze dwójki, tzn.  $N=2^v$ ,  $v=1,2,3\dots$ . Dla ciągu  $x[n]$  DFT jest zdefiniowane jako

$$X[k] = \sum_{n=0}^{N-1} x[n] W_N^{kn}, \quad 0 \leq n \leq N-1, \quad 0 \leq k \leq N-1. \quad (8.29)$$

Obliczmy osobno współczynniki Fouriera o indeksach parzystych i nieparzystych. Dla indeksów parzystych:

$$X[2r] = \sum_{n=0}^{N-1} x[n] W_N^{n(2r)}, \quad 0 \leq r \leq N/2-1, \quad (8.30)$$

co można zapisać w postaci:

$$X[2r] = \sum_{n=0}^{N/2-1} x[n] W_N^{2nr} + \sum_{n=N/2}^{N-1} x[n] W_N^{2nr}. \quad (8.31)$$

Po zamianie sumowania w drugiej sumie otrzymujemy:

$$X[2r] = \sum_{n=0}^{N/2-1} x[n] W_N^{2nr} + \sum_{n=0}^{N/2-1} x[n+N/2] W_N^{2r(n+N/2)}, \quad (8.32)$$

ponieważ:  $W_N^{2r(n+N/2)} = W_N^{2rn} W_N^{rN} = W_N^{2rn}$  i  $W_N^2 = W_{N/2}$ , otrzymujemy:

$$X[2r] = \sum_{n=0}^{N/2-1} (x[n] + x[n+N/2]) W_{N/2}^{nr}, \quad 0 \leq r \leq N/2-1. \quad (8.33)$$

Równanie (8.33) jest  $N/2$  punktowym DFT ciągu utworzonego z sumy pierwszej i drugiej połowy ciągu  $x[n]$ . Współczynniki Fouriera o indeksach nieparzystych wyznacza się następująco:

$$X[2r+1] = \sum_{n=0}^{N-1} x[n] W_N^{n(2r+1)}, \quad 0 \leq r \leq N/2-1, \quad (8.34)$$

$$X[2r+1] = \sum_{n=0}^{N/2-1} x[n] W_N^{n(2r+1)} + \sum_{n=N/2}^{N-1} x[n] W_N^{n(2r+1)}. \quad (8.35)$$

Drugą sumę w (8.35) można przedstawić w postaci ( $W_N^{(N/2)2r} = 1$  i  $W_N^{N/2} = -1$ ):

$$\sum_{n=N/2}^{N-1} x[n] W_N^{n(2r+1)} = \sum_{n=0}^{N/2-1} x[n+N/2] W_N^{(n+N/2)(2r+1)} = W_N^{(N/2)(2r+1)} \sum_{n=0}^{N/2-1} x[n+N/2] W_N^{n(2r+1)} = - \sum_{n=0}^{N/2-1} x[n+N/2] W_N^{n(2r+1)}. \quad (8.36)$$

Ostatecznie:

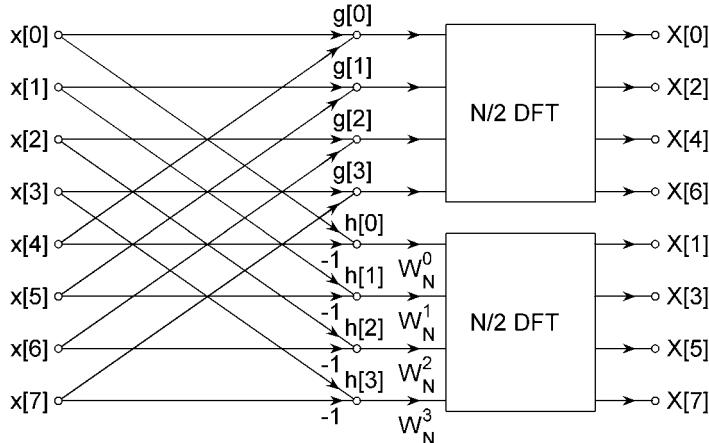
$$X[2r+1] = \sum_{n=0}^{N/2-1} (x[n] - x[n+N/2]) W_N^{n(2r+1)}, \quad (8.37)$$

lub uwzględniając  $W_N^2 = W_{N/2}$ :

$$X[2r+1] = \sum_{n=0}^{N/2-1} (x[n] - x[n+N/2]) W_N^n W_{N/2}^{nr}. \quad (8.38)$$

Równanie (8.38) jest  $N/2$  punktowym DFT ciągu utworzonego z różnicą pierwszej i drugiej połowy ciągu  $x[n]$  przemnożonej przez  $W_N^n$ .

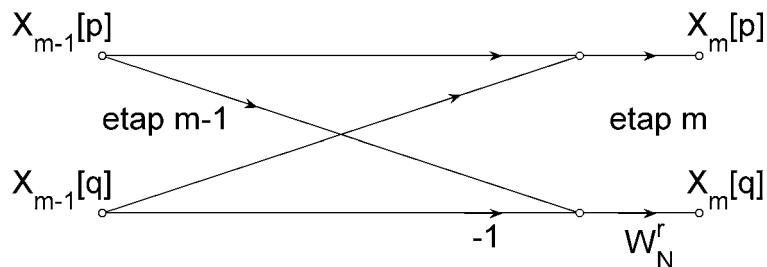
Uwzględniając wyrażenia na  $X[2r]$  (8.33) i  $X[2r+1]$  (8.38) otrzymujemy algorytm dekompozycji  $N$  punktowego DFT na dwa  $N/2$  punktowe DFT przedstawiony na rys.8.23.



Rys.8.23 Dekompozycja  $N$  punktowej DFT na sumę dwóch  $N/2$  punktowych DFT wg (8.33) i (8.38), dla  $N=8$ .

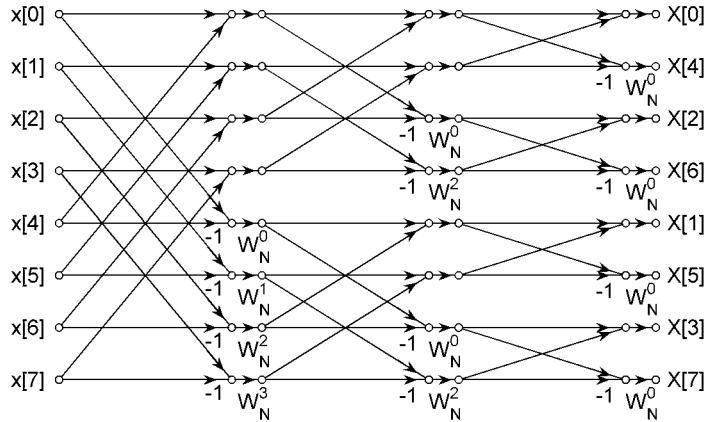
Dekompozycję przedstawioną na rys.8.23 można powtarzać, aż do uzyskania 2 punktowych DFT. Struktura uproszczonej obliczeń motylkowych dla FFT z podziałem w częstotliwości przedstawiona jest na rys.8.24, jest ona opisana równaniami:

$$\begin{aligned} X_m[p] &= X_{m-1}[p] + X_{m-1}[q] \\ X_m[q] &= (X_{m-1}[p] - X_{m-1}[q]) W_N^r. \end{aligned} \quad (8.39)$$



Rys.8.24 Uproszczone obliczenia motylkowe, FFT z podziałem w częstotliwości.

Rys.8.25 przedstawia graf przepływowego obliczeń FFT z podziałem w częstotliwości, dla  $N=8$ , z zastosowaniem uproszczonej obliczeń motylkowych. Dane wejściowe  $x[n]$  są w porządku naturalnym, a współczynniki DFT  $X[k]$  w porządku z odwróconą kolejnością bitów.



Rys.8.25 Graf przepływowego obliczeń FFT z podziałem w częstotliwości, dla  $N=8$ , z zastosowaniem uproszczonych obliczeń motylkowych.

### 8.2.7 DFT sygnałów o wartościach rzeczywistych

Danymi wejściowymi dla algorytmów DFT są liczby zespolone. Jednak często wyznacza się widmo liczb rzeczywistych (np. z wyjścia układu AC), wówczas korzystając z symetrii DFT można wyznaczyć:

- 1) jednym  $N$  punktowym DFT widma dwóch sygnałów rzeczywistych o długości  $N$ ,
- 2) jednym  $N$  punktowym DFT widmo sygnału rzeczywistego o długości  $2N$ .

Ad.1

Rozważmy dwa ciągi o wartościach rzeczywistych  $x_1[n]$  i  $x_2[n]$  i długościach  $N$ , ich widma DFT wynoszą  $X_1[k]$  i  $X_2[k]$ . Utwórzmy ciąg zespolony  $x[n]$ , którego część rzeczywistą i urojoną stanowią ciągi  $x_1[n]$  i  $x_2[n]$ :

$$x[n] = x_1[n] + jx_2[n]. \quad (8.40)$$

Wyznaczając DFT (liczone przez FFT) ciągu  $x[n]$  otrzymujemy:

$$X[k] = \sum_{n=0}^{N-1} (x_1[n] + jx_2[n]) e^{-j(2\pi/N)kn} = \sum_{n=0}^{N-1} x_1[n] e^{-j(2\pi/N)kn} + j \sum_{n=0}^{N-1} x_2[n] e^{-j(2\pi/N)kn} = X_1[k] + jX_2[k]. \quad (8.41)$$

Obliczmy  $X[N-k]$ :

$$X[N-k] = \sum_{n=0}^{N-1} x[n] e^{-j(2\pi/N)(Nn-kn)} = \sum_{n=0}^{N-1} x[n] e^{-j2\pi n} e^{j(2\pi/N)kn} = \sum_{n=0}^{N-1} x[n] e^{j(2\pi/N)kn}. \quad (8.42)$$

Podstawiając ciąg zespolony  $x[n] = x_1[n] + jx_2[n]$  oraz sprzężając  $X[N-k]$ , otrzymujemy:

$$X^*[N-k] = \sum_{n=0}^{N-1} (x_1[n] - jx_2[n]) e^{-j(2\pi/N)kn} = X_1[k] - jX_2[k]. \quad (8.43)$$

Dodając i odejmując  $X[k]$  (8.41) i  $X^*[N-k]$  (8.43), otrzymujemy zależności na widma części rzeczywistej i urojonej ciągu  $x[n]$ :

$$X_1[k] = \frac{X[k] + X^*[N-k]}{2}, \quad X_2[k] = \frac{X[k] - X^*[N-k]}{2j}. \quad (8.44)$$

Równania (8.44) nie mogą być wyznaczone przy użyciu DFT dla  $k=0$ , ponieważ wymagałoby to znajomości  $X[N]$ , a algorytm DFT wylicza współczynniki  $X[k]$  dla  $k = 0, 1, \dots, N-1$ . Ze względu na okresowość względem  $k$   $X[N]=X[0]$  i wówczas:

$$X_1[k=0] = \frac{X[0] + X^*[0]}{2} = \operatorname{Re}\{X[0]\}, \quad X_2[k=0] = \frac{X[0] - X^*[0]}{2j} = \operatorname{Im}\{X[0]\}. \quad (8.45)$$

### Ad.2

Rozważmy ciąg  $x[n]$  o wartościach rzeczywistych i długości  $N$ .  $N$  punktowe DFT takiego ciągu

$$X[k] = \sum_{n=0}^{N-1} x[n] W_N^{kn}, \quad 0 \leq n \leq N-1, \quad 0 \leq k \leq N-1 \quad (8.46)$$

można przedstawić w postaci dwóch  $N/2$  punktowych DFT (8.27):

$$X[k] = \sum_{r=0}^{N/2-1} x[2r] W_{N/2}^{rk} + W_N^k \sum_{r=0}^{N/2-1} x[2r+1] W_{N/2}^{rk} = G[k] + W_N^k H[k], \quad (8.47)$$

gdzie  $G[k]$  jest to DFT próbek  $x[n]$  o indeksach parzystych, a  $H[k]$  jest to DFT próbek  $x[n]$  o indeksach nieparzystych. Do wyznaczenia obu tych widm można zastosować algorytm z Ad.1.

## 9. Analiza sygnałów z wykorzystaniem DFT

DFT jest często stosowane do analizy sygnałów ciągłych (analogowych). Przy analizie sygnałów ciągłych za pomocą DFT niezbędne są następujące etapy przetwarzania:

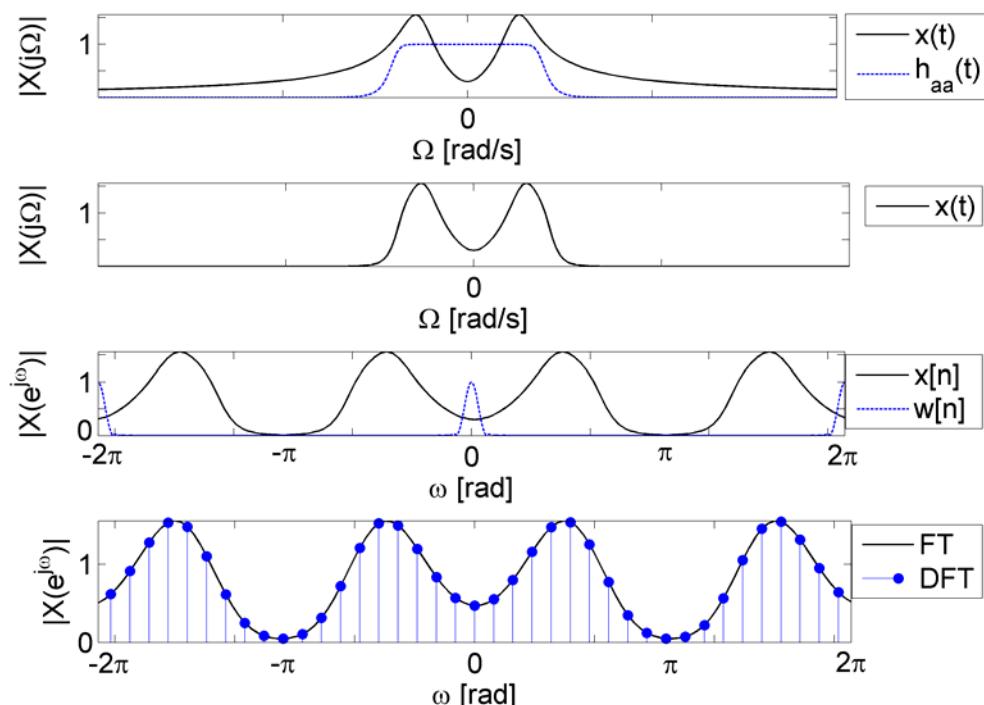
- 1) filtracja antialiasingowa i próbkowanie sygnału,
- 2) okienkowanie sygnału dyskretnego  $x[n]$ , tzn. mnożenie przez współczynniki okna  $w[n] = w[n]x[n]$ , w wyniku którego zauważamy splot widma sygnału z widmem okna:

$$V(e^{j\omega}) = \frac{1}{2\pi} \int_{-\pi}^{\pi} X(e^{j\Theta})W(e^{j(\omega-\Theta)})d\Theta, \quad (9.1)$$

co powoduje rozmycie (wygładzenie) widma sygnału.

- 3) próbkowanie ciągłego widma sygnału dyskretnego  $X(e^{j\omega})$  przez DFT, tj. wyznaczanie wartości  $X(e^{j\omega})$  dla pulsacji dyskretnych  $\omega = (2\pi/N)k$ .

Etapy 1-3 zmieniają oryginalne widmo sygnału analogowego, co oznacza, że obserwujemy widmo sygnału zniekształcone przez zastosowane narzędzia przetwarzania sygnałów.



Rys.9.1 Analiza DFT sygnału ciągłego, od góry: widmo sygnału analogowego  $x(t)$  i charakterystyka amplitudowa filtra antialiasingowego  $h_{aa}(t)$ , widmo sygnału analogowego o ograniczonym paśmie, widmo sygnału dyskretnego  $x[n]$  i widmo okna czasowego, ciągle widmo okienkowanego sygnału dyskretnego i jego widmo DFT.

Rys.9.1 przedstawia widma analizowanego sygnału na poszczególnych etapach przetwarzania. Przedmiotem analizy jest nieograniczone widmo, tzn. dla  $\Omega \in (-\infty, +\infty)$ , sygnału analogowego  $x(t)$ . Przed próbkowaniem widmo to musi zostać ograniczone filtrem antialiasingowym. Po próbkowaniu widmo sygnału dyskretnego  $x[n]$  jest okresowe. DFT można wyznaczyć tylko dla ciągu o skończonej długości, co jest równoważne z okienkowaniem sygnału. Jeżeli nie zadajemy wprost żadnego okna, to sygnał jest mnożony przez okno prostokątne. W wyniku tego mnożenia obserwujemy splot widma sygnału z

widmem okna, a nie widmo samego sygnału. Powoduje to rozmycie (*smear*) widma sygnału. Widmo DFT próbuje ciągle widmo sygnału dyskretnego w częstotliwościach  $\omega=(2\pi/N)k$ .

## 9.1 Analiza DFT sygnałów sinusoidalnych

Transformata Fouriera ciągu sinusoidalnego  $x[n] = \cos(\omega_0 n + \phi)$  składa się z impulsów Diraca występujących okresowo w pulsacjach  $\omega_0$ :

$$X(e^{j\omega}) = \sum_{k=-\infty}^{\infty} [\pi e^{j\phi} \delta(\omega - \omega_0 + 2\pi k) + \pi e^{-j\phi} \delta(\omega + \omega_0 + 2\pi k)]. \quad (9.2)$$

Okienkowanie sygnału w dziedzinie czasu powoduje rozmycie (zwiększenie szerokości) reprezentacji Fouriera (widma). Długość okna  $N$  wpływa na gęstość próbkowania widma sygnału, ponieważ współczynniki DFT wyznaczane są z krokiem  $2\pi/N$ .

Rozważmy sygnał ciągły w postaci sumy dwóch harmonicznych:

$$s_c(t) = A_0 \cos(\Omega_0 t + \Theta_0) + A_1 \cos(\Omega_1 t + \Theta_1), \quad -\infty < t < \infty. \quad (9.3)$$

W wyniku idealnego próbkowania otrzymujemy sygnał dyskretny ( $T$  - okres próbkowania):

$$x[n] = A_0 \cos(\omega_0 n + \Theta_0) + A_1 \cos(\omega_1 n + \Theta_1), \quad \omega_0 = \Omega_0 T, \quad \omega_1 = \Omega_1 T, \quad -\infty < n < \infty. \quad (9.4)$$

Okienkowanie opisujemy operacją mnożenia:

$$v[n] = w[n]x[n] = A_0 w[n] \cos(\omega_0 n + \Theta_0) + A_1 w[n] \cos(\omega_1 n + \Theta_1). \quad (9.5)$$

Ciąg  $v[n]$  można przedstawić w postaci zespolonej:

$$v[n] = \frac{A_0}{2} w[n] e^{j\Theta_0} e^{j\omega_0 n} + \frac{A_0}{2} w[n] e^{-j\Theta_0} e^{-j\omega_0 n} + \frac{A_1}{2} w[n] e^{j\Theta_1} e^{j\omega_1 n} + \frac{A_1}{2} w[n] e^{-j\Theta_1} e^{-j\omega_1 n}, \quad (9.6)$$

a następnie wyznaczyć jego widmo korzystając z twierdzenia o przesunięciu (rozdział 2.8.5):

$$V(e^{j\omega}) = \frac{A_0}{2} e^{j\Theta_0} W(e^{j(\omega-\omega_0)}) + \frac{A_0}{2} e^{-j\Theta_0} W(e^{j(\omega+\omega_0)}) + \frac{A_1}{2} e^{j\Theta_1} W(e^{j(\omega-\omega_1)}) + \frac{A_1}{2} e^{-j\Theta_1} W(e^{j(\omega+\omega_1)}). \quad (9.7)$$

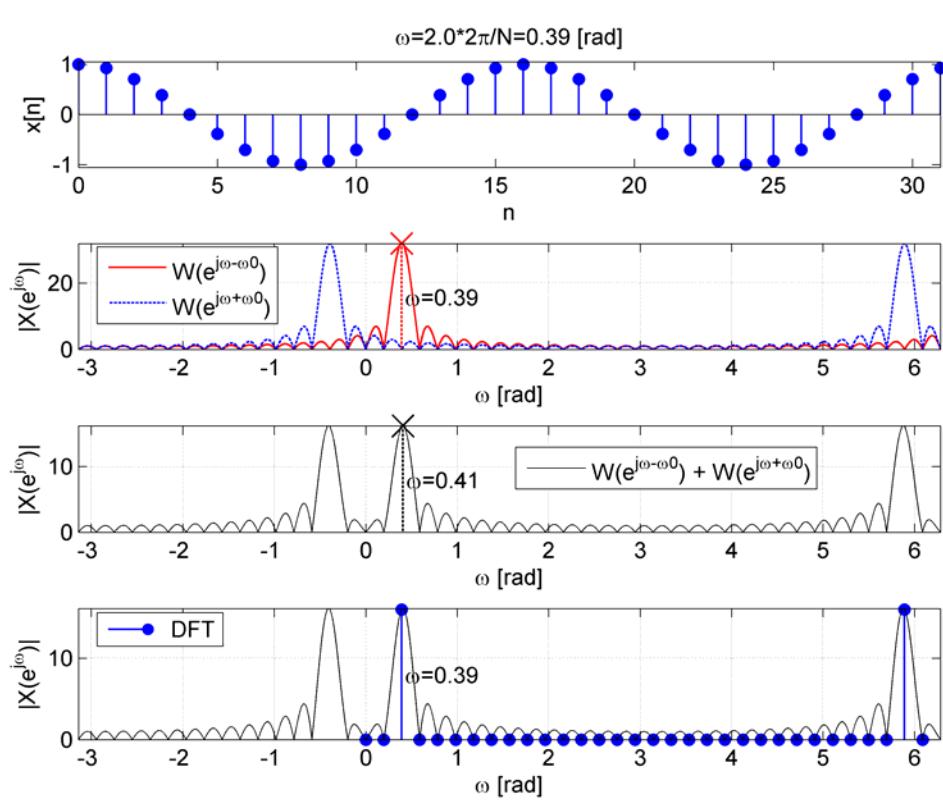
Zgodnie z (9.7) transformata Fouriera dyskretnego, okienkowanego sygnału sumy harmonicznych składa się z kopii widm Fouriera okna powtarzanych w częstotliwościach  $\pm\omega_0, \pm\omega_1$  i przeskalowanych przez zespolone amplitudy poszczególnych harmonicznych, tzn. że w częstotliwościach  $\pm\omega_0, \pm\omega_1$  obserwujemy widmo okna, a nie sygnału.

Jako pierwszy przykład widma DFT rozważmy pojedynczy sygnał sinusoidalny przedstawiony na rys.9.2-9.4. Rys.9.2 przedstawia sygnał dyskretny o częstotliwości  $\omega=2(2\pi/N)$  i długości  $N=32$  próbki. Częstotliwość sygnału jest całkowitą (dwukrotną) krotnością kroku częstotliwości DFT. Oznacza to, że DFT jest obliczane z całkowitej liczby okresów sygnału, jest to tzw. próbkowanie synchroniczne. Rys.9.2 (drugi od góry) przedstawia moduły zespolonych składowych teoretycznego widma sygnału sinusoidalnego wg (9.7), widmo sygnału jest sumą tych składowych. Oba widma są widmami okna czasowego, w tym przypadku okna prostokątnego, i składają się z tzw. listka głównego i listków bocznych. Pożądane jest, aby listek główny był jak najwęższy, a listki boczne jak

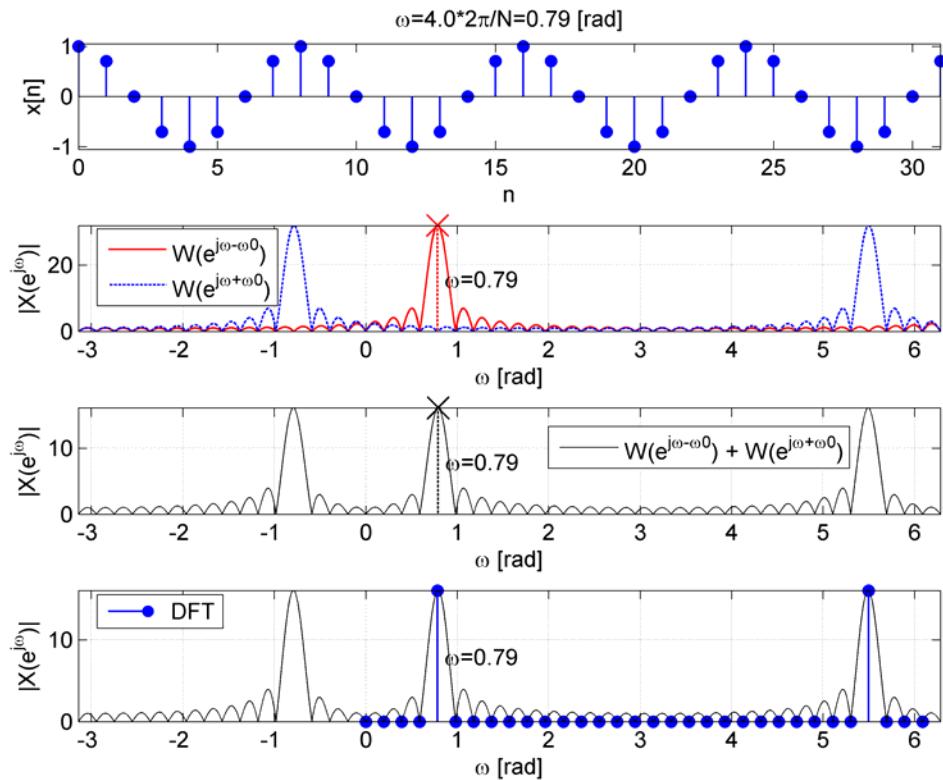
najmniejsze. Transformacja Fouriera zachowuje energię sygnału, tzn., że energia sygnału w dziedzinie czasu i częstotliwości jest taka sama. Listki boczne zawierają energię sygnału, która rozlewa się w całym zakresie częstotliwości, jest to tzw. przeciek widmowy (*spectral leakage*). Przeciek widmowy może w sposób znaczący zakłócić widmo sygnału dyskretnego. Ilustracją wpływu przecieku widmowego na widmo amplitudowe sygnału jest rys.9.2 (trzeci od góry). Widmo sygnału powstało w wyniku sumowania widm składowych, ponieważ listki główne są położone blisko siebie, więc zakłóciły się one wzajemnie swoimi listkami bocznymi. Skutkiem tego zjawiska jest maksimum widma przesunięte z 0.39 rad do 0.41 rad, pojawił się więc błąd analizy częstotliwościowej. Widmo DFT próbkuje oś częstotliwości w punktach  $(2\pi/N)k$  i dla próbkowania synchronicznego estymacja częstotliwości jest dokładna.

Rys.9.3 przedstawia kolejny przykład próbkowania synchronicznego sygnału, tzn., za czas równy całkowitej liczbie jego okresów. W tym przypadku listki główne są od siebie oddalone na tyle, że nie zakłócają się w sposób widoczny i na podstawie widma sygnału można prawidłowo określić jego częstotliwość jako 0.79 rad.

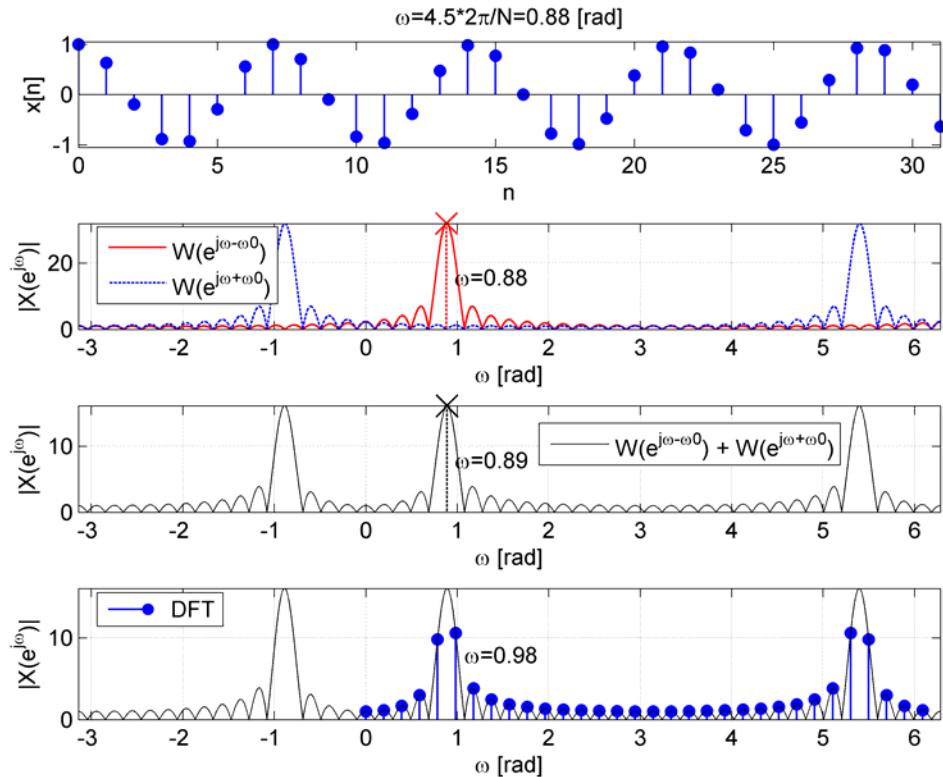
Rys.9.4 przedstawia przykład niesynchronicznego próbkowania sygnału, tzn. częstotliwość sygnału nie jest całkowitą krotnością  $2\pi/N$ . W takim przypadku przeciek widma powoduje błąd pomiaru częstotliwości, amplitudy i fazy. Ponieważ widmo DFT próbkuje oś częstotliwości tylko w punktach  $(2\pi/N)k$ , więc prążek widma dla częstotliwość sygnału, która w przykładzie wynosi  $4.5(2\pi/N)$ , nie jest wyznaczany przez DFT, co może prowadzić do znaczących błędów przy interpretacji widma DFT.



Rys.9.2 Próbkowanie synchroniczne, listki główne położone blisko siebie; od góry: sygnał dyskretny, składowe widma teoretycznego wg (9.7), widmo ciągłe sygnału dyskretnego i widmo DFT.



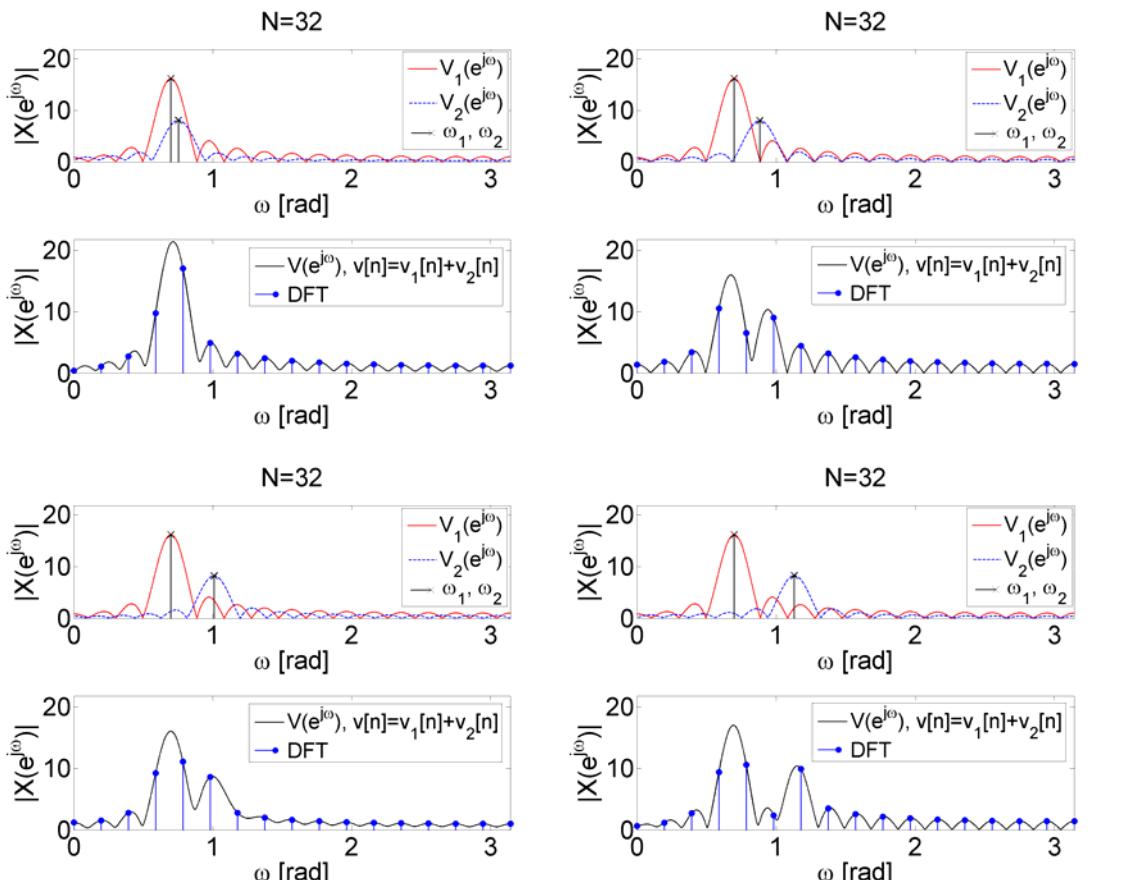
Rys.9.3 Próbkowanie synchroniczne, listki główne oddalone od siebie; od góry: sygnał dyskretny, składowe widma teoretycznego wg (9.7), widmo ciągłe sygnału dyskretnego i widmo DFT.



Rys.9.4 Próbkowanie niesynchroniczne; od góry: sygnał dyskretny, składowe widma teoretycznego wg (9.7), widmo ciągłe sygnału dyskretnego i widmo DFT.

W przykładach na rys.9.2-9.4 analizowany sygnał sinusoidalny zakłócał sam siebie. W przypadku, gdy sygnał składa się z sumy przebiegów sinusoidalnych sygnały składowe dodatkowo zakłócają się wzajemnie.

Rys.9.5 przedstawia wyniki analizy widmowej sumy dwóch sygnałów sinusoidalnych. W przypadku, gdy różnica pomiędzy częstotliwościami sygnałów jest mniejsza niż szerokość listka głównego, występują bardzo duże błędy analizy widmowej i oba sygnały sinusoidalne mogą być nieroróżnicalne. W przypadku, kiedy oba sygnały są od siebie oddalone w częstotliwości i mogą być rozróżnione, analiza widmowa jest nadal obarczona przeciekiem widmowym, który przesuwa listki główne, oraz próbkowaniem widma przez DFT.



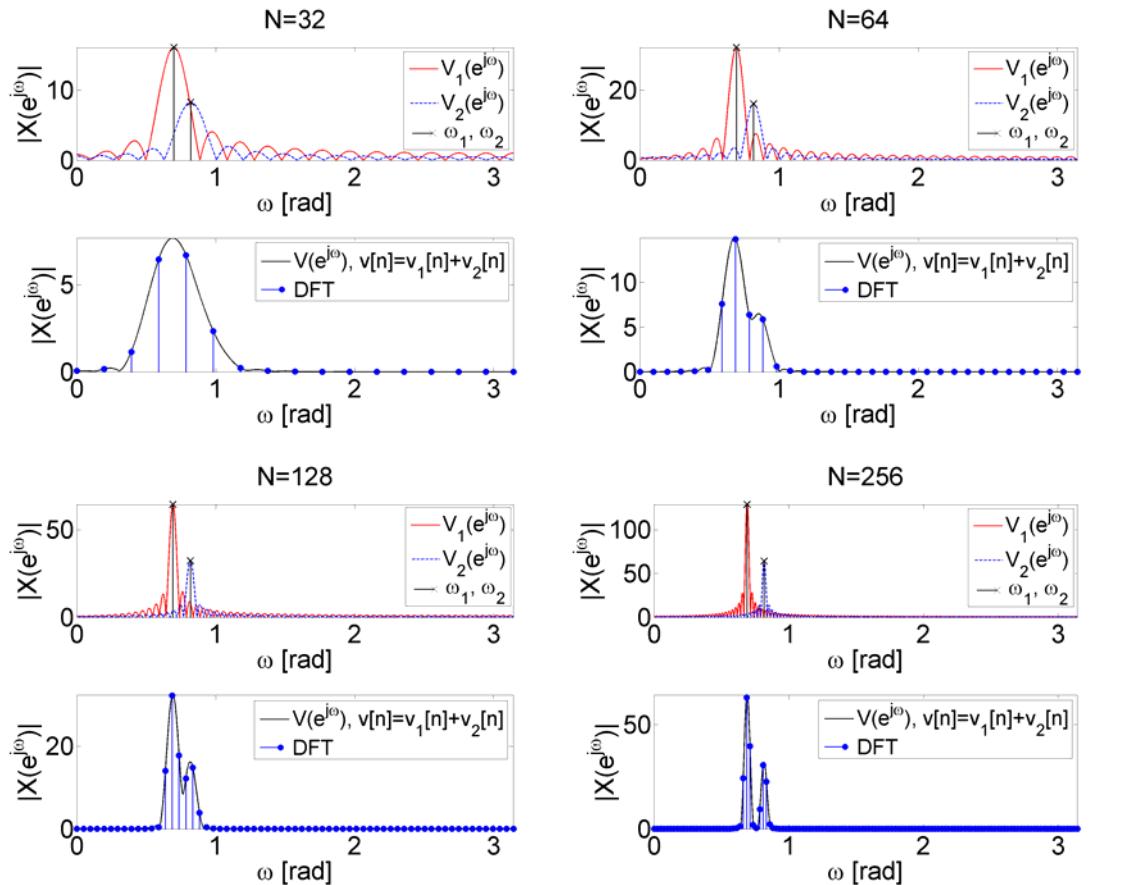
Rys.9.5 Ilustracja przecieku widmowego w przypadku analizy sumy dwóch sygnałów sinusoidalnych.

Rozdzielcość częstotliwościową analizy widmowej, rozumianą jako zdolność do rozróżnienia dwóch blisko siebie położonych częstotliwości w widmie sygnału, można zwiększyć przez wydłużenie okna czasowego, tj. zwiększenie liczby próbek sygnału  $N$ .

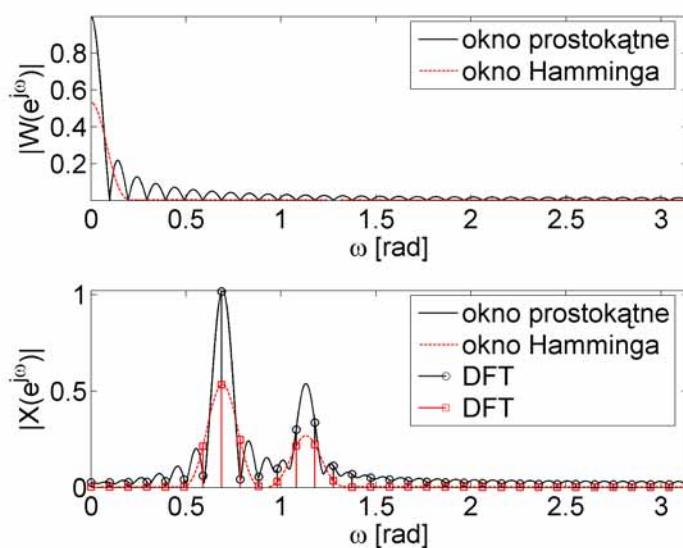
Rys.9.6 przedstawia widma sumy dwóch sygnałów sinusoidalnych dla różnych długości okna prostokątnego. Dla sygnału o długości  $N=32$  próbki częstotliwości sygnałów nie są rozróżnialne ze względu na zbyt szerokie listki główne. Wydłużenie okna czasowego powoduje proporcjonalne zawężenie listka głównego jego widma. Dla  $N=256$  listki główne są na tyle wąskie, że pozwalają rozróżnić w widmie sygnału dwie częstotliwości.

Przeciek widmowy, tzn. energia listków bocznych, może być ograniczany przez stosowanie okien czasowych innych niż prostokątne. Okno prostokątne ma najwyższe położone listki boczne i jednocześnie najwęższy listek główny. Inne okna czasowe mają niższe położone listki boczne za cenę poszerzenia listka głównego. Przykład zastosowania okna Hamminga do analizy DFT sygnału jest przedstawiony na rys.9.7. Rys.9.7 przedstawia widmo okna prostokątnego i okna Hamminga oraz widmo DFT i widmo ciągłe sygnału

złożonego z sumy dwóch sinusoid przemnożonego przez każde z tych okien. Jak widać z rys.9.7, poziom listków bocznych jest wyraźnie ograniczony przez zastosowanie okna Hamminga, ale listek główny jest szerszy. Zawężenie listka głównego, a więc poprawę rozdzielczości częstotliwościowej, można uzyskać przez analizę dłuższego fragmentu sygnału, co pokazano na rys.9.6.



Rys.9.6 Zwiększenie rozdzielczości analizy widmowej przez zastosowanie dłuższej obserwacji sygnału.



Rys.9.7 Zmniejszenie przecieku widmowego przez zastosowanie okna Hamminga.

Podsumowując, dwa główne skutki stosowania okna czasowego to:

1. zmniejszona rozdzielczość częstotliwościowa i
2. przeciek widmowy.

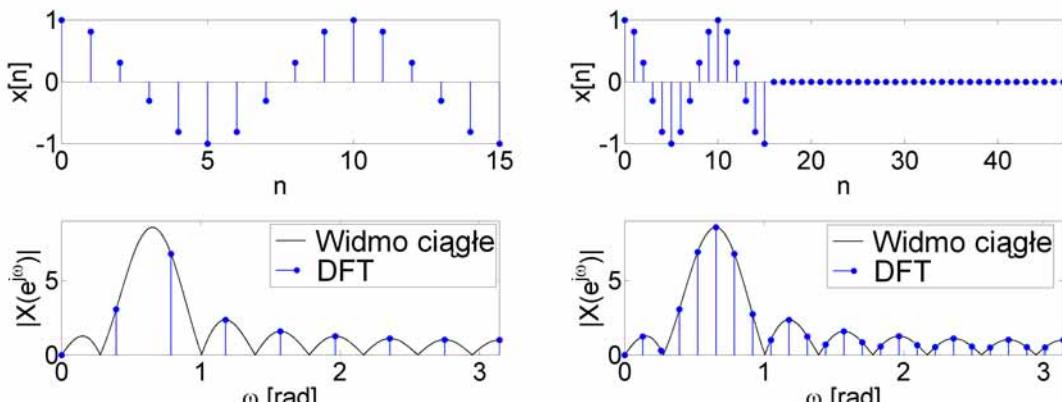
Rozdzielczość częstotliwościowa jest ograniczona przez szerokość listka głównego widma okna, natomiast przeciek widmowy przez poziom listków bocznych widma okna (w stosunku do listka głównego).

Szerokość listka głównego widma okna zależy od długości okna w czasie (im dłuższe okno tym węższy listek główny), natomiast położenie listków bocznych zależy od rodzaju okna (i nie zależy od jego długości).

Okno prostokątne jest oknem o najwęższym listku głównym i najmniejszym tłumieniu listków bocznych (13 dB).

W celu redukcji przecieku widmowego stosuje się okna inne niż prostokątne (podobnie jak przy projektowaniu filtrów FIR), w szczególności parametryczne okno Kaisera.

Długość okna czasowego określa długość analizowanego fragmentu sygnału  $N$ , a tym samym krok próbkowania osi częstotliwości w widmie DFT. Rys.9.8 ilustruje zwiększenie gęstości próbkowania osi częstotliwości w widmie DFT przez dołączenie wektora zer na końcu sygnału. Wraz ze wzrostem  $N$  maleje krok próbkowania osi częstotliwości  $(2\pi/N)k$ . Dołączenie zer nie poprawia rozdzielczości częstotliwościowej DFT.



Rys.9.8 Zwiększenie gęstości próbkowania osi częstotliwości przez DFT.

## 9.2 Interpolowane DFT

W celu redukcji błędów analizy częstotliwościowej z zastosowaniem DFT spowodowanych niesynchronicznym próbkowaniem sygnału stosowane są algorytmy interpolowanego DFT. Zadanie interpolacji DFT jest zilustrowane na rys.9.9. Z widma DFT sygnału sinusoidalnego wybierane są prążki o największych modułach, na ich podstawie należy określić przesunięcie  $\delta$  korygujące częstotliwość  $\omega_k$  do poprawnej częstotliwości sygnału  $\omega_0$ . Częstotliwość mierzonego sygnału sinusoidalnego wynosi:

$$\omega_0 = (k \pm \delta) \frac{2\pi}{N}, \quad 0 < \delta \leq 0.5. \quad (9.8)$$

Prążek DFT z indeksem  $k$  ma największy moduł. Jeżeli drugi, co do wielkości modułu, prążek DFT ma indeks  $k+1$  to w nawiasie (9.8) występuje znak '+', jest to przypadek przedstawiony na rys.9.9. Jeżeli drugi, co do wielkości modułu, prążek DFT ma indeks  $k-1$  to w nawiasie (9.8) występuje znak '-'.

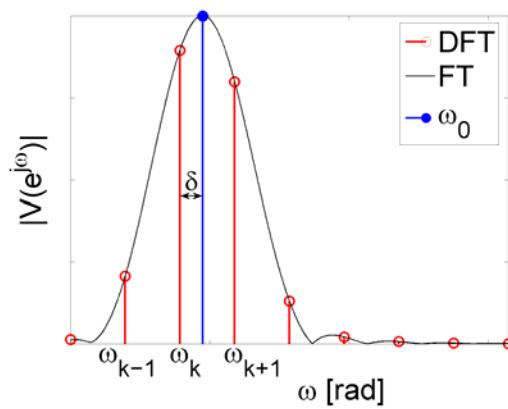
Wyznaczając stosunek modułów dwóch największych prążków DFT i korzystając z zależności  $\omega_k = \omega_0 - \delta 2\pi / N$ ,  $\omega_{k+1} = \omega_0 - \delta 2\pi / N + 2\pi / N$  zilustrowanych na rys.9.9, otrzymujemy:

$$\frac{|V[k+1]|}{|V[k]|} = \frac{|V(\omega_{k+1})|}{|V(\omega_k)|} = \frac{|V(\omega_0 - \delta 2\pi/N + 2\pi/N)|}{|V(\omega_0 - \delta 2\pi/N)|}, \quad (9.9)$$

gdzie  $v[n]=w[n]x[n]$ ,  $V[k]=\text{DFT}\{v[n]\}$ ,  $W[k]=\text{DFT}\{w[n]\}$ ,  $x[n]$  - analizowany sygnał sinusoidalny,  $w[n]$  - okno czasowe. Ponieważ w częstotliwości  $\omega_0$  obserwujemy widmo okna przesunięte z częstotliwości  $\omega=0$ , więc warunek (9.9) jest równoważny warunkowi (9.10) w postaci:

$$\frac{|V[k+1]|}{|V[k]|} \approx \frac{|W(-\delta 2\pi/N + 2\pi/N)|}{|W(-\delta 2\pi/N)|}. \quad (9.10)$$

W (9.10) zastosowano znak przybliżenia ' $\approx$ ' a nie równości, ponieważ kształt widma okienkowanego sygnału  $V[k]$  może się nieznacznie różnić od kształtu widma okna  $W[k]$  ze względu na przeciek widmowy.



Rys.9.9 Ilustracja problemu interpolowanego DFT.

Zależność (9.10) jest podstawą dwupunktowych algorytmów interpolacji DFT. Analityczne rozwiązanie (9.10) ze względu na  $\delta$  jest możliwe tylko dla niektórych okien czasowych. W przypadku braku rozwiązania analitycznego stosowana jest interpolacja wielomianowa lub iteracyjne wyszukiwanie.

Dla interpolacji trzypunktowej korektę  $\delta$  wyznacza się z następującego stosunku modułów prążków DFT:

$$\frac{|V[k]| + |V[k+1]|}{|V[k]| + |V[k-1]|} = \frac{|V(\omega_0 - \delta 2\pi/N)| + |V(\omega_0 - \delta 2\pi/N + 2\pi/N)|}{|V(\omega_0 - \delta 2\pi/N)| + |V(\omega_0 - \delta 2\pi/N - 2\pi/N)|}. \quad (9.11)$$

Podobnie jak w (9.10) stosunek modułów prążków widma sygnału sinusoidalnego jest równoważny stosunkowi modułów widma dla  $\omega_0=0$ :

$$\frac{|V[k]| + |V[k-1]|}{|V[k]| + |V[k+1]|} \approx \frac{|W(-\delta 2\pi/N)| + |W(-\delta 2\pi/N - 2\pi/N)|}{|W(-\delta 2\pi/N)| + |W(-\delta 2\pi/N + 2\pi/N)|}. \quad (9.12)$$

Rozważmy dyskretny sygnał sinusoidalny z oknem prostokątnym o długości  $N$ :

$$w_p[n] = \begin{cases} 1, & 0 \leq n < N \\ 0, & 0 > n \geq N \end{cases}. \quad (9.13)$$

Widmo okna prostokątnego jest następujące:

$$W_p(e^{j\omega}) = e^{-j\omega(N-1)/2} \frac{\sin(\omega N/2)}{\sin(\omega/2)}. \quad (9.14)$$

Zależność (9.10) przyjmuje więc postać:

$$\frac{|V[k+1]|}{|V[k]|} = \left| \frac{\sin(-\delta\pi + \pi)}{\sin(-\delta\pi/N + \pi/N)} \right| \left| \frac{\sin(-\delta\pi/N)}{\sin(-\delta\pi)} \right|. \quad (9.15)$$

Dla małych wartości argumentów funkcji sinus w (9.15), funkcje te mogą być zastąpione ich argumentami:

$$\frac{|V[k+1]|}{|V[k]|} \approx \left| \frac{\delta\pi/N}{-\delta\pi/N + \pi/N} \right| = \left| \frac{\delta}{-\delta + 1} \right|. \quad (9.16)$$

Z (9.16)

$$\delta \approx \frac{|V[k+1]|}{|V[k+1]| + |V[k]|}. \quad (9.17)$$

Częstotliwość sygnału wynosi, więc:

$$\omega_0 \approx \left( k \pm \frac{|V[k+1]|}{|V[k+1]| + |V[k]|} \right) \frac{2\pi}{N}. \quad (9.18)$$

Moduł prążka widma dla częstotliwości  $\omega_0$  można wyznaczyć z proporcji:

$$\frac{|V(\omega_0)|}{|V(\omega_k)|} \approx \frac{|W_p(0)|}{|W_p(\delta 2\pi/N)|} = N \left| \frac{\sin(\delta\pi/N)}{\sin(\delta\pi)} \right| \approx \left| \frac{\delta\pi}{\sin(\delta\pi)} \right|. \quad (9.19)$$

Z (9.19) otrzymujemy:

$$|V(\omega_0)| \approx |V[k]| \left| \frac{\delta\pi}{\sin(\delta\pi)} \right| = |V[k]| \frac{\pi}{\sin(\delta\pi)} \delta. \quad (9.20)$$

Fazę prążka widma dla częstotliwości  $\omega_0$  można wyznaczyć z zależności:

$$\arg\{V(\omega_0)\} - \arg\{V(\omega_k)\} \approx \arg\{W_p(0)\} - \arg\{W_p(\delta 2\pi/N)\} = -\arg\{e^{-j\delta(\pi/N)(N-1)}\}. \quad (9.21)$$

Z (9.21) otrzymujemy:

$$\arg\{V(\omega_0)\} \approx \arg\{V(\omega_k)\} \pm \arg\{e^{-j\delta(\pi/N)(N-1)}\}. \quad (9.22)$$

Znak '+' lub '-' w (9.22) wybiera się tak samo jak w (9.8).

Rozważmy dyskretny sygnał sinusoidalny z oknem Hanninga o długości  $N$ . Okresowe (periodic) okno Hanninga (Hanna) zdefiniowane jest następująco:

$$w_H[n] = \begin{cases} 0.5 - 0.5 \cos\left(\frac{2\pi}{N}n\right), & 0 \leq n < N, \\ 0, & \text{w przeciwnym przypadku.} \end{cases} \quad (9.23)$$

Częstotliwość dyskretna funkcji kosinus w (9.23) wynosi  $2\pi/N$  i jest równa krokowi próbkowania osi częstotliwości przez DFT. Dla porównania symetryczne (*symmetric*) okno Hanninga, stosowane do projektowania filtrów FIR metodą okien, jest zbudowane z funkcji kosinus o częstotliwości  $2\pi/(N-1)$ .

Okno w postaci (9.23) można interpretować jako sumę funkcji stałej i funkcji kosinus przemnożoną przez okno prostokątne, dlatego widmo tego okna jest splotem impulsów Diraca występujących w częstotliwościach  $-2\pi/N$ ,  $0$  i  $2\pi/N$  z widmem okna prostokątnego. Wynikiem tego splotu jest suma trzech widm:

$$W_H(e^{j\omega}) = -0.25W_p(e^{j(\omega-\omega_1)}) + 0.5W_p(e^{j\omega}) - 0.25W_p(e^{j(\omega+\omega_1)}), \quad \omega_1 = 2\pi/N, \quad (9.24)$$

gdzie  $W_p(e^{j\omega})$  jest widmem okna prostokątnego (9.14).

Widmo DFT sygnału z oknem Hanninga można uzyskać na dwa sposoby: 1) licząc DFT sygnału przemnożonego prze to okno, lub 2) licząc DFT sygnału z oknem prostokątnym, a następnie licząc splot dyskretny zespolonych prążków DFT z filtrem  $H=[-0.25, 0.5, -0.25]$ . Okno Hanninga zyskało dużą popularność ze względu na łatwość implementacji. Współczynniki tego okna są wagami dwójki i mnożenia mogą być realizowane jako przesunięcia bitów w prawo, a wartości funkcji kosinus nie muszą być osobno przechowywane w pamięci, ponieważ są takie same jak funkcje bazowe FFT.

Podstawiając (9.14) do (9.24) otrzymujemy widmo okna Hanninga:

$$\begin{aligned} W_H(e^{j\omega}) &= -0.25e^{-j(\omega-2\pi/N)(N-1)/2} \frac{\sin((\omega-2\pi/N)\cdot N/2)}{\sin((\omega-2\pi/N)/2)} \\ &\quad + 0.5e^{-j\omega(N-1)/2} \frac{\sin(\omega\cdot N/2)}{\sin(\omega/2)} \\ &\quad - 0.25e^{-j(\omega+2\pi/N)(N-1)/2} \frac{\sin((\omega+2\pi/N)\cdot N/2)}{\sin((\omega+2\pi/N)/2)}. \end{aligned} \quad (9.25)$$

Po przekształceniach oraz przyjmując przybliżenie  $e^{j(\pi/N)(N-1)} \approx -1 + j\pi/N$  otrzymujemy:

$$W_H(e^{j\omega}) \approx e^{-j\omega(N-1)/2} \left( 0.25(-1 + j\pi/N) \frac{\sin(\omega N/2)}{\sin(\omega/2 - \pi/N)} + 0.5 \frac{\sin(\omega N/2)}{\sin(\omega/2)} + 0.25(-1 - j\pi/N) \frac{\sin(\omega N/2)}{\sin(\omega/2 + \pi/N)} \right). \quad (9.26)$$

Zaniedbując część urojoną w nawiasie (9.26) otrzymujemy wyrażenie na moduł widma okna Hanninga:

$$|W_H(e^{j\omega})| \approx \left| -0.25 \frac{\sin(\omega N/2)}{\sin(\omega/2 - \pi/N)} + 0.5 \frac{\sin(\omega N/2)}{\sin(\omega/2)} - 0.25 \frac{\sin(\omega N/2)}{\sin(\omega/2 + \pi/N)} \right| \quad (9.27)$$

Wyznaczmy moduły widma okna Hanninga dla częstotliwości podanych w (9.10).

$$\begin{aligned} |W_H(-\delta 2\pi/N)| &\approx \left| -0.25 \frac{\sin(-\delta\pi)}{\sin(-\delta\pi/N - \pi/N)} + 0.5 \frac{\sin(-\delta\pi)}{\sin(-\delta\pi/N)} - 0.25 \frac{\sin(-\delta\pi)}{\sin(-\delta\pi/N + \pi/N)} \right| \approx \\ &\approx \left| \frac{0.25}{-\delta - 1} - \frac{0.5}{-\delta} + \frac{0.25}{-\delta + 1} \frac{|\sin(\delta\pi)|}{\pi/N} \right|. \end{aligned} \quad (9.28)$$

$$\begin{aligned} |W_H(-\delta 2\pi/N + 2\pi/N)| &\approx \left| -0.25 \frac{\sin(-\delta\pi + \pi)}{\sin(-\delta\pi/N)} + 0.5 \frac{\sin(-\delta\pi + \pi)}{\sin(-\delta\pi/N + \pi/N)} - 0.25 \frac{\sin(-\delta\pi + \pi)}{\sin(-\delta\pi/N + 2\pi/N)} \right| \approx \\ &\approx \left| \frac{0.25}{-\delta} + \frac{0.5}{-\delta + 1} - \frac{0.25}{-\delta + 2} \frac{|\sin(\delta\pi)|}{\pi/N} \right|. \end{aligned} \quad (9.29)$$

Podstawiając (9.28) i (9.29) do (9.10) otrzymujemy:

$$\frac{|V[k+1]|}{|V[k]|} \approx \frac{\left| \frac{0.25}{\delta} - \frac{0.5}{\delta-1} + \frac{0.25}{\delta-2} \right|}{\left| -\frac{0.25}{\delta+1} + \frac{0.5}{\delta} - \frac{0.25}{\delta-1} \right|}. \quad (9.30)$$

Sprawdzając do wspólnego mianownika ułamki w (9.30) otrzymujemy:

$$\frac{|V[k+1]|}{|V[k]|} \approx \frac{\left| \delta^2(2 \cdot 0.25 - 0.5) + \delta(2 \cdot 0.5 - 4 \cdot 0.25) + 2 \cdot 0.25 \right|}{\delta(\delta-1)(\delta-2)} \frac{\delta(\delta-1)(\delta+1)}{\delta^2(0.5 - 2 \cdot 0.25) - 0.5}. \quad (9.31)$$

Jak widać wartości współczynników okna Hanninga zerują współczynniki przy potegach  $\delta$  (9.31), w wyniku czego otrzymujemy łatwą do rozwiązania zależność:

$$\frac{|V[k+1]|}{|V[k]|} \approx \left| \frac{\delta+1}{\delta-2} \right|, \quad (9.32)$$

z której:

$$\delta \approx \frac{2|V[k+1]| - |V[k]|}{|V[k]| + |V[k+1]|}. \quad (9.33)$$

Częstotliwość sygnału wynosi, więc (9.8):

$$\omega_0 \approx \left( k \pm \frac{2|V[k+1]| - |V[k]|}{|V[k+1]| + |V[k]|} \right) \frac{2\pi}{N}. \quad (9.34)$$

Moduł prążka widma dla częstotliwości  $\omega_0$  można wyznaczyć analogicznie do (9.19):

$$|V(\omega_0)| \approx |V[k]| \frac{|W_H(0)|}{|W_H(\delta 2\pi/N)|} \approx |V[k]| \frac{0.5\pi}{\sin(\delta\pi)} \left| -\frac{0.25}{\delta-1} + \frac{0.5}{\delta} - \frac{0.25}{\delta+1} \right|. \quad (9.35)$$

Również fazę prążka widma dla częstotliwości  $\omega_0$  można wyznaczyć analogicznie do zależności (9.21), korzystając z (9.26) i zaniedbując część urojoną w nawiasie (9.26):

$$\arg\{V(\omega_0)\} \approx \arg\{V(\omega_k)\} \pm \arg\{e^{-j\delta(\pi/N)(N-1)}\}. \quad (9.36)$$

Jak widać zależności korygujące fazę dla okna prostokątnego (9.22) i Hanninga (9.36) są takie same, z tą różnicą, że są w nich wykorzystywane współczynniki korekcji  $\delta$  wyznaczone odpowiednio przez (9.17) i (9.33).

Okno prostokątne i okno Hanninga należą do klasy czasowych okien kosinusowych w postaci:

$$w[n] = \begin{cases} \sum_{m=0}^M (-1)^m A[m] \cos\left(\frac{2\pi}{N} m n\right), & 0 \leq n < N, \\ 0, & 0 > n \geq N. \end{cases} \quad (9.37)$$

Zdefiniujmy następującą sumę:

$$S_{k+n} = \left| \sum_{m=-M}^M \frac{(-1)^{m+n} A'[m]}{\delta - (m+n)} \right|, \quad \begin{cases} \text{dla } m=0 \ A'[m] = A[0], \\ \text{w przeciwnym przypadku } A'[m] = A[|m|]/2. \end{cases} \quad (9.38)$$

Należy zwrócić uwagę, że współczynniki  $A'[m]$  w (9.38), za wyjątkiem  $A[0]$ , są dzielone przez 2.

Dla okien (9.37) zależność (9.10) przyjmuje postać:

$$\frac{|V[k+1]|}{|V[k]|} \approx \frac{|W(-\delta 2\pi/N + 2\pi/N)|}{|W(-\delta 2\pi/N)|} = \frac{S_{k+1}}{S_k}, \quad (9.39)$$

a zależność (9.12):

$$\frac{|V[k]| + |V[k-1]|}{|V[k]| + |V[k+1]|} \approx \frac{|W(-\delta 2\pi/N)| + |W(-\delta 2\pi/N - 2\pi/N)|}{|W(-\delta 2\pi/N)| + |W(-\delta 2\pi/N + 2\pi/N)|} \approx \frac{S_k + S_{k-1}}{S_k + S_{k+1}}. \quad (9.40)$$

Współczynniki  $A[m]$  okien czasowych, dla których, analogicznie do okna prostokątnego i Hanniga, można wyprowadzić formuły interpolacyjne, zebrane są w Tab.9.1, dla kilku pierwszych wartości  $M$ . Dla  $M=0$  otrzymujemy okno prostokątne, a dla  $M=1$  okno Hanninga (Hanna). Dla unormowania współczynników  $A[m]$  do  $A[0]=1$ , wartość średnia okna wynosi 1, nie trzeba więc skalować prążka DFT w celu określenia amplitudy sygnału.

Okna w postaci (9.37) ze współczynnikami z Tab.9.1 są nazywane w literaturze oknami Rifea-Vincenta klasy I lub oknami typu  $\cos^\alpha(X)$  dla  $\alpha=0,2,4,6\dots$  zdefiniowanymi jako:

$$w[n] = \begin{cases} \sin^\alpha \left( \frac{\pi}{N} n \right), & 0 \leq n < N, \\ 0, & 0 > n \geq N. \end{cases} \quad (9.41)$$

Tab.9.1 Współczynniki  $A[m]$  dla okien Rifea-Vincenta klasy I (9.37).

	$m=0$	$m=1$	$m=2$	$m=3$	$m=4$	$m=5$	$m=6$
$A[m], M=0$	1						
$A[m], M=1$	1	1					
$A[m], M=2$	1	4/3	1/3				
$A[m], M=3$	1	3/2	3/5	1/10			
$A[m], M=4$	1	8/5	4/5	8/35	1/35		
$A[m], M=5$	1	105/63	60/63	45/126	5/63	1/126	
$A[m], M=6$	1	396/231	495/462	110/231	33/231	6/231	1/462

Dla okien Rifea-Vincenta klasy I z (9.39) otrzymujemy (interpolacja dwupunktowa):

$$\delta \approx \frac{(M+1)|V[k+1]| - M|V[k]|}{|V[k]| + |V[k+1]|}. \quad (9.42)$$

Dla  $M=0$  i  $M=1$  zależność (9.42) jest zgodna z wyprowadzonymi uprzednio zależnościami (9.17) i (9.33) dla okna prostokątnego i okna Hanninga.

Częstotliwość analizowanego sygnału sinusoidalnego wynosi:

$$\omega_0 \approx \left( k \pm \frac{(M+1)|V[k+1]| - M|V[k]|}{|V[k+1]| + |V[k]|} \right) \frac{2\pi}{N}. \quad (9.43)$$

Dla okien Rifea-Vincenta klasy I z (9.40) otrzymujemy (interpolacja trzypunktowa):

$$\delta \approx (M+1) \frac{|V[k+1]| - |V[k-1]|}{2|V[k]| + |V[k-1]| + |V[k+1]|}, \quad (9.44)$$

i

$$\omega_{03P} \approx \left( k \pm (M+1) \frac{|V[k+1]| - |V[k-1]|}{2|V[k]| + |V[k-1]| + |V[k+1]|} \right) \frac{2\pi}{N}. \quad (9.45)$$

Przy wyprowadzaniu formuły interpolacyjnej dla interpolacji trzypunktowej zakłada się, że daleki przeciek widmowy jest, w przybliżeniu, taki sam dla sąsiednich prążków DFT, a w kolejnych prążkach zmienia znak. W zależnościach (9.44) i (9.45) wpływ dalekiego przecieku widmowego kompensuje się więc w liczniku i mianowniku, dzięki czemu błędy systematyczne metody są mniejsze niż w przypadku interpolacji dwupunktowej.

Podobnie jak poprzednio (9.35), moduł prążka widma dla częstotliwości  $\omega_0$  można wyznaczyć z proporcji:

$$|V(\omega_0)| \approx |V[k]| \frac{|W(0)|}{|W(\delta 2\pi/N)|} \approx \frac{2\pi}{\sin(\delta\pi)} \frac{|V[k]|}{S_k}. \quad (9.46)$$

Dla algorytmu interpolacji trzypunktowej w proporcji uwzględnia się podwojony prążek dla  $\omega_0$  w celu kompensacji dalekiego przecieku widmowego:

$$\frac{|V(\omega_0 - 2\pi/N)| + 2|V(\omega_0)| + |V(\omega_0 + 2\pi/N)|}{|V[k-1]| + 2|V[k]| + |V[k+1]|} \approx \frac{|W(-2\pi/N)| + 2|W(0)| + |W(2\pi/N)|}{|W(-\delta 2\pi/N - 2\pi/N)| + 2|W(-\delta 2\pi/N)| + |W(-\delta 2\pi/N + 2\pi/N)|}. \quad (9.47)$$

Ponieważ:

$$\begin{aligned} & |V(\omega_0 - 2\pi/N)| + 2|V(\omega_0)| + |V(\omega_0 + 2\pi/N)| = 2|V(\omega_0)| + 2|V(\omega_0 + 2\pi/N)| = 2|V(\omega_0)| \left( 1 + \frac{|V(\omega_0 + 2\pi/N)|}{|V(\omega_0)|} \right) = \\ & = 2|V(\omega_0)| \left( 1 + \frac{|W[1]|}{|W[0]|} \right) = 2|V(\omega_0)| \left( 1 + \frac{(A[1]/2)N}{A[0]N} \right) = 2|V(\omega_0)| (1 + A[1]/(2A[0])) \end{aligned} \quad (9.48)$$

$$|W(-2\pi/N)| + 2|W(0)| + |W(2\pi/N)| = 2|W(0)| + 2|W(2\pi/N)| = 2A[0]N + 2(A[1]/2)N = N(2A[0] + A[1]), \quad (9.49)$$

$$|W(-\delta 2\pi/N - 2\pi/N)| + 2|W(-\delta 2\pi/N)| + |W(-\delta 2\pi/N + 2\pi/N)| = \frac{A[0]\sin(a\pi)}{\pi/N} (S_{k-1} + 2S_k + S_{k+1}), \quad (9.50)$$

z (9.47) otrzymujemy

$$|V(\omega_0)| \approx \frac{2A[0] + A[1]}{1 + A[1]/(2A[0])} \frac{A[0]\pi}{\sin(\delta\pi)} \frac{|V[k-1]| + 2|V[k]| + |V[k+1]|}{S_{k-1} + 2S_k + S_{k+1}}, \quad (9.51)$$

i ostatecznie

$$|V(\omega_0)| \approx \frac{2\pi}{\sin(\delta\pi)} \frac{|V[k-1]| + 2|V[k]| + |V[k+1]|}{S_{k-1} + 2S_k + S_{k+1}}. \quad (9.52)$$

Jak widać, sumowanie ważonych prążków DFT w liczniku z (9.52) zaowocuje kompensacją dalekiego przecieku widmowego, podobnie jak w (9.45).

Fazę prążka widma dla częstotliwości  $\omega_0$  można wyznaczyć tak samo, jak w przypadku okna prostokątnego (9.22) i okna Hanninga (9.36):

$$\arg\{V(\omega_0)\} \approx \arg\{V(\omega_k)\} \pm \arg\{e^{-j\delta(\pi/N)(N-1)}\}. \quad (9.53)$$

Amplitudę i fazę sygnału można również wyznaczyć licząc prążek widma dla  $\omega_0$  z definicji transformacji Fouriera:

$$V(e^{j\omega_0}) = \sum_{n=0}^{N-1} v[n] e^{-j\omega_0 n}, \quad v[n] = w[n]x[n], \quad (9.54)$$

wymaga to jednak wykonania  $N$  zespolonych mnożeń i dodawań.

### 9.3 Krótkoczasowa transformacja Fouriera, spektrogram

Przy analizie DFT zakłada się, że częstotliwości występujące w sygnale są stałe w trakcie obserwacji (nie zmieniają się przez całą długość okna czasowego), jednak większość sygnałów posiada widmo częstotliwościowe zmienne w czasie (np. sygnał mowy lub sygnały telekomunikacyjne), tzn. mogą zmieniać się amplitudy, fazy i częstotliwości składowych sinusoidalnych tworzących sygnał. Krótkoczasową transformację Fouriera, oznaczaną jako STFT (*Short-Time Fourier Transform*), definiuje się następująco:

$$X[n, \lambda] = \sum_{m=-\infty}^{\infty} x[n+m]w[m]e^{-j\lambda m}, \quad (9.55)$$

gdzie  $w[n]$  oznacza wektor współczynników okna czasowego. Zgodnie z (9.55) jednowymiarowy ciąg  $x[n]$  zostaje przedstawiony jako dwuwymiarowa funkcja czasu (z  $n$  - dyskretnym) i częstotliwości (z  $\lambda$  ciągłym), co oznacza, że  $X[n, \lambda]$  są to transformaty Fouriera fragmentów sygnału  $x[n]$  wybieranych przez okno  $w[m]$  (sygnał przesuwa się przez okno).

Zmieniając kolejność sumowania w (9.55) przez podstawienie  $m' = n + m$  otrzymujemy:

$$X[n, \lambda] = \sum_{m'=-\infty}^{\infty} x[m']w[-(n-m')]e^{-j\lambda(n-m')}. \quad (9.56)$$

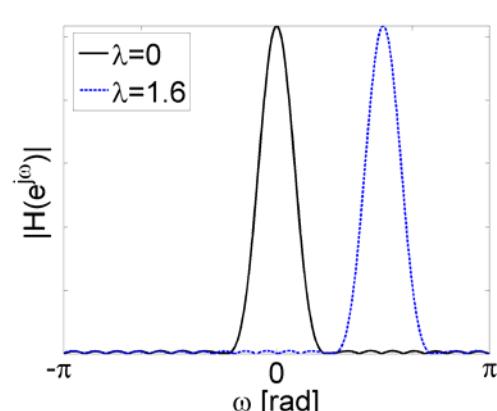
Równanie (9.56) jest splotem w postaci

$$X[n, \lambda] = x[n] * h_{\lambda}[n], \quad (9.57)$$

gdzie

$$h_{\lambda}[n] = w[-n]e^{j\lambda n} \quad (9.58)$$

jest modulowaną odpowiedzią impulsową filtra. Modulacja w dziedzinie czasu przesuwa widmo filtra w częstotliwości  $H_{\lambda}(e^{j\omega}) = W(e^{j(\lambda-\omega)})$ , co pokazano na rys.9.10.



Rys.9.10 Widmo okna i widmo filtru pasmowego w STFT.

W definicji (9.55) położenie czasowe okna jest stałe (zafiksowane), a sygnał jest przesuwany przez okno. Analogiczną reprezentację czasowo-częstotliwościową można otrzymać przesuwając okno po sygnale:

$$\check{X}[n, \lambda] = \sum_{m=-\infty}^{\infty} x[m]w[m-n]e^{-j\lambda m}. \quad (9.59)$$

Zależność pomiędzy definicjami (9.55) i (9.59) jest następująca:

$$\check{X}[n, \lambda] = e^{-j\lambda n} X[n, \lambda]. \quad (9.60)$$

Stosowanie okna w krótkoczasowej transformacie Fouriera ma na celu wybór fragmentu sygnału, w którym jego parametry są w przybliżeniu stacjonarne. Analiza sygnałów szybkozmiennych wymaga stosowania krótkich okien czasowych, co pogarsza rozdzielcość częstotliwościową. Wybór długości okna jest kompromisem pomiędzy rozdzielcością czasową i częstotliwościową analizy.

W praktyce oblicza się wartości  $X[n, \lambda]$  STFT za pomocą DFT dla okna  $w[m]=0, m < 0 \text{ i } m > L-1 \text{ i } N \geq L$ , tzn. próbuje się osiągnąć częstotliwości:

$$X[n, k] = X[n, \lambda = 2\pi k / N] = \sum_{m=0}^{L-1} x[n+m]w[m]e^{-j(2\pi/N)km}, \quad 0 \leq k \leq N-1. \quad (9.61)$$

Oznaczmy  $X[n, k]$  (9.61) jako STDFT (*Short-Time Discrete Fourier Transform*). Podobnie jak dla STFT zachodzi:

$$X[n, k] = x[n] * h_k[n], \quad 0 \leq k \leq N-1, \quad (9.62)$$

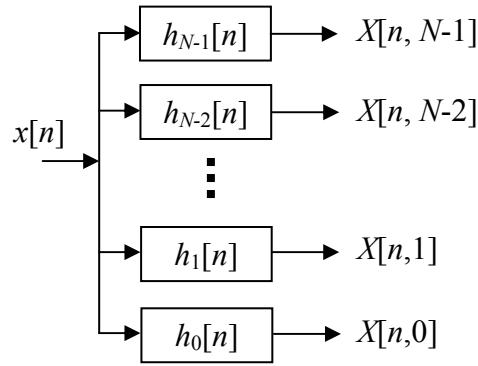
gdzie

$$h_\lambda[n] = w[-n]e^{j(2\pi/N)kn}. \quad (9.63)$$

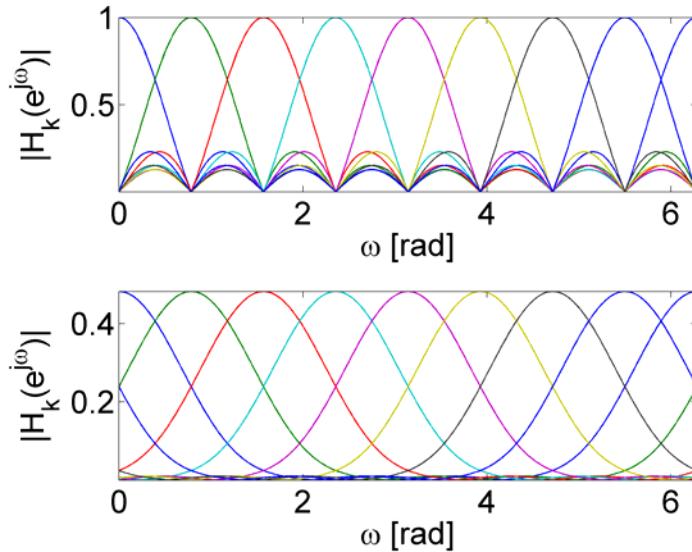
Równania (9.62) i (9.63) można interpretować jako zespół (bank)  $N$  modulowanych filtrów, których odpowiedzi częstotliwościowe są następujące:

$$H_k(e^{j\omega}) = W\left(e^{j((2\pi k/N)-\omega)}\right). \quad (9.64)$$

Interpretację STDFT w postaci zespołu filtrów przedstawia rys.9.11. Rys.9.12 przedstawia charakterystyki amplitudowe zespołu filtrów STDFT dla  $N=8$ , dla okna prostokątnego i okna Hamminga. Na rys.9.12 pasma przepustowe filtrów zachodzą na siebie, również listki boczne zachodzą na pasma przepustowe, co może być przyczyną aliasingu.



Rys.9.11 STDFT jako zespół (bank) filtrów.



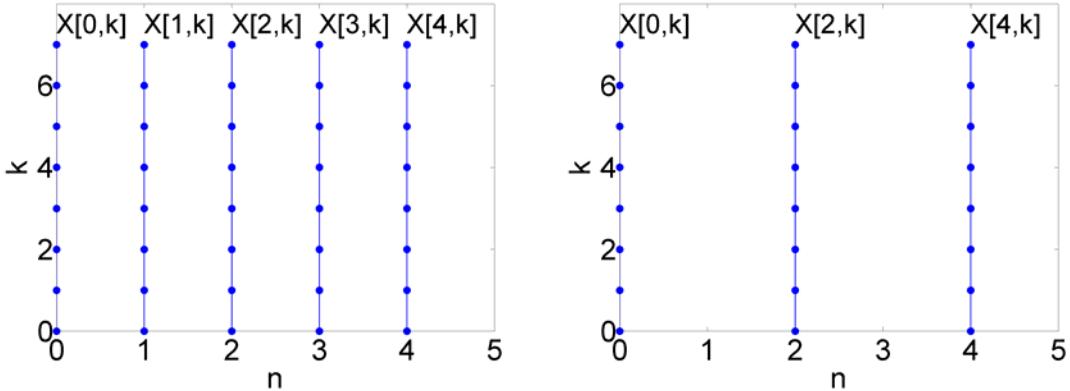
Rys.9.12 Charakterystyki amplitudowe zespołu filtrów STDFT dla okna prostokątnego (góra) i okna Hamminga (dół) dla  $N=8$ .

Wprowadźmy do definicji  $X[n,k]$  (9.61) dodatkowe oznaczenia ( $r$  - liczba całkowita, a  $R$  naturalna):

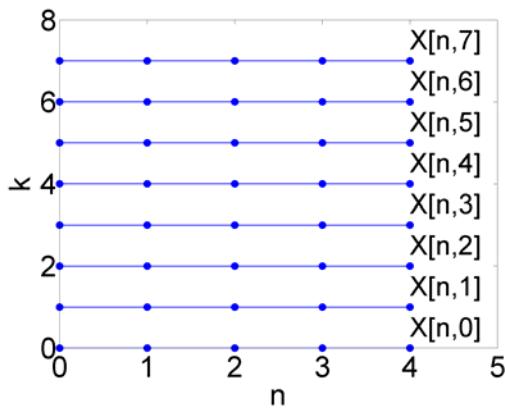
$$X[rR, k] = \sum_{m=0}^{L-1} x[rR + m]w[m]e^{-j(2\pi/N)km}. \quad (9.65)$$

Upraszczając notację  $X_r[k] = X[rR, k]$  widzimy, że próbkowana, krótkoczasowa transformata Fouriera STDFT składa się z  $N$  punktowych ciągów DFT liczonych z fragmentów sygnału  $x$  wybieranych oknem  $w$  o długości  $L$ , które przesuwa się po sygnale, co  $R$  próbek. Aby istniała możliwość rekonstrukcji sygnału, musi zachodzić  $N \geq L \geq R$ . Rys. 9.13 przedstawia interpretację graficzną próbkowania płaszczyzny  $X[n, k]$  dla DFT o długości 8. Minimalne przesunięcie okna wzdłuż sygnału wynosi  $n=1$  (rys.9.13 lewy), jednak okno może być przesuwane o większą liczbę próbek (np. o dwie jak na rys.9.13 prawy). Dla każdego przesunięcia okna obliczane jest DFT sygnału, tak więc płaszczyzna  $X[n, k]$  jest zbudowana z widm sygnałów wyliczonych dla kolejnych przesunięć okna.

Innym punktem spojrzenia na płaszczyznę  $X[n, k]$  jest stwierdzenie, że dla danego  $k$  otrzymujemy ciąg będący efektem filtracji filtrem pasmowym, co zilustrowano na rys.9.14. Częstotliwość próbkowania każdego z podpasm może być zredukowana  $2\pi/\Delta_{ml}$  razy, gdzie  $\Delta_{ml}$  jest szerokością listka głównego transformaty Fouriera okna.



Rys.9.13 Przykłady próbkowania płaszczyzny  $X[n, k]$ , dla  $k=0,1,\dots,7$ .



Rys.9.14 Interpretacja płaszczyzny  $X[n, k]$ , jako wyniku filtracji zespołem filtrów pasmowych.

Powszechnym zastosowaniem spektrogramu jest analiza dyskretnych sygnałów niestacjonarnych, czyli takich, których parametry (tj. amplituda, częstotliwość lub faza) zmieniają się w czasie (np. sygnał mowy, sygnały radarowe i inne).

Ograniczenia rozdzielczości częstotliwościowej i amplitudowej DFT dotyczą również spektrogramu i mogą być redukowane przez dobór właściwego okna analizy.

Jako ilustrację reprezentacji czasowo-częstotliwościowej sygnału rozważmy sygnał sinusoidalny z sinusoidalną modulacją amplitudy, sygnał sinusoidalny z sinusoidalną modulacją częstotliwości oraz sygnał mowy.

Sygnał sinusoidalny zmodulowany amplitudowo sygnałem  $m[n]=\cos(\omega_m n)$ ,  $\omega_m \ll \omega_0$  ma postać:

$$x[n] = (1 + k m[n]) \cos(\omega_0 n), \quad 0 < k \leq 1, \quad (9.66)$$

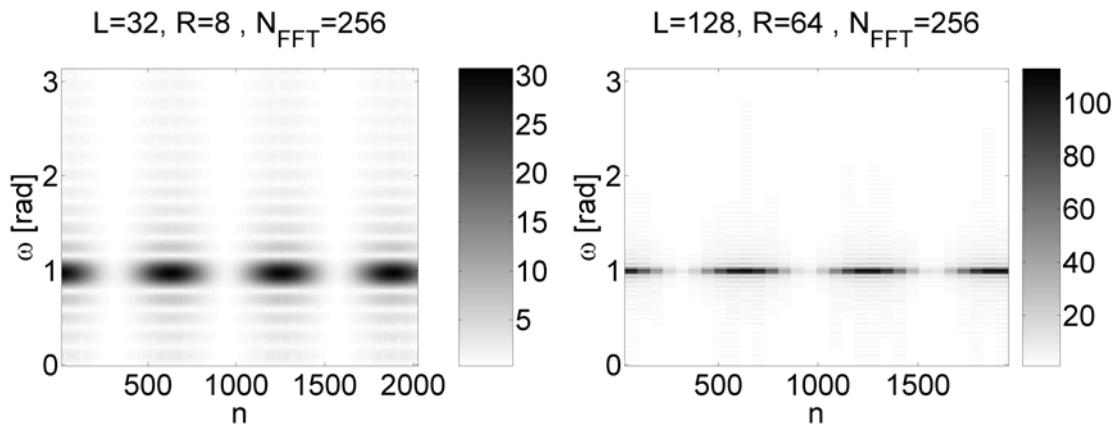
a sygnał o częstotliwość chwilowej  $m[n] = \omega_0 - k \sin(\omega_m n)$  jest dany wzorem:

$$x[n] = \cos(\omega_0 n + k \cos(\omega_m n) / \omega_m). \quad (9.67)$$

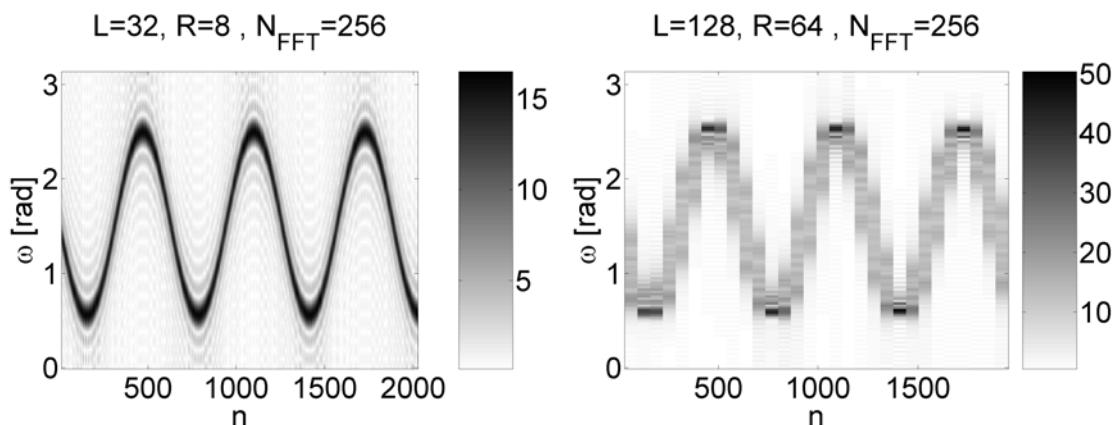
Rys.9.15 przedstawia spektrogramy, tj.  $|X[n, k]|$ , sygnału z modulacją amplitudy dla różnych parametrów  $L$ ,  $R$  i  $N_{FFT}$ , gdzie  $L$  jest długością okna czasowego,  $R$  przesunięciem okna czasowego, a  $N_{FFT}$  długością FFT. W przypadku, gdy  $N_{FFT} > L$ , wybrany przez okno fragment sygnału jest uzupełniany zerami do długości  $N_{FFT}$ . Oś  $n$  na rys.9.15 jest osią czasu, a osią  $\omega$  osią częstotliwości. Płaszczyzny  $|X[n, k]|$  ilustrują więc zmianę częstotliwości sygnału w czasie. W przypadku modulacji amplitudy częstotliwość nośnej jest stała, na rys.9.15 wynosi ona 1 rad, natomiast amplituda nośnej zmienia się w czasie. Na rys.9.14 widoczne są listki

boczne okna prostokątnego. Zwiększenie długości okna analizy  $L$  powoduje zwiększenie rozdzielczości częstotliwościowej i obniżenie rozdzielczości czasowej.

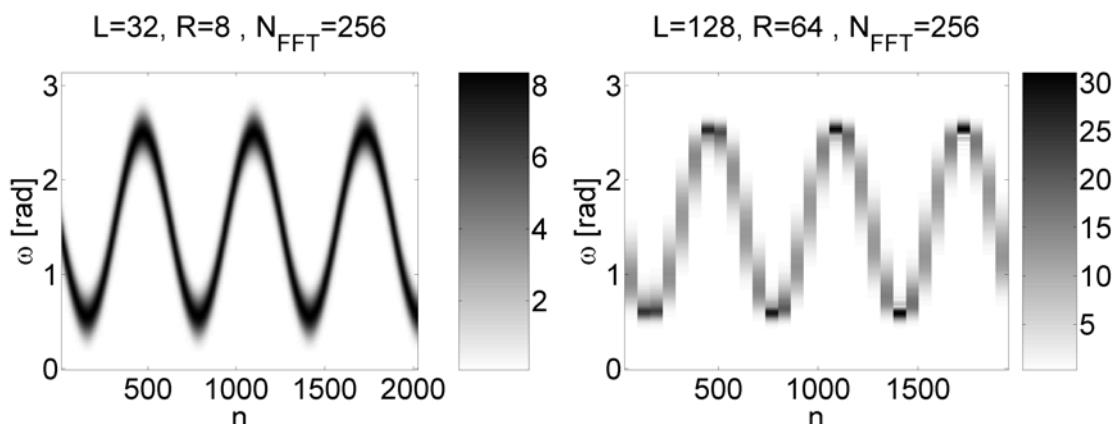
Rys.9.16 i rys.9.17 przedstawiają spektrogramy sygnału z sinusoidalną modulacją częstotliwości wyznaczone z oknem prostokątnym i oknem Hamminga. Zastosowanie okna Hamminga powoduje tłumienie listków bocznych, niestety szerszy listek główny tego okna powoduje rozmycie spektrogramu.



Rys.9.15 Modulacja AM, okno prostokątne.



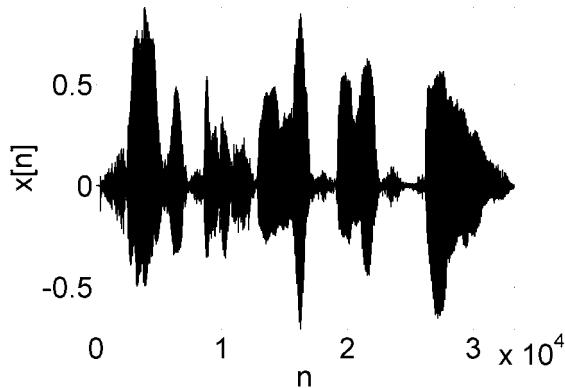
Rys.9.16 Modulacja FM, okno prostokątne.



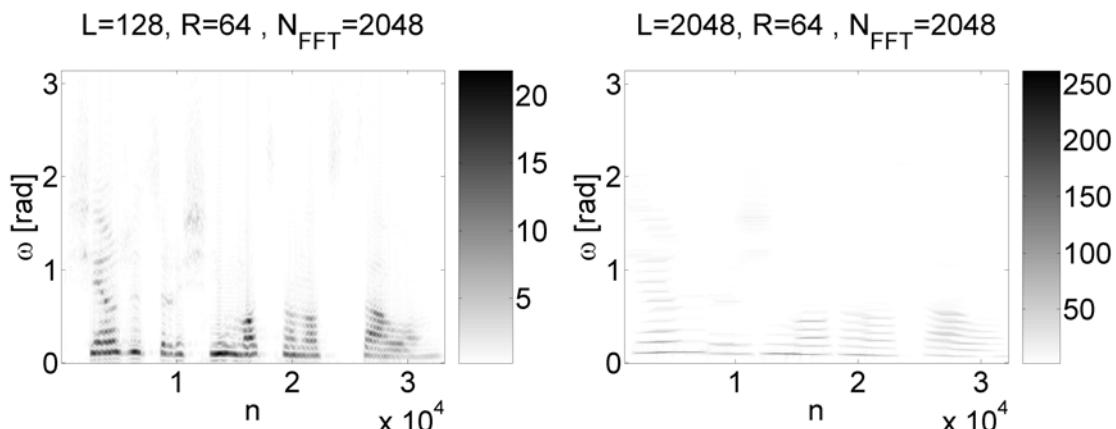
Rys.9.17 Modulacja FM, okno Hamminga.

Rys.9.18 przedstawia przebieg czasowy sygnału mowy. Jest to zdanie 'książyc lśnił mocno noc cała' wypowiadane przez kobietę. Rys.9.19 przedstawia spektrogramy tego sygnału mowy dla krótkiego  $L=128$  i długiego  $L=2048$  prostokątnego okna analiz. Dla

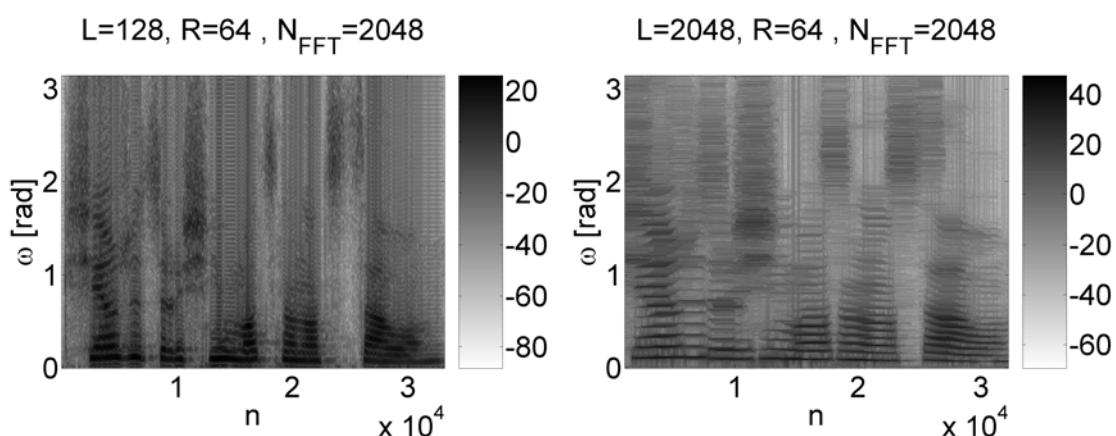
sygnałów o dużej dynamice w spektrogramie stosuje się skalę decybelową,  $20\log_{10}|X[n, k]|$ , co pokazano na rys.9.20. W spektrogramach sygnału mowy widoczne są głoski dźwięczne zawierające częstotliwości harmoniczne. Reprezentacje czasowo - częstotliwościowe sygnału mowy są podstawą algorytmów rozpoznawania treści, mówcy i patologii narządu mowy.



Rys.9.18 Sygnał mowy, lektor kobieta, fraza 'księżyca lśnił mocno noc całą'.



Rys.9.19 Sygnał mowy, okno prostokątne.



Rys.9.20 Sygnał mowy, skala decybelowa, okno prostokątne.

#### **9.4 Analiza stacjonarnych sygnałów losowych - periodogram**

W przypadku, gdy sygnał może być modelowany jako suma sinusoid, DFT jest naturalnym narzędziem analizy, natomiast dla sygnałów mających charakter szumowy stosuje się modele

sygnałów losowych z parametrami takimi jak: wartość średnia (DC), wariancja (moc średnia), funkcja autokorelacji, gęstość widmowa mocy. Typowa estymata wartości średniej dla skończonej realizacji stacjonarnego procesu losowego o długości  $L$  jest zdefiniowana jako średnia próbek:

$$\hat{m}_x = \frac{1}{L} \sum_{n=0}^{L-1} x[n]. \quad (9.68)$$

Podobnie wariancja jest zdefiniowana jako wariancja próbek:

$$\hat{\sigma}_x^2 = \frac{1}{L} \sum_{n=0}^{L-1} (x[n] - \hat{m}_x)^2. \quad (9.69)$$

Oba powyższe estymatory są asymptotycznie nieobciążone (*unbiased*), tzn. wartość oczekiwana  $\hat{m}_x$  osiąga prawdziwą średnią  $m_x$ , a wartość oczekiwana wariancji  $\hat{\sigma}_x^2$  osiągną prawdziwą wariancję  $\sigma_x^2$ , gdy  $L$  osiąga  $\infty$ . Dodatkowo oba estymatory są zgodne, tzn. poprawiają estymację wraz ze wzrostem  $L$ , tzn. ich wariancja maleje do 0, gdy  $L$  osiąga  $\infty$ .

Rozważmy estymatę widmowej gęstości mocy w postaci:

$$I(\omega) = \frac{1}{LU} |V(e^{j\omega})|^2, \quad (9.70)$$

gdzie:

$$V(e^{j\omega}) = \sum_{n=0}^{L-1} w[n]x[n]e^{-j\omega n}, \quad (9.71)$$

a  $U$  jest stałą normalizującą:

$$U = \frac{1}{L} \sum_{n=0}^{L-1} (w[n])^2. \quad (9.72)$$

Dla okna prostokątnego taki estymator nazywa się periodogramem, a dla okna innego niż prostokątne  $I(\omega)$  nazywa się zmodyfikowanym periodogramem. Wzór definicyjny (9.70) można przedstawić w postaci:

$$I(\omega) = \frac{1}{LU} \sum_{m=-(L-1)}^{L-1} c_{vv}[m] e^{-j\omega m}, \quad (9.73)$$

gdzie  $c_{vv}[m]$  jest autokorelacją ciągu  $v[n] = x[n]w[n]$

$$c_{vv}[m] = \sum_{n=0}^{L-1} x[n]w[n]x[n+m]w[n+m]. \quad (9.74)$$

Periodogram jest transformatą Fouriera autokorelacji okienkowanego ciągu. Wartości  $I(\omega)$  są rzeczywiste i parzyste względem połowy częstotliwości próbkowania.

Periodogram wyznacza się dla skończonego zbioru częstotliwości dyskretnych za pomocą DFT.

Przed liczeniem periodogramu od sygnału można odjąć wartość średnią, co zmniejsza przeciek widmowy od składowej stałej widma.

Periodogram ze stałą normalizującą  $U$  jest asymptotycznie nieobciążony, tzn. obciążenie osiąga zero wraz ze wzrostem długości okna  $L$ .

Periodogram nie jest estymatorem zgodnym, ponieważ jego wariancja nie dąży do zera wraz ze wzrostem długości okna  $L$ .

W celu uzyskania asymptotycznie nieobciążonego i zgodnego estymator widmowej gęstości mocy stosuje się metodę Welch'a uśredniania zmodyfikowanych periodogramów. Ciąg danych  $x[n]$  jest dzielony na fragmenty o długości  $L$  próbek, które następnie są okienkowane:

$$x_r[n] = x[rR + n]w[n], \quad 0 \leq n \leq L - 1. \quad (9.75)$$

Jeżeli  $R < L$  to segmenty zachodzą na siebie, w metodzie Welch'a stosuje się zachodzenie poszczególnych segmentów o połowę długości okna, gdyż powoduje to najlepsze zmniejszenie wariancji estymatora. Periodogram  $r$ -tego segmentu jest obliczany jako:

$$I_r(\omega) = \frac{1}{LU} |X_r(e^{j\omega})|^2. \quad (9.76)$$

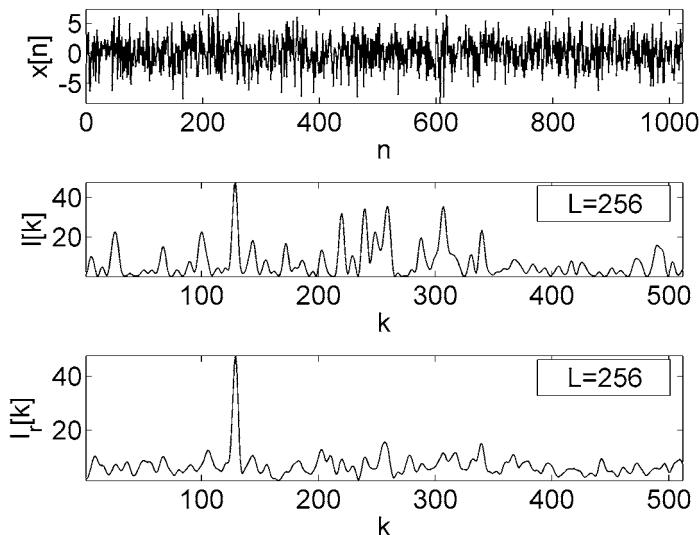
Następnie  $K$  periodogramów poszczególnych segmentów jest uśredniane:

$$\hat{I}(\omega) = \frac{1}{K} \sum_{r=0}^{K-1} I_r(\omega). \quad (9.77)$$

Uśrednianie periodogramów daje asymptotycznie nieobciążony i zgodny estymator widmowej gęstości mocy. W praktyce do liczenia uśrednionych periodogramów stosuje się DFT, wówczas:

$$I_r[k] = \frac{1}{LU} |X_r[k]|^2, \quad \hat{I}[k] = \frac{1}{K} \sum_{r=0}^{K-1} I_r[k]. \quad (9.78)$$

Głównym zastosowaniem estymatorów widmowej gęstości mocy jest poszukiwanie oscylacji w sygnałach zakłóconych. Rys.9.21 przedstawia sygnał sinusoidalny zakłócony addytywnie szumem o rozkładzie normalnym i odchyleniu standardowym równym 2.5 amplitudy sygnału. Zmodyfikowany periodogram sygnału  $I[k]$  wyznaczony z oknem Hamminga zawiera wiele błędnie estymowanych częstotliwości, natomiast periodogram uśredniony, z tym samym oknem, poprawnie estymuje jedną częstotliwość w sygnale.



Rys.9.21 Zakłócony sygnał sinusoidalny i jego zmodyfikowany periodogram i uśredniony periodogram.



## 10. Aproksymacja sygnałów, filtry optymalne i filtry adaptacyjne

### 10.1 Aproksymacja sygnałów w przestrzeni wielowymiarowej

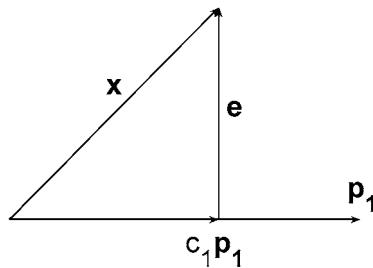
Rozważmy następujący problem: dany jest dwuwymiarowy (dwuelementowy) wektor  $\mathbf{x}$ , zadanie polega na aproksymacji tego wektora za pomocą liniowej kombinacji dwuwymiarowego wektora  $\mathbf{p}_1$ , w taki sposób, aby błąd aproksymacji był jak najmniejszy. Sygnał aproksymowany jest, więc w postaci:

$$\hat{\mathbf{x}} = c_1 \mathbf{p}_1, \quad (10.1)$$

a błąd aproksymacji dany jest wzorem:

$$\|\mathbf{e}\| = \|\mathbf{x} - \hat{\mathbf{x}}\| = \|\mathbf{x} - c_1 \mathbf{p}_1\|, \quad (10.2)$$

gdzie  $\|\cdot\|$  oznacza przyjętą normę wektora. Dla  $N$  wymiarowej normy euklidesowej, tj.  $\|\mathbf{x}\| = \sqrt{x^2[0] + x^2[1] + \dots + x^2[N-1]}$ , powyższy problem ma interpretację geometryczną przedstawioną na rys.10.1.



Rys.10.1 Interpretacja geometryczna problemu aproksymacji na płaszczyźnie.

Jak widać z rys.10.1, w rozważanym przypadku, błąd aproksymacji  $\mathbf{e}$  jest najmniejszy wtedy, gdy jest on prostopadły do wektora  $\mathbf{p}_1$ :

$$\mathbf{e} \perp \mathbf{p}_1, \quad (10.3)$$

co oznacza, że iloczyn skalarny tych wektorów jest równy zeru

$$\langle \mathbf{e}, \mathbf{p}_1 \rangle = \langle \mathbf{x} - c_1 \mathbf{p}_1, \mathbf{p}_1 \rangle = 0. \quad (10.4)$$

Współczynnik aproksymacji można wyliczyć z zależności ( $\langle \mathbf{x}, \mathbf{p}_1 \rangle = \sum_{n=0}^{N-1} x[n] p_1[n]$ ):

$$c_1 = \frac{\langle \mathbf{x}, \mathbf{p}_1 \rangle}{\|\mathbf{p}_1\|^2} = \frac{x[0]p_1[0] + x[1]p_1[1]}{p_1^2[0] + p_1^2[1]}. \quad (10.5)$$

Geometrycznie  $c_1$  jest projekcją (rzutem) wektora  $\mathbf{x}$  w kierunku  $\mathbf{p}_1$ .

Powyższy problem można również rozwiązać licząc pochodną błędu aproksymacji (10.2) względem  $c_1$  i przyrównując ją do zera.

Analogiczną dyskusję można przeprowadzić dla przestrzeni trzywymiarowej, dla której interpretację geometryczną przedstawia rys.10.2. Zadanie polega na aproksymacji wektora  $\mathbf{x}$ , jako liniowej kombinacji wektorów  $\mathbf{p}_1$  i  $\mathbf{p}_2$ , tzn.:

$$\mathbf{x} = c_1 \mathbf{p}_1 + c_2 \mathbf{p}_2 + \mathbf{e}. \quad (10.6)$$

Z rys.10.2 widać, że błąd aproksymacji jest najmniejszy, kiedy wektor  $\mathbf{e}$  jest prostopadły do wektorów  $\mathbf{p}_1$  i  $\mathbf{p}_2$ , co pozwala sformułować następujące warunki ortogonalności:

$$\begin{aligned} \langle \mathbf{x} - (c_1 \mathbf{p}_1 + c_2 \mathbf{p}_2), \mathbf{p}_1 \rangle &= 0 \\ \langle \mathbf{x} - (c_1 \mathbf{p}_1 + c_2 \mathbf{p}_2), \mathbf{p}_2 \rangle &= 0 \end{aligned} \quad (10.7)$$

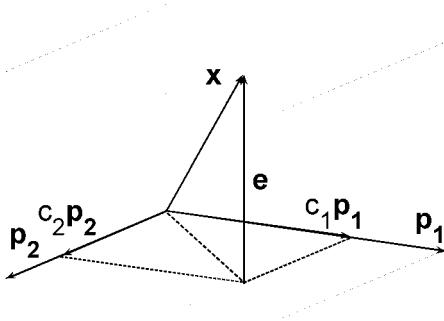
Korzystając w własności iloczynu skalarnego otrzymujemy:

$$\begin{aligned} \langle \mathbf{x}, \mathbf{p}_1 \rangle &= c_1 \langle \mathbf{p}_1, \mathbf{p}_1 \rangle + c_2 \langle \mathbf{p}_2, \mathbf{p}_1 \rangle \\ \langle \mathbf{x}, \mathbf{p}_2 \rangle &= c_1 \langle \mathbf{p}_1, \mathbf{p}_2 \rangle + c_2 \langle \mathbf{p}_2, \mathbf{p}_2 \rangle, \end{aligned} \quad (10.8)$$

a w postaci macierzowej:

$$\begin{bmatrix} \langle \mathbf{p}_1, \mathbf{p}_1 \rangle & \langle \mathbf{p}_2, \mathbf{p}_1 \rangle \\ \langle \mathbf{p}_1, \mathbf{p}_2 \rangle & \langle \mathbf{p}_2, \mathbf{p}_2 \rangle \end{bmatrix} \begin{bmatrix} c_1 \\ c_2 \end{bmatrix} = \begin{bmatrix} \langle \mathbf{x}, \mathbf{p}_1 \rangle \\ \langle \mathbf{x}, \mathbf{p}_2 \rangle \end{bmatrix}. \quad (10.9)$$

Szukane współczynniki  $c_1$  i  $c_2$  można wyliczyć odwracając macierz w (10.9).



Rys.10.2 Interpretacja geometryczna problemu aproksymacji w przestrzeni trzywymiarowej.

Ogólnie w przestrzeni wielowymiarowej wektor  $\mathbf{x}$  może być aproksymowany jako kombinacja liniowa wektorów  $\mathbf{p}_m$  minimalizująca błąd  $\|\mathbf{e}\| = \|\mathbf{x} - \hat{\mathbf{x}}\|$ :

$$\mathbf{x} = \sum_{m=1}^M c_m \mathbf{p}_m + \mathbf{e} = \hat{\mathbf{x}} + \mathbf{e}. \quad (10.10)$$

Dla przestrzeni euklidesowej jest to liniowy problem najmniejszych kwadratów (*linear least-squares problem*). Powyższy problem aproksymacji, tj. minimalizacji błędu aproksymacji, można wyrazić poprzez warunek ortogonalności błędu do każdego z wektorów  $\mathbf{p}_m$ , jest to tzw. zasada ortogonalności (*orthogonality principle*):

$$\left\langle \mathbf{x} - \sum_{m=1}^M c_m \mathbf{p}_m, \mathbf{p}_k \right\rangle = 0, \quad k = 1, 2, \dots, M, \quad (10.11)$$

co prowadzi do układu  $M$  równań z  $M$  niewiadomymi w postaci:

$$\begin{bmatrix} \langle \mathbf{p}_1, \mathbf{p}_1 \rangle & \langle \mathbf{p}_2, \mathbf{p}_1 \rangle & \cdots & \langle \mathbf{p}_M, \mathbf{p}_1 \rangle \\ \langle \mathbf{p}_1, \mathbf{p}_2 \rangle & \langle \mathbf{p}_2, \mathbf{p}_2 \rangle & \cdots & \langle \mathbf{p}_M, \mathbf{p}_2 \rangle \\ \vdots & \vdots & \ddots & \vdots \\ \langle \mathbf{p}_1, \mathbf{p}_M \rangle & \langle \mathbf{p}_2, \mathbf{p}_M \rangle & \cdots & \langle \mathbf{p}_M, \mathbf{p}_M \rangle \end{bmatrix} \begin{bmatrix} c_1 \\ c_2 \\ \vdots \\ c_M \end{bmatrix} = \begin{bmatrix} \langle \mathbf{x}, \mathbf{p}_1 \rangle \\ \langle \mathbf{x}, \mathbf{p}_2 \rangle \\ \vdots \\ \langle \mathbf{x}, \mathbf{p}_M \rangle \end{bmatrix}. \quad (10.12)$$

Definiując macierz autokorelacji  $\mathbf{R}$ , wektor współczynników  $\mathbf{c}$  oraz wektor korelacji wzajemnej  $\mathbf{p}$  w następujący sposób:

$$\mathbf{R} = \begin{bmatrix} \langle \mathbf{p}_1, \mathbf{p}_1 \rangle & \langle \mathbf{p}_2, \mathbf{p}_1 \rangle & \cdots & \langle \mathbf{p}_M, \mathbf{p}_1 \rangle \\ \langle \mathbf{p}_1, \mathbf{p}_2 \rangle & \langle \mathbf{p}_2, \mathbf{p}_2 \rangle & \cdots & \langle \mathbf{p}_M, \mathbf{p}_2 \rangle \\ \vdots & \vdots & \ddots & \vdots \\ \langle \mathbf{p}_1, \mathbf{p}_M \rangle & \langle \mathbf{p}_2, \mathbf{p}_M \rangle & \cdots & \langle \mathbf{p}_M, \mathbf{p}_M \rangle \end{bmatrix}, \quad \mathbf{c} = \begin{bmatrix} c_1 \\ c_2 \\ \vdots \\ c_M \end{bmatrix}, \quad \mathbf{p} = \begin{bmatrix} \langle \mathbf{x}, \mathbf{p}_1 \rangle \\ \langle \mathbf{x}, \mathbf{p}_2 \rangle \\ \vdots \\ \langle \mathbf{x}, \mathbf{p}_M \rangle \end{bmatrix}, \quad (10.13)$$

możemy zapisać (10.12) w postaci:

$$\mathbf{R}\mathbf{c} = \mathbf{p}. \quad (10.14)$$

Równania w postaci (10.14) są określane jako równania normalne. Macierz  $\mathbf{R}$  jest odwrotnalna, jeżeli wektory  $\mathbf{p}_m$  są liniowo niezależne.

Taki sam wynik, tj. (10.14), można uzyskać licząc gradienty z kwadratu błędu po wektorze  $\mathbf{c}$  i przyrównując je do zera. Funkcja kosztu jest następująca (dla uproszczenia notacji przyjęto, że wektor  $\mathbf{c}$  ma wartości ze zbioru liczb rzeczywistych):

$$J(\mathbf{c}) = \|\mathbf{e}\|^2 = \|\mathbf{x} - \hat{\mathbf{x}}\|^2 = \left\langle \mathbf{x} - \sum_{m=1}^M c_m \mathbf{p}_m, \mathbf{x} - \sum_{k=1}^M c_k \mathbf{p}_k \right\rangle = \langle \mathbf{x}, \mathbf{x} \rangle - 2 \sum_{m=1}^M c_m \langle \mathbf{x}, \mathbf{p}_m \rangle + \sum_{m=1}^M \sum_{k=1}^M c_m c_k \langle \mathbf{p}_m, \mathbf{p}_k \rangle. \quad (10.15)$$

Stosując oznaczenia (10.13) możemy zapisać (10.15) w postaci:

$$J(\mathbf{c}) = \|\mathbf{x}\|^2 - 2\mathbf{c}^T \mathbf{p} + \mathbf{c}^T \mathbf{R} \mathbf{c}. \quad (10.16)$$

Gradient funkcji kosztu (10.16) wynosi:

$$\frac{\partial}{\partial \mathbf{c}} (\|\mathbf{x}\|^2 - 2\mathbf{c}^T \mathbf{p} + \mathbf{c}^T \mathbf{R} \mathbf{c}) = -\mathbf{p} + \mathbf{R} \mathbf{c}, \quad (10.17)$$

co po przyrównaniu do zera daje (10.14).

Problem najmniejszych kwadratów można sformułować wykorzystując notację macierzową. Aproksymacja sygnału jest w postaci:

$$\hat{\mathbf{x}} = \sum_{m=1}^M c_m \mathbf{p}_m = [\mathbf{p}_1 \ \mathbf{p}_2 \ \cdots \ \mathbf{p}_M] \begin{bmatrix} c_1 \\ c_2 \\ \vdots \\ c_M \end{bmatrix}, \quad (10.18)$$

gdzie  $\mathbf{p}_m$  są wektorami kolumnowymi. Definiując macierz  $\mathbf{A} = [\mathbf{p}_1 \ \mathbf{p}_2 \ \dots \ \mathbf{p}_M]$  możemy zapisać (10.18) w postaci  $\hat{\mathbf{x}} = \mathbf{Ac}$ . Problem aproksymacji polega na tym, żeby wyznaczyć współczynniki  $\mathbf{c}$ , które minimalizują błąd  $\|\mathbf{e}\|^2$  w wyrażeniu

$$\boxed{\mathbf{x} = \mathbf{Ac} + \mathbf{e} = \hat{\mathbf{x}} + \mathbf{e}}. \quad (10.19)$$

Minimum błędu występuje, kiedy błąd  $\mathbf{e}$  jest ortogonalny do każdego z wektorów  $\mathbf{p}_m$ :

$$\langle \mathbf{x} - \mathbf{Ac}, \mathbf{p}_m \rangle = 0, \quad m = 1, 2, \dots, M, \quad (10.20)$$

co możemy przepisać w postaci:

$$\begin{bmatrix} \mathbf{p}_1^H \\ \mathbf{p}_2^H \\ \vdots \\ \mathbf{p}_M^H \end{bmatrix} (\mathbf{x} - \mathbf{Ac}) = \mathbf{0}. \quad (10.21)$$

Ponieważ wektory  $\mathbf{p}_m^H$  tworzą macierz  $\mathbf{A}^H$ , otrzymujemy:

$$\mathbf{A}^H \mathbf{Ac} = \mathbf{A}^H \mathbf{x}. \quad (10.22)$$

Optymalne współczynniki dane są więc zależnością:

$$\boxed{\mathbf{c} = (\mathbf{A}^H \mathbf{A})^{-1} \mathbf{A}^H \mathbf{x} = \mathbf{R}^{-1} \mathbf{p}}. \quad (10.23)$$

Macierz  $(\mathbf{A}^H \mathbf{A})^{-1} \mathbf{A}^H$  jest macierzą pseudoodwrotną macierzy  $\mathbf{A}$ . Zależność (10.23) nie jest stosowana w praktyce ze względu na złe własności numeryczne.

Aproksymacja sygnału dana jest zależnością:

$$\boxed{\hat{\mathbf{x}} = \mathbf{Ac} = \mathbf{A}(\mathbf{A}^H \mathbf{A})^{-1} \mathbf{A}^H \mathbf{x} = \mathbf{Px}}, \quad (10.24)$$

gdzie  $\mathbf{P}$  jest tzw. macierzą rzutowania (*projection matrix*).

Jako przykład zastosowania metody najmniejszych kwadratów rozważmy regresję liniową. Zadanie polega na tym, aby określić parametry  $a$  i  $b$  prostej  $y[n] = ax[n] + b$  najlepiej dopasowanej do zbioru danych  $\{(x[0], y[0]), (x[1], y[1]), \dots, (x[M-1], y[M-1])\}$ . Wykorzystując dane pomiarowe możemy zapisać układ równań:

$$\begin{bmatrix} y[0] \\ y[1] \\ \vdots \\ y[M-1] \end{bmatrix} = \begin{bmatrix} ax[0] + b \\ ax[1] + b \\ \vdots \\ ax[M-1] + b \end{bmatrix} + \begin{bmatrix} e[0] \\ e[1] \\ \vdots \\ e[M-1] \end{bmatrix}. \quad (10.25)$$

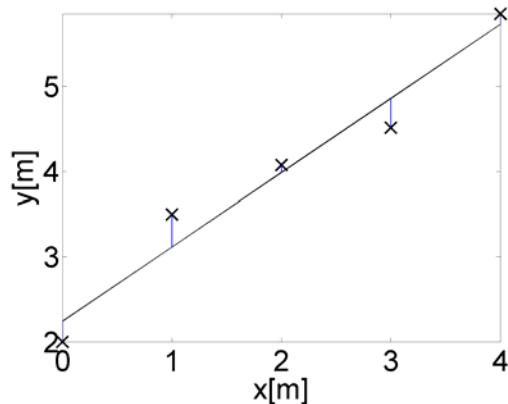
Stosując podstawienia:

$$\mathbf{y} = [y[0] \ y[1] \ \dots \ y[M-1]]^T, \quad \mathbf{e} = [e[0] \ e[1] \ \dots \ e[M-1]]^T, \quad \mathbf{c} = \begin{bmatrix} a \\ b \end{bmatrix} \text{ i } \mathbf{A} = \begin{bmatrix} x[0] & 1 \\ x[1] & 1 \\ \vdots & \vdots \\ x[M-1] & 1 \end{bmatrix}, \quad (10.26)$$

możemy przepisać (10.25) w postaci:

$$\mathbf{y} = \mathbf{Ac} + \mathbf{e}, \quad (10.27)$$

analogicznej do (10.19), dla której rozwiązańem dane jest zależnością (10.23). Dla znalezionej prostej suma kwadratów odległości wertykalnych punktów pomiarowych od tej prostej jest najmniejsza. Przykład regresji liniowej dla zbioru pięciu punktów pomiarowych przedstawiony jest na rys.10.3. Rys.10.3 przedstawia również błędy aproksymacji jako odległości wertykalne punktów pomiarowych od prostej.



Rys.10.3 Regresja liniowa.

## 10.2 Filtracja optymalna w sensie najmniejszych kwadratów

Zadanie filtracji optymalnej polega na wyznaczeniu współczynników odpowiedzi impulsowej filtra  $h[n]$  o długości  $M$ , dla których sygnał wyjściowy filtra  $y[n]$  jest dopasowany do sygnału pożądanego (odniesienia)  $d[n]$ . Sygnał na wyjściu filtra opisany jest przez splot liniowy:

$$y[n] = \sum_{k=0}^{M-1} h[k]x[n-k]. \quad (10.28)$$

Podstawiając  $d[n]=y[n]+e[n]$  do (10.28) otrzymujemy:

$$d[n] = \sum_{k=0}^{M-1} h[k]x[n-k] + e[n]. \quad (10.29)$$

Rozwiązańem najlepsze w sensie najmniejszych kwadratów minimalizuje sumę kwadratów błędów  $e[n]$  w przedziale od  $n_1$  do  $n_2$ :

$$\min \sum_{n=n_1}^{n_2} |e[n]|^2. \quad (10.30)$$

Uwzględniając zakres sygnału wejściowego od  $n_1$  do  $n_2$ , (10.29) można przedstawić w postaci macierzowej jako:

$$\begin{bmatrix} d[n_1] \\ d[n_1+1] \\ \vdots \\ d[n_2] \end{bmatrix} = \mathbf{A} \begin{bmatrix} h[0] \\ h[1] \\ \vdots \\ h[M-1] \end{bmatrix} + \begin{bmatrix} e[n_1] \\ e[n_1+1] \\ \vdots \\ e[n_2] \end{bmatrix}, \quad (10.31)$$

czyli

$$\mathbf{d} = \mathbf{Ah} + \mathbf{e}, \quad (10.32)$$

gdzie  $\mathbf{A}$  jest macierzą zbudowaną z próbek sygnału wejściowego  $x[n]$ . Rozwiążanie (10.32) jest w postaci (10.23).

W praktyce stosuje się kilka metod budowania macierzy  $\mathbf{A}$  w (10.32) dla sygnału  $x[n]$  o długości  $N$ :

1. Metoda kowariancji - użyte są tylko dostępne dane; brak założeń o sygnale poza przedziałem obserwacji; macierz danych  $\mathbf{A}$  ma rozmiar  $(N-M+1) \times M$ :

$$\mathbf{A} = \begin{bmatrix} x[M] & x[M-1] & x[M-2] & \cdots & x[1] \\ x[M+1] & x[M] & x[M-1] & \cdots & x[2] \\ \vdots & \vdots & \vdots & & \vdots \\ x[N] & x[N-1] & x[N-2] & \cdots & x[N-M+1] \end{bmatrix}. \quad (10.33)$$

2. Metoda autokorelacji - zakłada się, że wartość sygnału poza przedziałem obserwacji wynosi zero; macierz danych  $\mathbf{A}$  ma rozmiar  $(N+M+1) \times M$ :

$$\mathbf{A} = \begin{bmatrix} x[1] & 0 & 0 & \cdots & 0 \\ x[2] & x[1] & 0 & \cdots & 0 \\ x[3] & x[2] & x[1] & \cdots & 0 \\ \vdots & & & & \vdots \\ x[M] & x[M-1] & x[M-2] & \cdots & x[1] \\ x[M+1] & x[M] & x[M-1] & \cdots & x[2] \\ \vdots & & & & \vdots \\ x[N] & x[N-1] & x[N-2] & \cdots & x[N-M+1] \\ 0 & x[N] & x[N-1] & \cdots & x[N-M+2] \\ \vdots & & & & \vdots \\ 0 & 0 & 0 & \cdots & x[N] \end{bmatrix}. \quad (10.34)$$

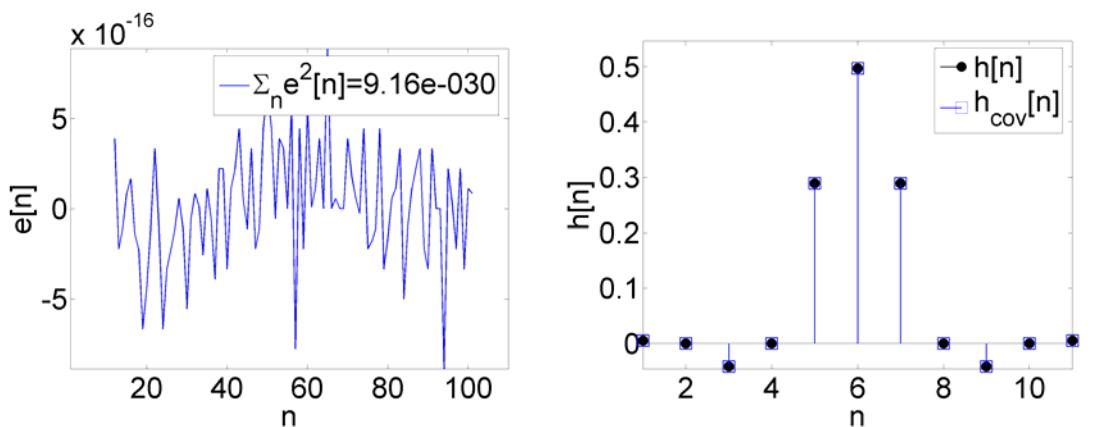
3. Metoda *pre-windowing* - zakłada się, że  $x[n]=0$  dla  $n<1$ ; macierz danych  $\mathbf{A}$  ma rozmiar  $N \times M$ :

$$\mathbf{A} = \begin{bmatrix} x[1] & 0 & 0 & \cdots & 0 \\ x[2] & x[1] & 0 & \cdots & 0 \\ x[3] & x[2] & x[1] & \cdots & 0 \\ \vdots & \vdots & \vdots & & \vdots \\ x[M] & x[M-1] & x[M-2] & \cdots & x[1] \\ x[M+1] & x[M] & x[M-1] & \cdots & x[2] \\ \vdots & \vdots & \vdots & & \vdots \\ x[N] & x[N-1] & x[N-2] & \cdots & x[N-M+1] \end{bmatrix}. \quad (10.35)$$

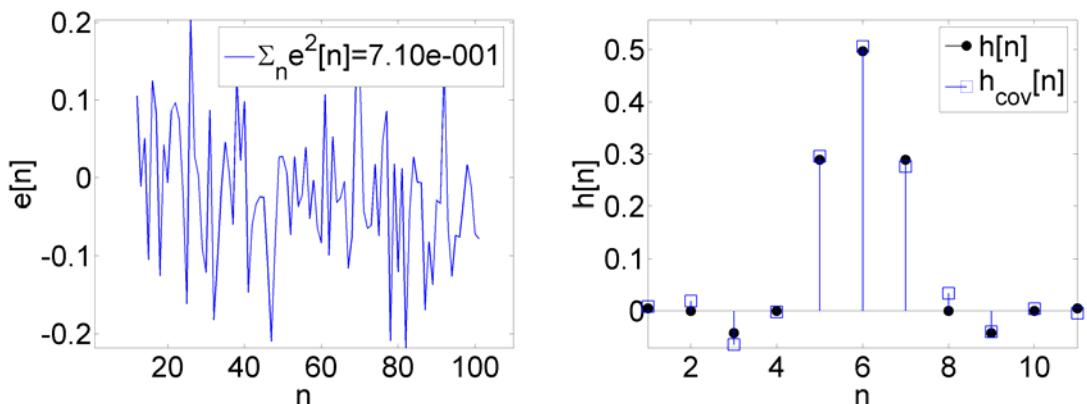
4. Metoda *post-windowing* - zakłada się, że  $x[n]=0$  dla  $n>N$ ; macierz danych  $\mathbf{A}$  ma rozmiar  $N \times M$ :

$$\mathbf{A} = \begin{bmatrix} x[M] & x[M-1] & x[M-2] & \cdots & x[1] \\ x[M+1] & x[M] & x[M-1] & \cdots & x[2] \\ \vdots & \vdots & \vdots & & \vdots \\ x[N] & x[N-1] & x[N-2] & \cdots & x[N-M+1] \\ 0 & x[N] & x[N-1] & \cdots & x[N-M+2] \\ \vdots & \vdots & \vdots & & \vdots \\ 0 & 0 & 0 & \cdots & x[N] \end{bmatrix}. \quad (10.36)$$

Jako przykład zastosowania filtracji optymalnej rozważmy identyfikację odpowiedzi impulsowej  $h[n]$  układu LTI na podstawie znajomości odpowiedzi  $y[n]$  tego układu na znane wymuszenie  $x[n]$ . Szukane współczynniki odpowiedzi impulsowej wyznaczymy rozwiązując  $\mathbf{d} = \mathbf{Ah} + \mathbf{e}$  (10.32), przy czym sygnałem odniesienia jest sygnał wyjściowy identyfikowanego układu, tj.  $d[n]=y[n]$ .



Rys.10.4 Identyfikacja odpowiedzi impulsowej - pomiar bez zakłóceń.



Rys.10.5 Identyfikacja odpowiedzi impulsowej - pomiar zakłócony szumem addytywnym.

Przykładowe wyniki identyfikacji odpowiedzi impulsowej  $h[n]$  przedstawione są na rys.10.4 i rys.10.5. Sygnałem wejściowym (wymuszeniem) był szum biały o rozkładzie normalnym. Tego typu sygnał testowy jest często wykorzystywany, ponieważ jego widmo zawiera wszystkie częstotliwości. Sygnał odniesienia obliczono jako splot z zadaną odpowiedzią impulsową  $h[n]$ , zadanie polegało na identyfikacji tej odpowiedzi. Macierz  $\mathbf{A}$  zbudowano metodą kowariancji (10.33). Rys.10.4 przedstawia błąd aproksymacji oraz zadaną  $h[n]$  i wyznaczoną metodą kowariancji  $h_{cov}[n]$  odpowiedź impulsową filtra. W przypadku, kiedy odpowiedź układu jest dostępna bez zakłóceń, dokładność identyfikacji jest

ograniczona dokładnością obliczeń numerycznych. Rys. 10.5 przedstawia wyniki identyfikacji dla przypadku, gdy odpowiedź układu jest zakłócona szumem addytywnym.

### 10.3 Filtry adaptacyjne RLS

Problem filtracji optymalnej w sensie najmniejszych kwadratów sprowadza się do odwrócenia macierzy  $\mathbf{R} = \mathbf{A}^H \mathbf{A}$  w (10.23). Rozwiążanie jest uzyskiwane dla całego wektora dostępnych danych, tzn. blokowo. Innym sposobem uzyskania rozwiązania jest adaptacyjny algorytm rekurencyjny RLS (*recursive least-squares*), który wyznacza współczynniki filtra  $h[n]$  w miarę jak napływają kolejne próbki sygnału, czyli na bieżąco. Dodatkowo algorytm RLS nie wymaga odwracania macierzy  $\mathbf{R}$ .

Macierz  $\mathbf{R}$  możemy przedstawić w następującej postaci:

$$\mathbf{R} = \mathbf{A}^H \mathbf{A} = \sum_{n=1}^N \mathbf{x}_n \mathbf{x}_n^H, \quad (10.37)$$

gdzie  $\mathbf{x}_N = \begin{bmatrix} x[n] \\ x[n-1] \\ \vdots \\ x[n-M+1] \end{bmatrix}$  oznacza wektor sygnału wejściowego do chwili  $N$ . Współczynniki filtra dane są zależnością (10.23):

$$\mathbf{h} = \mathbf{R}^{-1} \mathbf{A}^H \mathbf{d} = \mathbf{R}^{-1} \mathbf{p}. \quad (10.38)$$

Współczynniki filtra  $\mathbf{h}$  (10.38) można wyznaczać adaptacyjnie dla każdej nowej próbki sygnału wejściowego  $x[n]$ , tzn. dla ... $\mathbf{x}_{N-2}$ ,  $\mathbf{x}_{N-1}$ ,  $\mathbf{x}_N$ , jednak taki algorytm nie jest wydajny obliczeniowo, ponieważ w każdej iteracji trzeba odwrócić macierz  $\mathbf{R}$ . W celu obniżenia złożoności obliczeniowej odwrotność macierzy  $\mathbf{R}$  jest wyznaczana rekurencyjnie.

Przedstawmy macierz  $\mathbf{R}$  w postaci sumy dwóch macierzy:

$$\mathbf{R}_N = \sum_{n=1}^N \mathbf{x}_n \mathbf{x}_n^H = \sum_{n=1}^{N-1} \mathbf{x}_n \mathbf{x}_n^H + \mathbf{x}_N \mathbf{x}_N^H = \mathbf{R}_{N-1} + \mathbf{x}_N \mathbf{x}_N^H. \quad (10.39)$$

Indeksy przy macierzach  $\mathbf{R}$  oznaczają chwile czasu, do których uwzględniono dane przy wyznaczaniu tych macierzy. Macierz  $\mathbf{R}_N$  (obliczona z danych do chwili  $N$ ) jest uzyskiwana z macierzy  $\mathbf{R}_{N-1}$  (dla poprzedniej chwili czasowej) przez uaktualnienie macierzą pierwszego rzędu. Stosując do (10.39) rekurencyjny wzór Shermana-Morrisona otrzymujemy:

$$\mathbf{R}_N^{-1} = \mathbf{R}_{N-1}^{-1} - \frac{\mathbf{R}_{N-1}^{-1} \mathbf{x}_N \mathbf{x}_N^H \mathbf{R}_{N-1}^{-1}}{1 + \mathbf{x}_N^H \mathbf{R}_{N-1}^{-1} \mathbf{x}_N}. \quad (10.40)$$

Dla uproszczenia notacji przyjmijmy  $\mathbf{P}_N = \mathbf{R}_N^{-1}$  i

$$\mathbf{k}_N = \frac{\mathbf{R}_{N-1}^{-1} \mathbf{x}_N}{1 + \mathbf{x}_N^H \mathbf{R}_{N-1}^{-1} \mathbf{x}_N}, \quad (10.41)$$

wówczas (10.40) przyjmuje postać:

$$\mathbf{P}_N = \mathbf{P}_{N-1} - \mathbf{k}_N \mathbf{x}_N^H \mathbf{P}_{N-1}. \quad (10.42)$$

Wektor  $\mathbf{k}_N$  jest nazywany wzmocnieniem Kalmana.

Współczynniki filtra dane są zależnością (10.23):

$$\mathbf{h}_N = \mathbf{P}_N \mathbf{p}_N \quad (10.43)$$

ponieważ

$$\mathbf{p}_N = \mathbf{A}^H \mathbf{d} = \sum_{n=1}^N \mathbf{x}_n d_n = \mathbf{p}_{N-1} + \mathbf{x}_N d_N, \quad (10.44)$$

więc

$$\mathbf{h}_N = \mathbf{P}_N (\mathbf{p}_{N-1} + \mathbf{x}_N d_N). \quad (10.45)$$

Mnożąc obie strony równania (10.42) przez  $\mathbf{p}_{N-1}$  otrzymujemy:

$$\mathbf{P}_N \mathbf{p}_{N-1} = \mathbf{P}_{N-1} \mathbf{p}_{N-1} - \mathbf{k}_N \mathbf{x}_N^H \mathbf{P}_{N-1} \mathbf{p}_{N-1} = \mathbf{h}_{N-1} - \mathbf{k}_N \mathbf{x}_N^H \mathbf{h}_{N-1}. \quad (10.46)$$

Podstawiając (10.46) do (10.45) oraz wykorzystując  $\mathbf{k}_N = \mathbf{P}_N \mathbf{x}_N$  otrzymujemy:

$$\mathbf{h}_N = \mathbf{h}_{N-1} - \mathbf{k}_N \mathbf{x}_N^H \mathbf{h}_{N-1} + \mathbf{P}_N \mathbf{x}_N d_N = \mathbf{h}_{N-1} - \mathbf{k}_N \mathbf{x}_N^H \mathbf{h}_{N-1} + \mathbf{k}_N d_N = \mathbf{h}_{N-1} + \mathbf{k}_N (d_N - \mathbf{x}_N^H \mathbf{h}_{N-1}). \quad (10.47)$$

Wyrażenie w nawiasie (10.47) jest błędem dopasowania sygnału wyjściowego z filtra wyznaczonego dla czasu  $N-1$  do sygnału odniesienia w czasie  $N$ :

$$e_N = d_N - \mathbf{x}_N^H \mathbf{h}_{N-1}. \quad (10.48)$$

Implementacja algorytmu RLS wymaga znajomości wartości początkowej macierzy odwrotnej  $\mathbf{P}$ , przyjmuje się, że jest nią macierz diagonalna z wartościami  $\delta^{-1}$  na przekątnej, gdzie  $\delta$  jest małą wartością dodatnią.

Podsumowując, algorytm RLS jest następujący:

#### I. Inicjalizacja

1. Wybór długości filtra  $M$ ,
2. Początkowe współczynniki filtra można ustawić na zero  $\mathbf{h}_0 = \mathbf{0}$ ,
3. Początkowa macierz odwrotna  $\mathbf{P}_0 = \delta^{-1} \mathbf{I}$ , gdzie  $\delta$  jest małą wartością dodatnią.

#### IIa. Obliczenia dla $N=1,2,3,\dots$

1.  $\mathbf{k}_N = \frac{\mathbf{P}_{N-1} \mathbf{x}_N}{1 + \mathbf{x}_N^H \mathbf{P}_{N-1} \mathbf{x}_N},$
2.  $e_N = d_N - \mathbf{x}_N^H \mathbf{h}_{N-1},$
3.  $\mathbf{h}_N = \mathbf{h}_{N-1} + \mathbf{k}_N e_N,$
4.  $\mathbf{P}_N = \mathbf{P}_{N-1} - \mathbf{k}_N \mathbf{x}_N^H \mathbf{P}_{N-1}.$

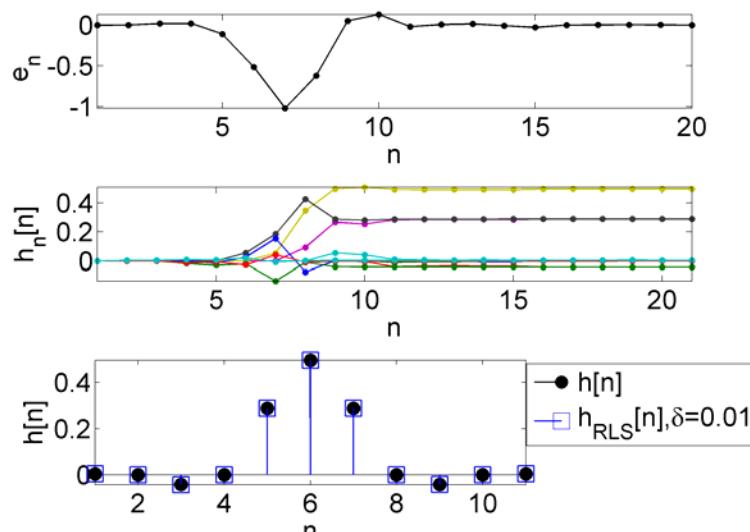
W przypadku, kiedy filtr RLS wykorzystywany jest do śledzenia parametrów zmiennych w czasie, stosuje się wykładnicze zapominanie najstarszych pomiarów i obliczenia przebiegają następująco:

#### IIb. Obliczenia dla $N=1,2,3,\dots$

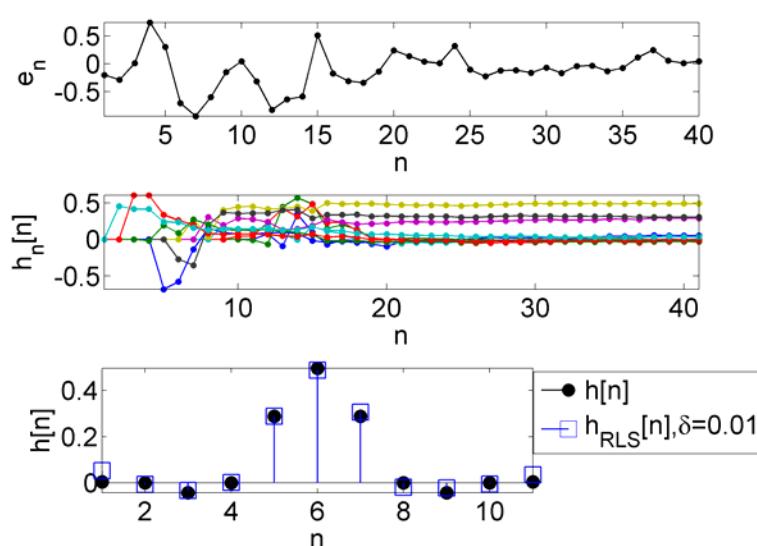
1.  $\mathbf{k}_N = \frac{\lambda^{-1} \mathbf{P}_{N-1} \mathbf{x}_N}{1 + \lambda^{-1} \mathbf{x}_N^H \mathbf{P}_{N-1} \mathbf{x}_N},$
2.  $e_N = d_N - \mathbf{x}_N^H \mathbf{h}_{N-1},$
3.  $\mathbf{h}_N = \mathbf{h}_{N-1} + \mathbf{k}_N e_N,$
4.  $\mathbf{P}_N = \lambda^{-1} \mathbf{P}_{N-1} - \lambda^{-1} \mathbf{k}_N \mathbf{x}_N^H \mathbf{P}_{N-1},$

gdzie  $\lambda$  jest współczynnikiem zapominania o wartości równej 1 lub niewiele mniejszej. Dla  $\lambda=1$  filtr RLS jest równoważny obliczeniom blokowym LS, a dla  $\lambda<1$  najstarsze pomiary (próbki) są zaniedbywane wykładniczo.

Rys.10.6 przedstawia wyniki identyfikacji odpowiedzi impulsowej układu LTI za pomocą filtra RLS. Wejściowym sygnałem testowym  $x[n]$  był szum o rozkładzie normalnym, a sygnałem odniesienia  $d[n]$  odpowiedź układu identyfikowanego na to wymuszenie zarejestrowana bez zakłóceń. Rys.10.6 (góra) przedstawia błąd dopasowania (10.48) w kolejnych chwilach czasu. Błąd ten szybko maleje, co oznacza, że sygnały  $x[n]$  i  $d[n]$  zostały dopasowane. Rys.10.7 (środek) przedstawia współczynniki filtra w kolejnych chwilach czasu. Algorytm został uruchomiony z początkowymi współczynnikami filtra równymi zero; w kolejnych iteracjach współczynniki filtra szybko osiągają wartości zadane. Rys.10.6 (dół) przedstawia odpowiedź impulsową układu identyfikowanego  $h[n]$  i współczynniki odpowiedzi impulsowej wyznaczone filtrem RLS w  $N=21$  iteracji.



Rys.10.6 Identyfikacja odpowiedzi impulsowej filtrem RLS - pomiar bez zakłóceń.



Rys.10.7 Identyfikacja odpowiedzi impulsowej filtrem RLS - pomiar zakłócony szumem addytywnym.

Rys.10.7 przedstawia wykresy analogiczne do rys.10.6 dla przypadku, kiedy sygnał odniesienia  $d[n]$  został zarejestrowany z zakłóceniem w postaci szumu addytywnego. Rys.10.7 (dół) przedstawia odpowiedź impulsową układu identyfikowanego  $h[n]$  i współczynniki odpowiedzi impulsowej wyznaczone filtrem RLS w  $N=41$  iteracji.

#### 10.4 Estymacja w sensie minimum średnich kwadratów

Estymacja w sensie najmniejszych kwadratów minimalizuje sumę kwadratów błędów bez dodatkowych założeń. Estymacja w sensie minimum średnich kwadratów MMS (*Minimum Mean-Square*) zakłada model probabilistyczny danych. Optymalny filtr MMS jest nazywany filtrem Wienera. Różnica pomiędzy filtrem LS i MMS polega na tym, że w drugim przypadku stosuje się wartość oczekiwana z iloczynu skalarnego, a nie iloczyn skalarny.

Zadanie filtracji optymalnej sformułowane jest jak poprzednio, tzn. należy wyznaczyć współczynniki odpowiedzi impulsowej filtra  $h[n]$  o długości  $M$ , dla których sygnał wyjściowy filtra  $y[n]$  jest dopasowany do sygnału pożądanego (odniesienia)  $d[n]$ .

$$y[n] = \sum_{k=0}^{M-1} h[k]x[n-k]. \quad (10.49)$$

Zakładamy, że sygnały są stacjonarne w szerokim sensie, tzn.:

$$E(x[n]) = E(x[n-k]) \text{ dla każdego } k \quad (10.50)$$

i

$$E(x[n]x^*[n-k]) \quad (10.51)$$

zależy tylko od przesunięcia w czasie  $k$ , a nie zależy od chwili czasu  $n$ .

Błąd estymacji wynosi  $e[n]=d[n]-y[n]$ . Zgodnie z zasadą ortogonalności (10.11) błąd średniokwadratowy w postaci

$$\|e[n]\|^2 = E(e[n]e^*[n]) \quad (10.52)$$

osiąga minimum, kiedy jest on ortogonalny do danych, tzn.:

$$\left\langle d[n] - \sum_{m=0}^{M-1} h[m]x[n-m], x[n-k] \right\rangle = 0, \quad k = 0, 1, \dots, \quad (10.53)$$

lub

$$\langle d[n], x[n-k] \rangle = \sum_{m=0}^{M-1} h[m] \langle x[n-m], x[n-k] \rangle. \quad (10.54)$$

Zastępując iloczyny skalarse w (10.54) operatorami wartości oczekiwanej otrzymujemy równania normalne nazywane równaniami Wienera-Hopfa w postaci:

$$\sum_{m=0}^{M-1} h[m] E(x[n-m]x^*[n-k]) = E(x^*[n-k]d[n]). \quad (10.55)$$

Równania (10.55) można przepisać w postaci:

$$\sum_{m=0}^{M-1} h[m]r[k-m] = p[k], \quad k = 0, 1, \dots, \quad (10.55)$$

gdzie:

$$r[k-m] = E(x[n-m]x^*[n-k]), \quad (10.56)$$

$$p[k] = E(x^*[n-k]d[n]), \quad (10.57)$$

Macierz (10.56) jest macierzą autokorelacji sygnału wejściowego, a wektor (10.57) jest wektorem korelacji wzajemnej pomiędzy sygnałem wejściowym a sygnałem odniesienia.

Dla filtra FIR o  $M$  współczynnikach splot można zapisać w postaci:

$$y[n] = \mathbf{x}^H \mathbf{h}. \quad (10.58)$$

gdzie:

$$\mathbf{x}[n] = [x^*[n] \quad x^*[n-1] \quad \dots \quad x^*[n-M+1]]^T, \quad (10.59)$$

$$\mathbf{h} = [h[0] \quad h[1] \quad \dots \quad h[M-1]]^T. \quad (10.60)$$

Równania (10.55) można zapisać w postaci macierzowej:

$$\mathbf{R}\mathbf{h} = \mathbf{p}, \quad (10.61)$$

gdzie:

$$\mathbf{R} = \begin{bmatrix} r[0] & r^*[1] & r^*[2] & \dots & r^*[M-1] \\ r[1] & r[0] & r^*[1] & \dots & r^*[M-2] \\ r[2] & r[1] & r[0] & \dots & r^*[M-3] \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ r[M-1] & r[M-2] & r[M-3] & \dots & r[0] \end{bmatrix} = E(\mathbf{x}[n]\mathbf{x}^H[n]). \quad (10.63)$$

i

$$\mathbf{p} = \begin{bmatrix} p(0) \\ p(1) \\ \vdots \\ p(M-1) \end{bmatrix} = E(\mathbf{x}[n]d[n]). \quad (10.64)$$

Optymalne współczynniki filtra są, więc dane zależnością:

$$\mathbf{h} = \mathbf{R}^{-1} \mathbf{p}. \quad (10.65)$$

Macierz  $\mathbf{R}$  jest macierzą Toeplitza, ponieważ przekątne są sobie równe i istnieją szybkie algorytmy odwracania macierzy w takiej postaci.

Porównując metody LS i MMS można stwierdzić, że:

1. W metodzie LS wykorzystywane są tylko dostępne dane, nie zakłada się żadnego modelu statystycznego, macierz  $\mathbf{R}$  może nie być macierzą Toeplitza.
2. W metodzie MMS zakłada się model statystyczny sygnału, tzn. muszą być dostępne współczynniki autokorelacji i korelacji wzajemnej. Współczynniki te można określić na podstawie wiedzy o sygnale, lub mogą być one estymowane na podstawie danych pomiarowych. Przykładem estymatora (obciążonego) autokorelacji jest:

$$\hat{r}[n] = \frac{1}{N} \sum_{k=n}^{N-1} x[k]x^*[k-n], \quad n = 0, 1, \dots, N-1 \quad (10.66)$$

3. W przypadku, gdy do wyznaczenia estymat współczynników korelacji i współczynników filtra używa się tych samych danych pomiarowych, metoda MMS jest równoważna LS.

## 10.5 Filtry adaptacyjne LMS

Dla kryterium MMS można sformułować funkcję kosztu, korzystając z (10.58), w postaci:

$$J[\mathbf{h}] = E((d[n] - y[n])^2) = E((d[n] - \mathbf{x}^H \mathbf{h})^2) = \sigma_d^2 - \mathbf{p}^T \mathbf{h} - \mathbf{h}^T \mathbf{p} + \mathbf{h}^T \mathbf{R} \mathbf{h}. \quad (10.67)$$

gdzie:  $\mathbf{R}$  i  $\mathbf{p}$  są zdefiniowane przez (10.63) i (10.64) i  $E(d^2[n]) = \sigma_d^2$ .

Do minimalizacji funkcji kosztu (10.67) względem wektora współczynników filtra  $\mathbf{h}$  stosuje się algorytm najszybszego spadku (*steepest descent*). Gradient funkcji wskazuje kierunek jej najszybszego wzrostu, ponieważ szukamy minimum funkcji kosztu, więc współczynniki filtra są uaktualniane w przeciwnym kierunku:

$$\mathbf{h}_N = \mathbf{h}_{N-1} - \frac{1}{2} \mu \frac{\partial J(\mathbf{h}_{N-1})}{\partial \mathbf{h}_{N-1}}, \quad (10.68)$$

gdzie  $\mu$  jest parametrem określającym wielkość kroku optymalizacji. Wartość  $\mu$  wpływa na szybkość zbieżności algorytmu, dla zbyt małych  $\mu$  zbieżność jest wolna, a dla zbyt dużych  $\mu$  algorytm nie jest zbieżny.

Gradient w (10.68) wynosi:

$$\frac{\partial J(\mathbf{h}_{N-1})}{\partial \mathbf{h}_{N-1}} = -2\mathbf{p} + 2\mathbf{R}\mathbf{h}. \quad (10.69)$$

Uaktualnione współczynniki filtra dane są, więc zależnością:

$$\mathbf{h}_N = \mathbf{h}_{N-1} + \mu(\mathbf{p} - \mathbf{R}\mathbf{h}_{N-1}). \quad (10.70)$$

Niestety, zależność (10.70) nie ma znaczenia praktycznego, ponieważ wymaga znajomości współczynników korelacji, tj. macierzy  $\mathbf{R}$  i wektora  $\mathbf{p}$ . W praktyce stosuje się następujące estymatory tych macierzy:

$$\hat{\mathbf{R}} = \mathbf{x}_N \mathbf{x}_N^H, \quad (10.71)$$

i

$$\hat{\mathbf{p}} = \mathbf{x}_N d_N, \quad (10.72)$$

czyli zaniedbuje się operator wartości oczekiwanej.

Podstawiając (10.71) i (10.72) do (10.70) otrzymujemy algorytm LMS (*Least Mean Square*) uaktualniania współczynników filtra:

$$\mathbf{h}_N = \mathbf{h}_{N-1} + \mu \mathbf{x}_N (d_N - \mathbf{x}_N^T \mathbf{h}_{N-1}) = \mathbf{h}_{N-1} + \mu \mathbf{x}_N e_N, \quad (10.73)$$

gdzie  $e_N$  jest błędem pomiędzy wyjściem filtra a sygnałem odniesienia.

Filtry LMS charakteryzują się niską złożonością obliczeniową.

W zastosowaniach praktycznych stosuje się również algorytm NLMS (*Normalized LMS*), w którym wartość kroku  $\mu$  wyznacza się w każdej iteracji algorytmu na podstawie sygnału wejściowego wg zależności:

$$\mu_N = \frac{\mu}{a + \mathbf{x}_N^T \mathbf{x}_N}, \quad (10.74)$$

gdzie  $\mu$  jest wartością stałą, a  $a$  małą wartością dodatnią zabezpieczającą przed dzieleniem przez zero.

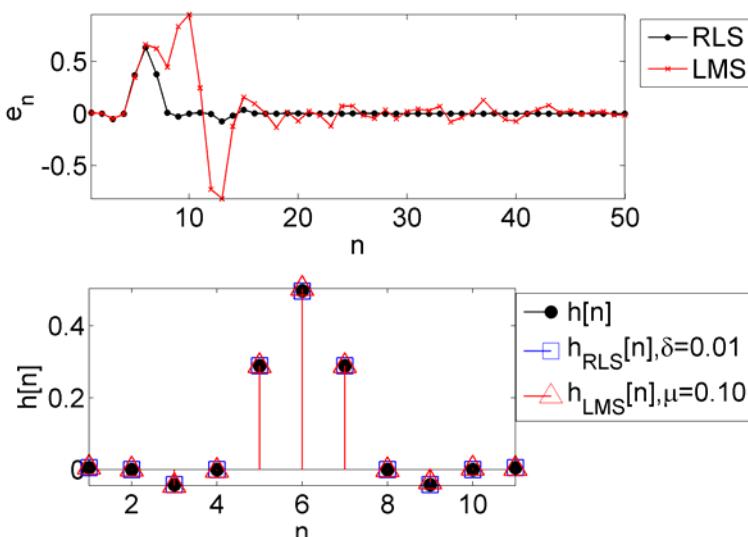
Podsumowując, algorytm LMS jest następujący:

I. Inicjalizacja

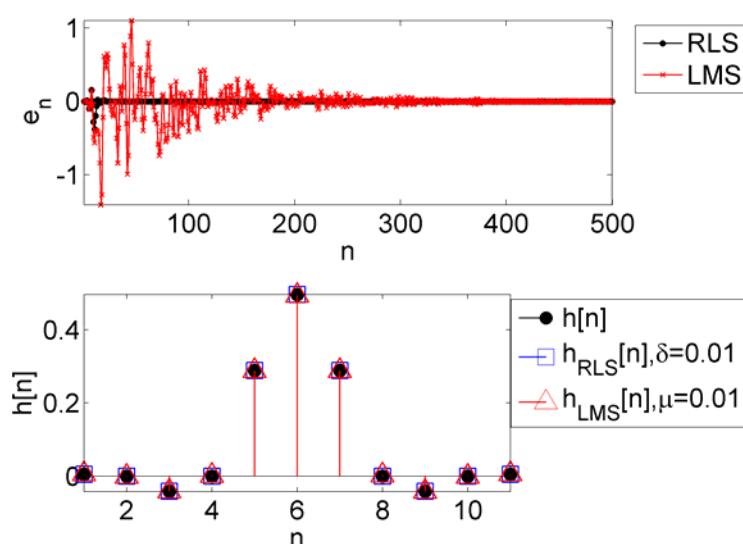
1. Wybór długości filtra  $M$ ,
2. Początkowe współczynniki filtra można ustawić na zero  $\mathbf{h}_0 = \mathbf{0}$ ,
3. Wybór wartości  $\mu$ .

II. Obliczenia dla  $N=1,2,3,\dots$

1.  $e_N = d_N - \mathbf{x}_N^T \mathbf{h}_{N-1}$ ,
2.  $\mathbf{h}_N = \mathbf{h}_{N-1} + \mu \mathbf{x}_N e_N$ .



Rys.10.8 Identyfikacja odpowiedzi impulsowej filtrami LMS i RLS - pomiar bez zakłóceń.



Rys.10.9 Identyfikacja odpowiedzi impulsowej filtrami LMS i RLS - pomiar bez zakłóceń.

Rys.10.8 i rys.10.9 przedstawiają wyniki identyfikacji odpowiedzi impulsowej układu LTI za pomocą filtra LMS dla  $\mu=0.1$  i  $\mu=0.01$ . Dla porównania przedstawiono również wyniki

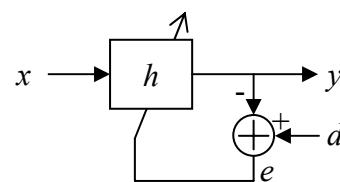
uzyskane algorytmem RLS. Widoczny jest silny wpływ wartości  $\mu$  na szybkość zbieżności algorytmu LMS.

## 10.6 Zastosowania filtrów adaptacyjnych

Ogólny schemat filtru adaptacyjnego przedstawiony jest na rys.10.10. Filtr adaptacyjny jest to filtr, przeważnie typu FIR, którego współczynniki  $h[n]$  są przestrajane w trakcie filtracji w taki sposób, aby sygnał wyjściowy filtra  $y[n]$  był dopasowany do sygnału pożądanego (*desired*)  $d[n]$  (nazywanego również sygnałem odniesienia), tzn., aby błąd zdefiniowany jako:

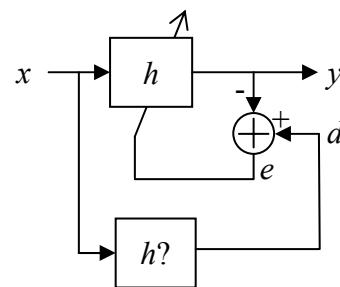
$$e[n] = d[n] - y[n], \quad (10.75)$$

był jak najmniejszy w sensie przyjętej metryki. Przykładami filtrów adaptacyjnych są filtry RLS i LMS. Poniżej podano typowe zastosowania filtrów adaptacyjnych.



Rys.10.10 Schemat filtru adaptacyjnego.

1. Identyfikacja systemu (np. odpowiedzi impulsowej kanału telekomunikacyjnego, toru pomiarowego, pomieszczenia odsłuchowego, itp.). Schemat włączenia filtru adaptacyjnego pokazany jest na rys.10.11. Sygnałem odniesienia jest sygnał wyjściowy identyfikowanego systemu. Testowym sygnałem wejściowym  $x[n]$  może być szum szerokopasmowy (stosowany np. w modemach). Dla małych wartości błędu  $e[n]$  filtr adaptacyjny zachowuje się jak nieznany system, tzn. odpowiedź impulsowa filtru adaptacyjnego jest w przybliżeniu równa odpowiedzi impulsowej identyfikowanego układu.



Rys.10.11 Identyfikacja odpowiedzi impulsowej.

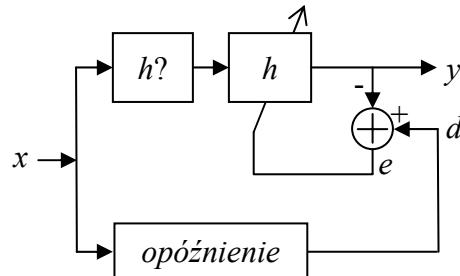
2. Wyznaczenie filtru odwrotnego. Schemat włączenia filtru adaptacyjnego pokazany jest na rys.10.12. Filtr adaptacyjny jest włączany za badanym systemem. Wprowadzenie dodatkowego opóźnienia (równego opóźnieniu systemu) pozwala zachować przyczynowość układu. Sygnałem odniesienia jest sygnał wejściowy, tzn. kaskadowe połączenie nieznanego układu i filtru adaptacyjnego powinno być przeźroczyste (wszechprzepustowe). Następnie filtr adaptacyjny można wykorzystać do korekcji zniekształceń wprowadzanych przez system (np. do korekcji toru pomiarowego czy linii telefonicznej).

3. Usuwanie zakłóceń. Schemat włączenia filtra adaptacyjnego pokazany jest na rys.10.13. Sygnałem odniesienia dla układów redukcji zakłóceń jest zakłócony sygnał  $d[n]=x[n]+n[n]$ , natomiast wejściem filtra jest zakłócenie  $w[n]$  skorelowane z  $n[n]$ . Sygnały  $n[n]$  i  $w[n]$ , reprezentują zakłócenie mierzone w różnych punktach obiektu i ich przebiegi czasowe nie są takie same. W tym przypadku wyjściem układu jest sygnał błędu  $e[n]$ , który dąży do  $x[n]$ .

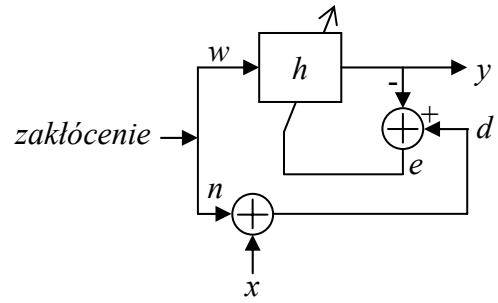
W praktyce często sygnał pomiarowy musi być zarejestrowany wraz z silnym zakłóceniem. W wielu sytuacjach istnieje jednak możliwość osobnej rejestracji zakłócenia, w takim przypadku zakłócenie może zostać usunięte adaptacyjnie w układzie z rys.10.13. Rozważmy kilka przykładów:

- a) Pomiar EKG płodu w łonie matki jest silnie zakłócony przez EKG matki. W tym przypadku sygnałem wejściowym dla filtra adaptacyjnego jest EKG matki, a sygnałem odniesienia suma sygnałów EKG matki i EKG płodu. W wyniku filtracji adaptacyjnej sygnał błędu będzie zawierał tylko sygnał EKG płodu.
- b) System głośnomówiący w samochodzie (helikopterze, czołgu lub innym pojazdzie). Sygnał mowy jest rejestrowany wraz z zakłóceniami spowodowanymi np. pracą silnika. Zakłócenia te można rejestrować osobnym mikrofonem, wówczas wejściem filtra są same zakłócenia, a sygnałem odniesienia suma zakłóceń i sygnału mowy. Sygnał błędu będzie zawierał tylko sygnał mowy.
- c) Usuwanie echa, pogłosu. Przy audio telekonferencjach wymagających jednoczesnej (*full-duplex transmission*) komunikacji sygnał z mikrofonu zawiera dwie składowe: sygnał bliski (*near-end*) mowy i daleki (*far-end*) z głośnika, dodatkowo sygnał z głośnika splata się z odpowiedzią impulsową pomieszczenia. W takim przypadku wejściem filtra adaptacyjnego jest sygnał *far-end* rejestrowany osobnym mikrofonem, a sygnałem odniesienia jest sygnał *near-end*. Sygnał błędu będzie zawierał tylko sygnał mowy.

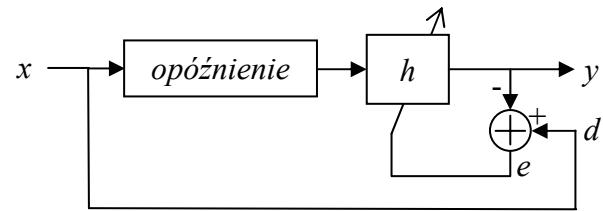
4. Predykcja. W układzie na rys.10.14 filtr adaptacyjny przewiduje bieżącą próbkę na podstawie próbek poprzednich. Sygnał  $x[n]$  musi być wolnozmienny. Układ taki można zastosować do usunięcia składowej deterministycznej z sygnału losowego. Predykcja adaptacyjna ma zastosowanie np. w kompresji danych, estymacji widmowej czy rozpoznawaniu.



Rys.10.12 Wyznaczenie filtra odwrotnego.



Rys.10.13 Usuwanie zakłóceń.



Rys.10.14 Predykcja.



## 11. Transformacja falkowa

### 11.1 Ciągła transformacja falkowa

Ciągła transformacja falkowa CWT (*Continuous Wavelet Transform*) sygnału  $x(t)$  zdefiniowana jest następująco:

$$CWT_x(u,s) = \int_{-\infty}^{\infty} x(t) \psi_{u,s}^*(t) dt , \quad (11.1)$$

gdzie:

$$\psi_{u,s}(t) = \frac{1}{\sqrt{s}} \psi\left(\frac{t-u}{s}\right), \quad u, s \in R, \quad s > 0 , \quad (11.2)$$

oraz musi być spełniony warunek (*admissibility condition*):

$$C_\psi = \int_{-\infty}^{\infty} \frac{|\Psi(\Omega)|^2}{|\Omega|} d\Omega < \infty . \quad (11.3)$$

We wzorze (11.3) zapis  $\Psi(\Omega)$  oznacza transformatę Fouriera funkcji  $\psi(t)$ . Z warunku (11.3) wynika, że musi zachodzić  $\Psi(0) = 0$ , a więc wartość średnia funkcji  $\psi(t)$  musi wynosić zero:

$$\int_{-\infty}^{\infty} \psi(t) dt = \Psi(0) = 0 . \quad (11.4)$$

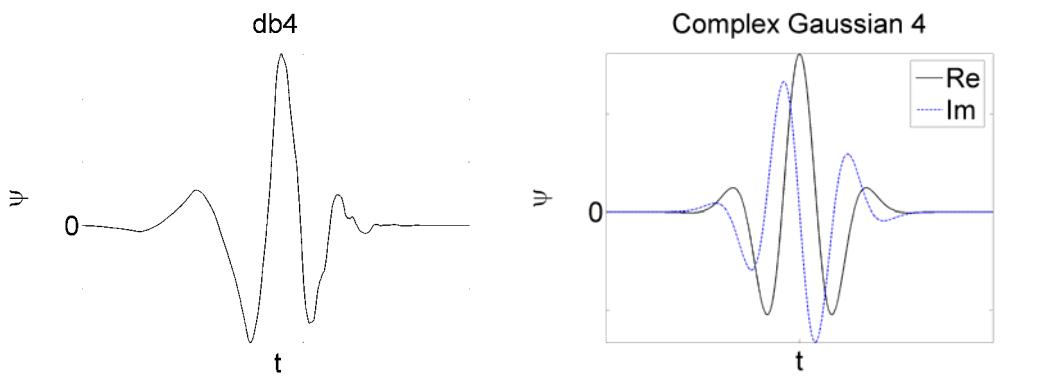
Funkcja  $\psi(t)$  jest nazywana falką matczyną (*mother wavelet*) lub główną, ma ona zerową wartość średnią i tzw. nośnik zwarty, tzn. jej czas trwania jest skończony. Gwiazdka w definicji (11.1) oznacza sprzężenie zespolone. We wzorze (11.2) występuje współczynnik skali  $s$ , który powoduje zmianę czasu trwania (rozciąganie lub ściskanie) falki, oraz współczynnik przesunięcia  $u$ , który zmienia położenie falki na osi czasu. Zgodnie z własnością transformacji Fouriera, rozciąganiu i ściskaniu falki w dziedzinie czasu towarzyszy odwrotna zmiana jej widma  $\psi(st) \leftrightarrow 1/|s| |\Psi(f/s)|$ , czyli odpowiednio jego zawężanie i rozszerzanie. Dla ustalonej skali  $s$  równanie (11.1) jest splotem sygnału  $x(t)$  z filtrem pasmowoprzepustowym  $\psi_s(t) = 1/\sqrt{s} \psi^*(-t/s)$ .

Wynikiem CWT jest płaszczyzna  $CWT_x(u,s)$  reprezentująca sygnał  $x(t)$  w układzie czas-skala, a ponieważ współczynnik skalujący  $s$  określa pasmo częstotliwości falki, więc  $CWT_x(u,s)$  jest również płaszczyzną czas-częstotliwość. Dla falek o wartościach zespolonych płaszczyzna  $CWT_x(u,s)$  jest również zespolona i może być reprezentowana w postaci charakterystyki amplitudowej i fazowej podobnie jak krótkoczasowa transformacja Fouriera STFT. Dla falek o wartościach rzeczywistych (tj. nie zespolonych) płaszczyzna  $CWT_x(u,s)$  reprezentuje wyniki filtracji pasmowej w czasie. Rys.11.1 przedstawia przykładową falkę o wartościach rzeczywistych o nazwie Daubechies 4 oznaczaną jako db4, gdzie liczba 4 oznacza rząd falki i zespoloną falkę Gaussa 4-tego rzędu. Obie falki na rys.11.1 mają wartość średnią równą zero i wartości niezerowe tylko w skończonym przedziale czasu.

Współczynniki  $CWT_x(u,s)$  można również wyznaczać dla sygnałów dyskretnych o skończonej długości. W takim przypadku współczynnik przesunięcia  $u$  można zastąpić indeksem  $n$ , ponieważ w trakcie analizy falka jest przesuwana wzdłuż sygnału o jedną próbkę. Rys.11.2 przedstawia STDFT sygnału z liniową modulacją częstotliwości w postaci  $x[n] = \cos(kn^2)$ ,  $k = \pi/(2N)$ ,  $n = 0, 1, 2, \dots, N-1$ . Rys.11.3 przedstawia współczynniki falkowe

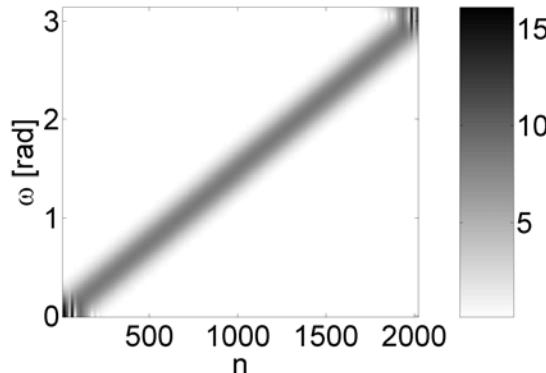
$CWT_x(n,s)$  tego samego sygnału dla analizy falką rzeczywistą i zespoloną. Współczynniki skali  $s$  w CWT dobierane są w sposób arbitralny, w prezentowanym przykładzie wynoszą one  $s=1:0.1:30$ . W przypadku falki rzeczywistej płaszczyzna  $CWT_x(n,s)$  nie reprezentuje obwiedni sygnału, jak ma to miejsce dla STDFT i falek zespolonych, tylko wynik filtracji pasmowej i współczynniki falkowe mogą mieć małe wartości nawet w przypadku, gdy w sygnale występuje sinusoida o dużej amplitudzie.

Porównując rys.11.2 i rys.11.3 widzimy, że zależność współczynnika skali  $s$  jest odwrotnie proporcjonalna do częstotliwości i jest nieliniowa.

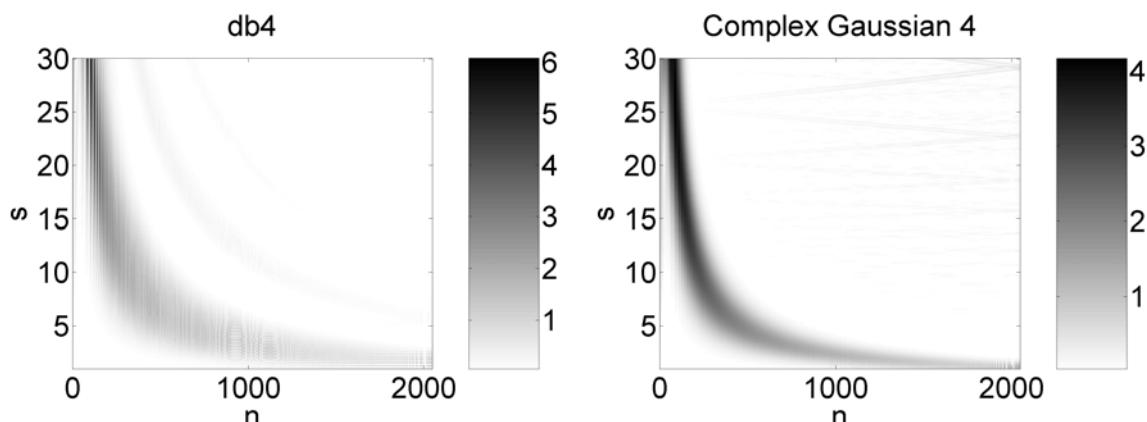


Rys.11.1 Falka Daubechies 4 i zespolona falka Gaussa 4-tego rzędu..

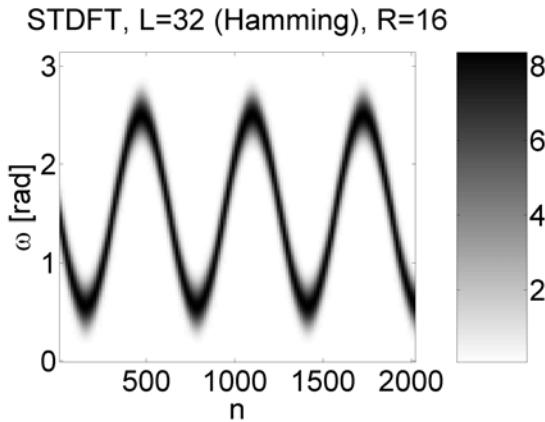
STDFT, L=32 (Hamming), R=16



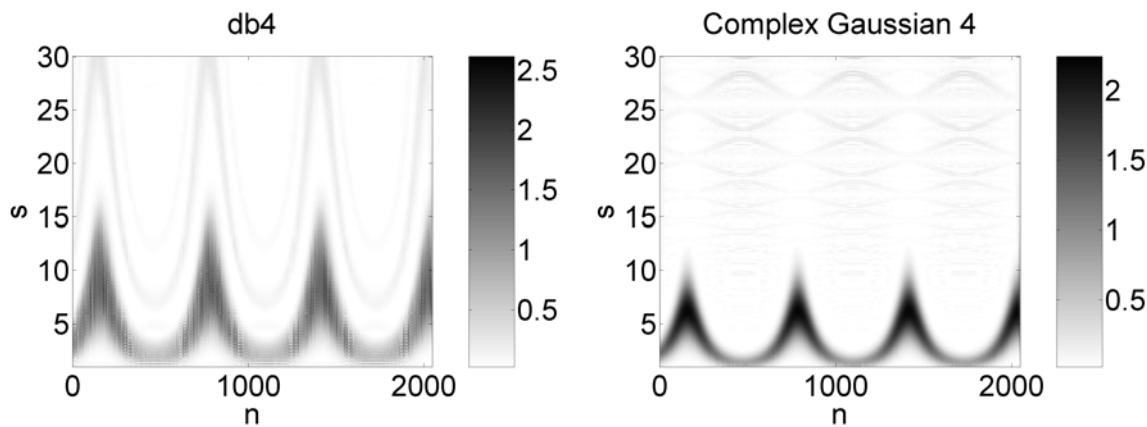
Rys.11.2 Krótkoczasowa, dyskretna transformacja Fouriera (9.65) sygnału sinusoidalnego z liniową modulacją częstotliwości.



Rys.11.3 Współczynniki falkowe  $CWT_x(n,s)$  sygnału sinusoidalnego z liniową modulacją częstotliwości,  $s=1:0.1:30$ .



Rys.11.4 Krótkoczasowa, dyskretna transformacja Fouriera (9.65) sygnału sinusoidalnego z sinusoidalną modulacją częstotliwości.



Rys.11.5 Współczynniki falkowe  $CWT_x(n,s)$  sygnału sinusoidalnego z sinusoidalną modulacją częstotliwości,  $s=1:0.1:30$ .

Rys.11.4 przedstawia STDFT sygnału z sinusoidalną modulacją częstotliwości. Współczynniki falkowe dla tego sygnału pokazane są na rys.11.5.

Odwrotną, ciągłą transformację falkową ICWT wyznacza się następująco:

$$x(t) = \frac{1}{C_\psi} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} CWT_x(u,s) \psi_{u,s}(t) \frac{ds du}{s^2} \quad (11.5)$$

gdzie  $C_\psi$  jest określone wzorem (11.3).

CWT ma następujące własności:

1. Zachowanie energii sygnału:

$$\int_{-\infty}^{\infty} |x(t)|^2 dt = \frac{1}{C_\psi} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} |CWT_x(u,s)|^2 \frac{ds du}{s^2}, \quad (11.6)$$

2. Przesunięcie:

$$x(t) = y(t - u') \Leftrightarrow CWT_x(u,s) = CWT_y(u - u',s), \quad (11.7)$$

3. Przeskalowanie:

$$x(t) = \frac{1}{\sqrt{a}} y\left(\frac{t}{a}\right) \Leftrightarrow CWT_x(u,s) = CWT_y\left(\frac{u}{a}, \frac{s}{a}\right). \quad (11.8)$$

## 11.2 Dyskretna transformacja falkowa

W dyskretnej transformacji falkowej oznaczanej jako DWT (*Discrete Wavelet Transform*) dobór współczynników skali i przesunięcia nie jest arbitralny. Współczynniki te są zmieniane, na podstawie potęgi liczby 2, wg zależności:

$$s = 2^j, u = 2^j k = sk \quad j, k \in C . \quad (11.9)$$

Podstawiając (11.9) do (11.2) otrzymujemy:

$$\psi_{k,j}[n] = 2^{-j/2} \psi[2^{-j} n - k], \quad j, k \in C . \quad (11.10)$$

W DWT wprowadza się dodatkowo funkcję o charakterze dolnoprzepustowym, tzw. funkcję skalującą:

$$\phi_{k,j}[n] = 2^{-j/2} \phi[2^{-j} n - k], \quad j, k \in C . \quad (11.11)$$

Analogicznie do równania (11.5), stosując oznaczenia (11.10) i (11.11), można przedstawić sygnał  $x[n]$  na poziomie dekompozycji  $I$  w postaci liniowej kombinacji funkcji skalującej i falek:

$$x[n] = \sum_k c_I[k] \phi_{k,I}[n] + \sum_k \sum_{j=1}^I d_j[k] \psi_{k,j}[n] . \quad (11.12)$$

Dekompozycja sygnału  $x[n]$  jest przeprowadzana w oparciu o dwie spokrewnione ze sobą funkcje bazowe: falkę  $\psi[n]$  o wartości średniej równej zero, oraz funkcję skalującą  $\phi[n]$ , która może przenosić składową stałą. Ponieważ falka  $\psi[n]$  ma charakter pasmowoprzepustowy, więc współczynniki  $d_{j,k}$ , tzw. detale, zawierają informację o wyższych częstotliwościach. Natomiast współczynniki  $c_k$ , tzw. aproksymacje lub sygnał zgrubny (*coarse*), zawierają informację dolnoprzepustową wraz ze składową stałą.

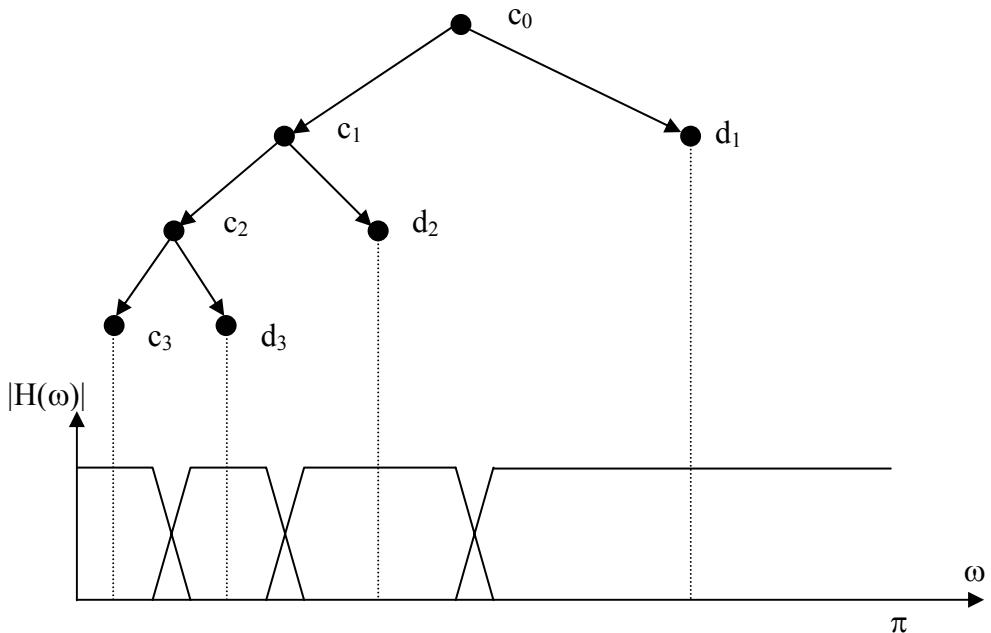
DWT umożliwia tzw. analizę wielopoziomową, której schematyczną reprezentację w postaci drzewa binarnego przedstawia rys.11.6. Sygnał aproksymacji  $c_j$  na kolejnych poziomach analizy jest dekomponowany na sygnał aproksymacji i sygnał detali. Rys.11.6 przedstawia również interpretację częstotliwościową dekompozycji falkowej. Na każdym poziomie widmo sygnału jest dzielone na pół, tj. na część dolnoprzepustową (aproksymację) i górnoprzepustową (detale). Dla przedstawionej na rys.11.6 dekompozycji trzypoziomowej sygnał jest reprezentowany za pomocą współczynników  $c_3[k]$ ,  $d_3[k]$ ,  $d_2[k]$  i  $d_1[k]$ .

Funkcja skalująca oraz falka spełniają następujące zależności (*two scale equation*):

$$\phi[n] = \sum_k h[k] \sqrt{2} \phi[2n - k], \quad k \in C , \quad (11.13)$$

$$\psi[n] = \sum_k h_l[k] \sqrt{2} \phi[2n - k], \quad k \in C . \quad (11.14)$$

gdzie  $h[k]$  i  $h_l[k]$  oznaczają, odpowiednio, odpowiedź impulsową filtra LP i HP. Równania (11.13), (11.14) pozwalają obliczyć zarówno falkę jak i funkcję skalującą na kolejnym poziomie dekompozycji  $j+1$  na podstawie znajomości funkcji skalującej na poziomie bieżącym  $j$ , co przedstawiono na rys.11.7 dla funkcji skalującej Daubechies 1.



Rys.11.6 Analiza wielopoziomowa.

$$\begin{aligned} \frac{1}{0} \quad \frac{1}{1} &= \frac{1}{0} \frac{1/2}{1/2} + \frac{1/2}{1} \frac{1}{1} \\ \frac{1}{0} \quad \frac{-1}{1} &= \frac{1}{0} \frac{1/2}{1/2} - \frac{1/2}{1} \frac{1}{1} \end{aligned}$$

Rys.11.7 Ilustracja równań (11.13), (11.14) dla funkcji skalującej db1 (Haara).

W celu określenia zależności pomiędzy współczynnikami transformacji falkowej na sąsiednich poziomach dekompozycji podstawmy  $n = 2^{-j}t - k$  do wzoru (11.13):

$$\phi[2^{-j}t - k] = \sum_n h[n] \sqrt{2} \phi[2(2^{-j}t - k) - n] = \sum_n h[n] \sqrt{2} \phi[2^{-j+1}t - 2k - n], \quad (11.15)$$

po zamianie zmiennych  $m = 2k + n$  otrzymujemy:

$$\phi[2^{-j}t - k] = \sum_m h[m - 2k] \sqrt{2} \phi[2^{-j+1}t - m]. \quad (11.16)$$

Na poziomie  $j-1$  sygnał  $x[n]$  można wyrazić jako jego aproksymację tylko za pomocą funkcji skalujących, tj. bez falek:

$$x[n] = \sum_k c_{j-1}[k] 2^{-(j+1)/2} \phi[2^{-j-1}n - k]. \quad (11.17)$$

Równoważny zapis na poziomie  $j$  składa się z aproksymacji uzupełnionej detalami:

$$x[n] = \sum_k c_j[k] 2^{-j/2} \phi[2^{-j} n - k] + \sum_k d_j[k] 2^{-j/2} \psi[2^{-j} n - k]. \quad (11.18)$$

Dla sygnałów dyskretnych całka we wzorze definicyjnym (11.1) zostaje zastąpiona sumą i współczynniki detali  $d_j[k]$  oblicza się ze wzoru:

$$d_j[k] = \sum_n x[n] 2^{-j/2} \psi[2^{-j} n - k], \quad (11.19)$$

analogicznie współczynniki aproksymacji  $c_j[k]$  oblicza się z zależności:

$$c_j[k] = \sum_n x[n] 2^{-j/2} \phi[2^{-j} n - k]. \quad (11.20)$$

Podstawiając (11.16) do (11.20) i zmieniając zmienne  $t \rightarrow n$  otrzymujemy:

$$c_j[k] = \sum_n x[t] 2^{-j/2} \phi[2^{-j} n - k] = \sum_n x[t] 2^{-j/2} \sum_m h[m - 2k] \sqrt{2} \phi[2^{-j+1} n - m]. \quad (11.21)$$

Zmieniając kolejność sumowania w (11.21) mamy:

$$c_j[k] = \sum_m h[m - 2k] \sum_n x[t] 2^{-(j+1)/2} \phi[2^{-j+1} n - m]. \quad (11.22)$$

Druga suma w (11.22) jest sygnałem aproksymacji na poziomie  $j-1$  (11.17), ostatecznie otrzymujemy

$$c_j[k] = \sum_m h[m - 2k] c_{j-1}[m], \quad (11.23)$$

lub równoważnie:

$$c_{j+1}[k] = \sum_m h[m - 2k] c_j[m] \quad (11.23)$$

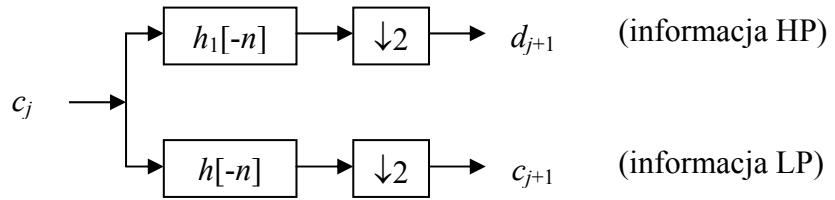
i analogicznie dla sygnału detali:

$$d_{j+1}[k] = \sum_m h_1[m - 2k] c_j[m]. \quad (11.24)$$

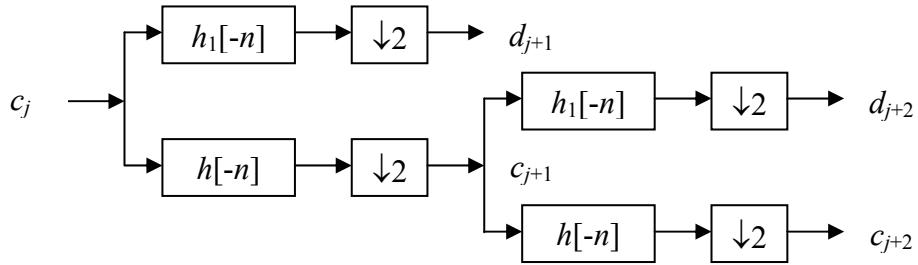
Wzory (11.23) i (11.24) stanowią podstawę teoretyczną obliczania współczynników DWT z wykorzystaniem decymowanego zespołu (banku) filtrów. Wzory te przypominają splot dwóch ciągów, z tą różnicą, że indeksy przy współczynnikach filtrów powodują konieczność odrzucenia co drugiej próbki oraz współczynniki filtrów są odwrócone w czasie

Rys.11.8a przedstawia schemat blokowy obliczeń DWT za pomocą decymowanego zespołu filtrów. Znak '-' występujący w  $h[-n]$  i  $h_1[-n]$  oznacza odwrócenie kolejności współczynników w stosunku do filtrów  $h[n]$  i  $h_1[n]$ . Po filtracji sygnał jest dwukrotnie decymowany, tak więc liczba współczynników falkowych  $d_j$  i  $c_j$  jest taka sama jak liczba próbek sygnału  $c_{j-1}$ . Rys.11.8b przedstawia analizę wielopoziomową, na kolejnych etapach dekompozycji filtrowany jest sygnał aproksymacji. Ze względu na operację decymacji, na każdym kolejnym poziomie analizy liczba współczynników aproksymacji i detali jest mniejsza o połowę.

a)

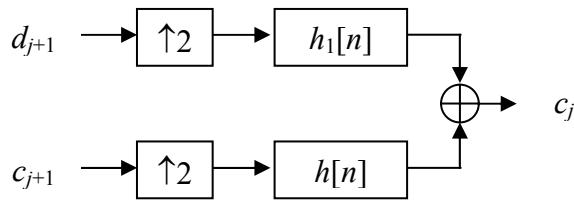


b)

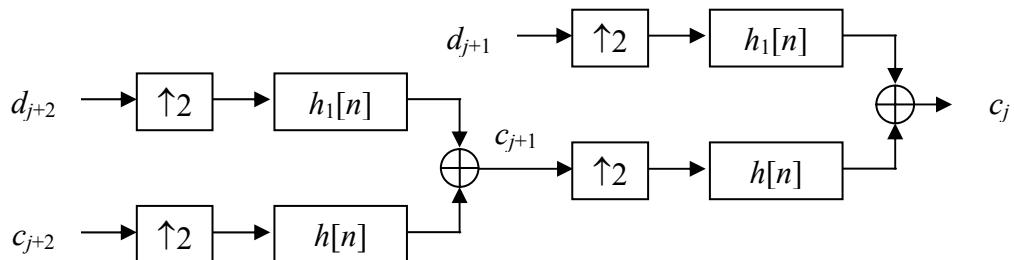


Rys.11.8 a) Wyznaczanie współczynników DWT przy pomocy zespołu (banku) filtrów; b) Analiza wielopoziomowa.

a)



b)



Rys.11.9 a) Odwrotna, dyskretna transformacja falkowa IDWT; b) Synteza wielopoziomowa.

Odwrotna, dyskretna transformacja falkowa oznaczana jako IDWT (*Inverse DWT*) polega na syntezie współczynników  $c_j$  na podstawie  $c_{j+1}$  i  $d_{j+1}$ :

$$c_j[k] = \sum_m c_{j+1}[m]h[k-2m] + \sum_m d_{j+1}[m]h_1[k-2m]. \quad (11.25)$$

Schemat blokowy IDWT przedstawia rys.11.9. Ponieważ przy wyznaczaniu współczynników DWT odrzucano co drugą próbkę, więc przy syntezie sygnał uzupełnia się zerami (co drugą próbke), aby otrzymać tę samą liczbę próbek sygnału co przed analizą.

Filtr  $h[n]$  musi spełniać następujące warunki:

1. liczba współczynników filtra musi być liczbą parzystą,

$$2. \sum_n h[n] = \sqrt{2}, \quad (11.26)$$

$$3. \sum_n h[n]h[n-2k] = \begin{cases} 1 & \text{dla } k=0 \\ 0 & \text{w pozostałych przypadkach} \end{cases} \quad (\text{ortogonalność}). \quad (11.27)$$

Współczynniki filtra  $h_1[n]$  oblicza się z zależności:

$$h_1[n] = (-1)^n h[N-n]. \quad (11.28)$$

Z warunków (11.26), (11.27) wynika, że dla filtra o długości  $N$ , przy czym  $N$  jest parzyste, istnieje  $N/2-1$  stopni swobody w wyborze  $h[n]$ . Wykorzystując stopnie swobody można projektować falki, które są optymalne w sensie przyjętego kryterium (np. falki Daubechies są najbardziej gładkie i regularne).

Jako przykład rozważmy filtr o długości  $N=2$ . W tym przypadku nie ma żadnych stopni swobody w doborze współczynników filtra  $h[n]$ . Z warunków (11.26) i (11.27) otrzymujemy:

$$\begin{cases} h[0] + h[1] = \sqrt{2} \\ h^2[0] + h^2[1] = 1 \end{cases}. \quad (11.29)$$

Jedynym rozwiązaniem układu równań (11.29) jest:

$$\{h[0], h[1]\} = \left\{ \frac{1}{\sqrt{2}}, \frac{1}{\sqrt{2}} \right\}. \quad (11.30)$$

Są to współczynniki filtra dla falki Daubechies pierwszego rzędu oznaczanej db1. Ze względu na to, że dla tak określonego filtra funkcje skalujące są funkcjami Haara, falkę tę nazywa się również falką Haara.

Jako kolejny przykład rozważmy filtr o długości  $N=4$ . W tym przypadku istnieje  $N/2-1=1$  stopień swobody w doborze współczynników filtra  $h[n]$ . Z warunków (11.26), (11.27) otrzymujemy:

$$\begin{cases} h[0] + h[1] + h[2] + h[3] = \sqrt{2} \\ h^2[0] + h^2[1] + h^2[2] + h^2[3] = 1 \\ h[0]h[2] + h[1]h[3] = 0 \end{cases}. \quad (11.31)$$

Współczynniki filtra można przedstawić w postaci:

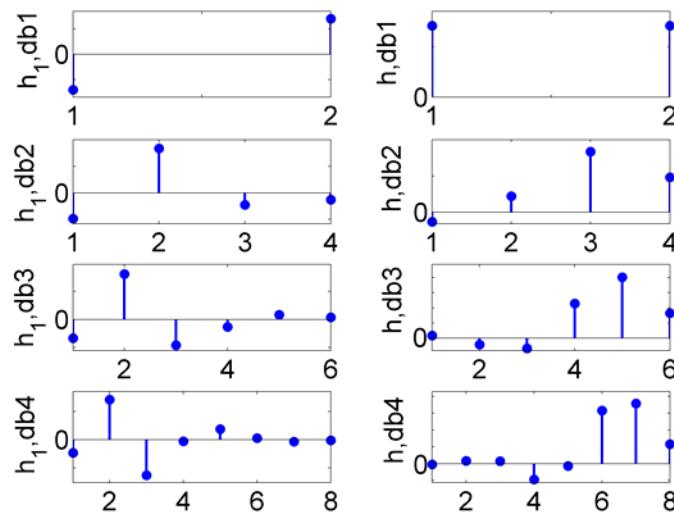
$$\begin{aligned} h[0] &= (1 - \cos(\alpha) + \sin(\alpha)) / (2\sqrt{2}) \\ h[1] &= (1 + \cos(\alpha) + \sin(\alpha)) / (2\sqrt{2}), \\ h[2] &= (1 + \cos(\alpha) - \sin(\alpha)) / (2\sqrt{2}) \\ h[3] &= (1 - \cos(\alpha) - \sin(\alpha)) / (2\sqrt{2}) \end{aligned} \quad (11.32)$$

gdzie  $\alpha$  jest parametrem. Dla  $\alpha=\pi/3$  otrzymuje się współczynniki filtra dla falki Daubechies drugiego rzędu oznaczanej db2.

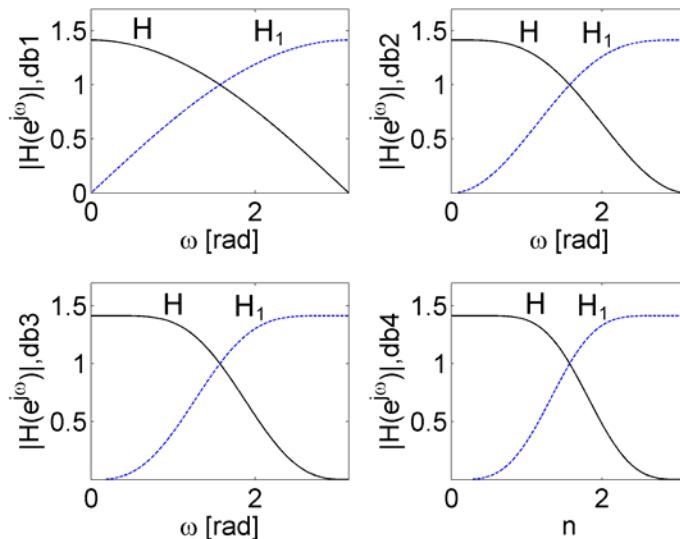
Rys.11.10 przedstawia współczynniki filtrów analizy  $h_1[-n]$  i  $h[-n]$  dla czterech pierwszych rzędów falki Daubechies. Wraz ze wzrostem rzędu falki rośnie długość filtra falkowego. Charakterystyki amplitudowe filtrów analizy przedstawione są na rys.11.11. Filtry analizy dzielą pasmo sygnału dyskretnego na część dolnoprzepustową i górnoprzepustową, wraz ze wzrostem rzędu falki filtry falkowe są coraz bardziej strome (selektywne). Falki i

funkcje skalujące skojarzone z filtrami z rys.11.10 są przedstawione na rys.11.12. Wartość średnia falek wynosi zero. Wraz ze wzrostem rzędu, falki stają się coraz gładzsze.

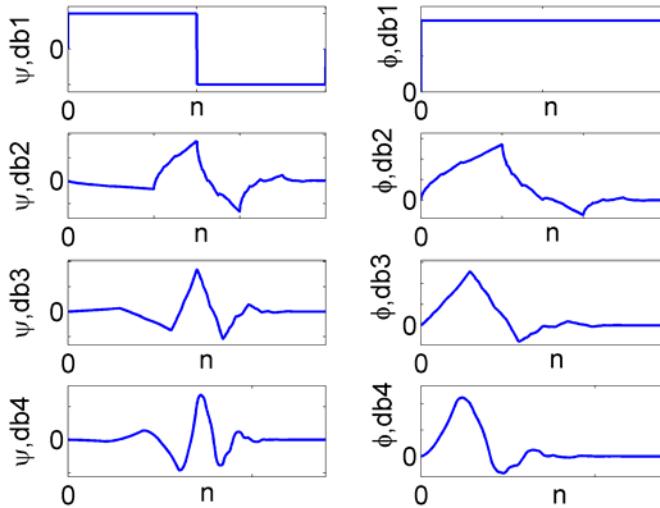
Większość falek, w tym również falki Daubechies, nie ma postaci analitycznej. Falki i funkcje skalujące są wyznaczane w sposób iteracyjny z wykorzystaniem analizy wielopoziomowej przedstawionej na rys.11.9b. Z definicji, współczynniki aproksymacji  $c$  i detali  $d$  określają podobieństwo (korelację) analizowanego sygnału do funkcji skalującej i falki. Dlatego funkcję skalującą otrzymamy syntezując sygnał w układzie z rys.11.9b, dla którego współczynniki aproksymacji  $c$  są impulsem jednostkowym, a współczynniki detali  $d$  są zerami. Analogicznie falkę otrzymamy syntezując sygnał, dla którego współczynniki aproksymacji  $c$  są zerami, a współczynniki detali  $d$  są impulsem jednostkowym. Rys.11.13 przedstawia iteracyjne wyznaczanie falki i funkcji skalującej db4 dla 1,2,3 i 4 iteracji. Jedna iteracja oznacza jeden poziom IDWT z rys.11.9b.



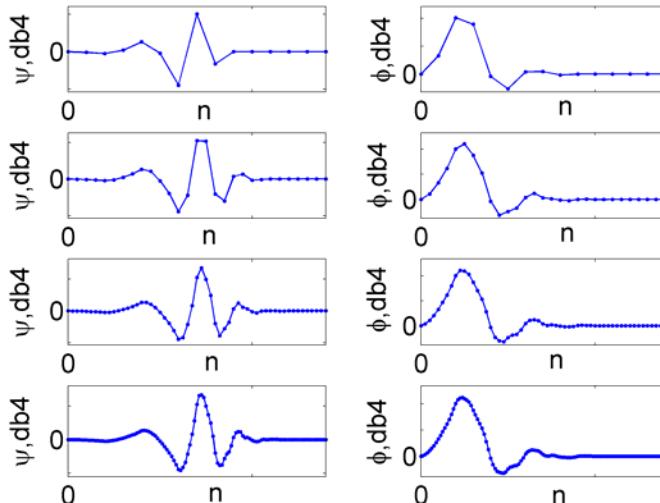
Rys.11.10 Współczynniki filtrów analizy  $h_1[-n]$  i  $h[-n]$  dla czterech pierwszych rzędów falki Daubechies.



Rys.11.11 Charakterystyki amplitudowe filtrów analizy  $h_1[-n]$  i  $h[-n]$  dla czterech pierwszych rzędów falki Daubechies.



Rys.11.12 Falki i funkcje skalujące typu Daubechies dla czterech pierwszych rzędów.

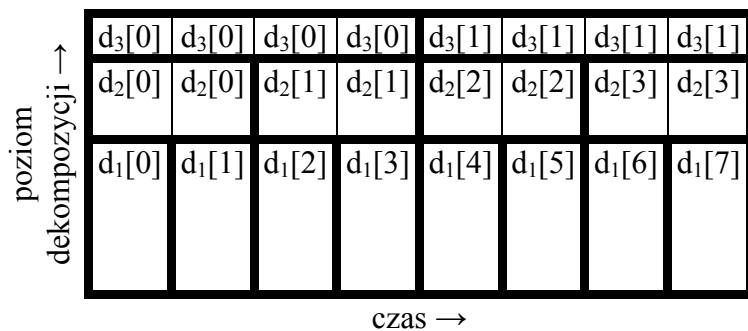


Rys.11.13 Iteracyjne wyznaczanie falki i funkcji skalującej db4 dla 1, 2, 3 i 4 iteracji.

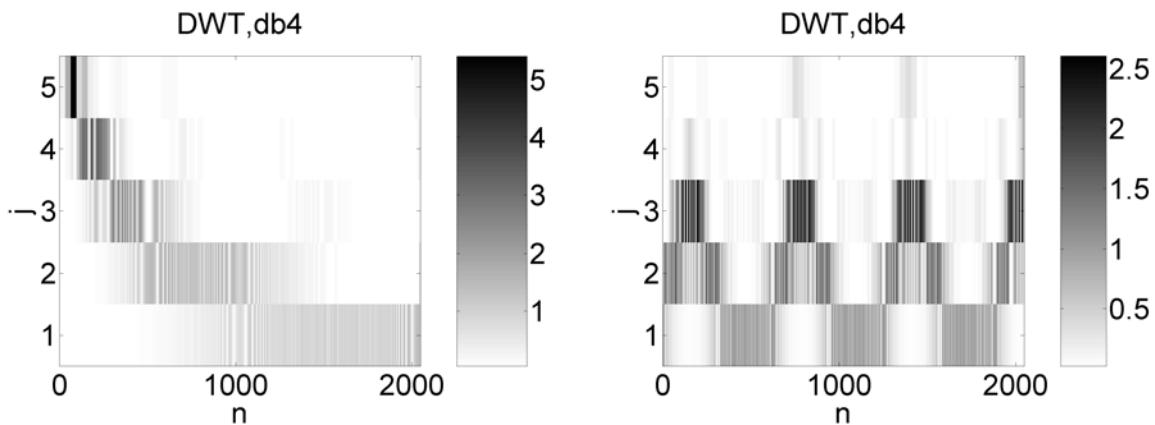
W wyniku analizy wielopoziomowej przedstawionej na rys.11.8b otrzymujemy zbiór współczynników detali, które tworzą płaszczyznę czas-skala podobnie jak w przypadku CWT, z tą różnicą, że skale są dobierane jako potęgi liczby 2 wg (11.9), a nie arbitralnie. Rys.11.14 przedstawia uporządkowanie współczynników falkowych DWT na płaszczyźnie. Poszczególne pola tworzą tzw. atomy czasowo-częstotliwościowe. Rozmiary tych atomów zmieniają się, jednak ich pole pozostaje takie samo. Dla porównania w przypadku STDFT wszystkie pola mają taki sam rozmiar.

Rys.11.15 przedstawia DWT sygnału sinusoidalnego z liniową i sinusoidalną modulacją częstotliwości. STDFD i CWT tych sygnałów pokazane są na rys.11.2-11.5. Oś  $OY$ , na rys.11.15 podaje poziom dekompozycji  $j$ . Dla DWT zależność pomiędzy poziomem dekompozycji  $j$  a skalą  $s$  wynosi  $s=2^j$  (11.9), tak więc poziomy dekompozycji  $j=\{1,2,3,4,5\}$  odpowiadają skalom  $s=\{2,4,8,16,32\}$ . Liczba poziomów dekompozycji jest ograniczona długością sygnału  $x[n]$ , ponieważ wektory współczynników DWT są krótsze o połowę na każdym kolejnym poziomie analizy. Dla sygnału o długości  $N=2048$  próbek sygnały aproksymacji i detali na 5 poziomie dekompozycji zawierają  $N/2^5=64$  próbki.

Falka db4 ma wartości rzeczywiste, dlatego płaszczyzna czas-skala na rys.11.15 przedstawia wyniki filtracji pasmowej, a nie amplitudę sygnału jak ma to miejsce w przypadku STDFT.



Rys.11.14 Prezentacja wyników DWT na płaszczyźnie czas-skala (czas-częstotliwość).



Rys.11.15 DWT sygnału sinusoidalnego z liniową i sinusoidalną modulacją częstotliwości.

Falki z rodziny Daubechies tworzą systemy ortogonalne. Uogólnieniem systemów ortogonalnych są systemy biortogonalne przedstawione na rys.11.16, dla których filtry analizy nie muszą być odwróconymi w czasie filtrami syntezy. Różnice pomiędzy tymi systemami są następujące:

1. W systemie ortogonalnym długość filtra  $h[n]$  musi być liczbą parzystą. W systemie biortogonalnym różnica długości filtrów  $h[n]$  i  $\tilde{h}[n]$  musi być liczbą parzystą.
2. System biortogonalny umożliwia projektowanie symetrycznych falek i funkcji skalujących. Jest to główna zaleta tego systemu.
3. System biortogonalny nie zachowuje energii sygnału. Jest to główna wada tego systemu.
4. W systemie biortogonalnym można zamienić filtry analizy z filtrami syntezy, co prowadzi do tzw. odwrotnych systemów biortogonalnych (*reverse biorthogonal*).
5. Szum biały pozostaje biały po transformacji ortogonalnej, natomiast dla transformacji biortogonalnych staje się szumem skorelowanym (kolorowym).

Dla układu biortogonalnego filtry  $h[n]$  i  $\tilde{h}[n]$  muszą spełniać następujące warunki:

1. różnica liczby współczynników filtrów musi być liczbą parzystą,

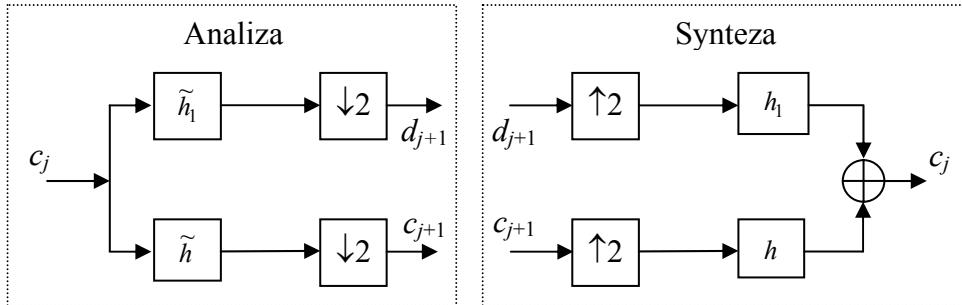
$$2. \sum_n h[n] = \sum_n \tilde{h}[n] = \sqrt{2}, \quad (11.33)$$

$$3. \sum_n \tilde{h}[n]h[n+2k] = \begin{cases} 1, & \text{dla } k = 0 \\ 0, & \text{w pozostałych przypadkach} \end{cases} \quad (\text{biortogonalność}). \quad (11.34)$$

Współczynniki filtrów  $h_l[n]$ ,  $\tilde{h}_l[n]$  oblicza się ze wzoru:

$$\begin{cases} \tilde{h}_l[n] = (-1)^n h[1-n] \\ h_l[n] = (-1)^n \tilde{h}[1-n] \end{cases}. \quad (11.35)$$

Z warunków (11.33) i (11.34) wynika, że przy określonych długościach filtrów istnieją stopnie swobody w doborze współczynników. Są one wykorzystywane np. do zapewniania symetrii funkcji bazowych w układach biortogonalnych.



Rys.11.16 Zespół filtrów biortogonalnych.

### 11.2.1 DWT w wersji predykcyjnej

Transformacja DWT w wersji predykcyjnej oznaczana jako LWT (*Lifting Wavelet Transform*) przedstawiona jest na rys.11.17. Bloczek *Podział* realizuje tzw. leniwy transformację falkową. Dzieli on sygnał oryginalny  $c_j$  (na  $j$ -tym poziomie dekompozycji) na dwie składowe polifazowe, jedną składającą się z próbek sygnału  $c_j$  o indeksach parzystych oznaczaną jako  $c_j^{(e)}$  oraz drugą zbudowaną z jego próbek o indeksach nieparzystych  $c_j^{(o)}$ . Zakładając, że sygnał  $c_j$  jest wektorem o parzystej liczbie elementów mamy:

$$c_j^{(e)}[n] = c_j[2n], \quad n = 0 \dots \frac{N}{2} - 1, \quad (11.36)$$

oraz

$$c_j^{(o)}[n] = c_j[2n + 1], \quad n = 0 \dots \frac{N}{2} - 1. \quad (11.37)$$

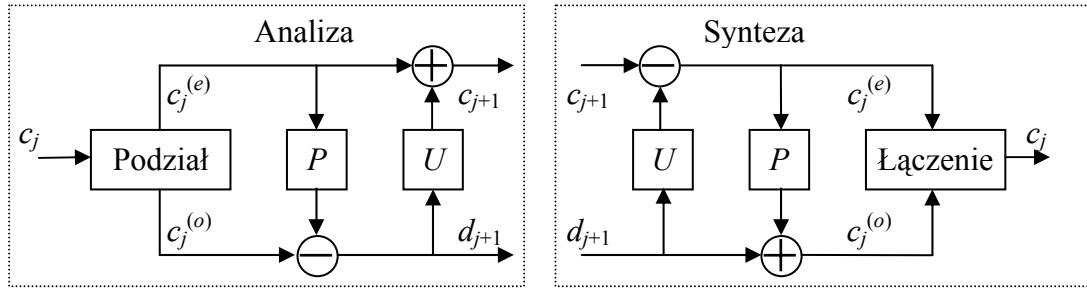
Wyjściowy sygnał detali  $d_{j+1}$  (na  $j+1$ -wszym poziomie dekompozycji) otrzymuje się jako błąd predykci sygnałów  $c_j^{(e)}$  na podstawie sygnałów  $c_j^{(o)}$ . Współczynniki  $d_{j+1}$  zawierają informację o wyższych częstotliwościach. Wyjściowy sygnał aproksymacji  $c_{j+1}$  otrzymuje się w wyniku działania operatora opisanego na schemacie bloczkiem *U* - uaktualnienie. Operator ten dobiera się tak, aby wartość średnia sygnałów  $c_j$  i  $c_{j+1}$  była taka sama. Stosując różne operatory  $P$  i  $U$  poprawia się własności leniwej transformacji falkowej metodą tzw. *liftingu*. W formie predykcyjnej można przedstawić wszystkie klasyczne układy falkowe (tj. zdefiniowane w postaci zespołu filtrów).

Podsumowując, wyznaczenie współczynników  $c_{j+1}$  i  $d_{j+1}$  przebiega następująco:

1. Podział:  $c_j \rightarrow c_j^{(e)}, c_j^{(o)}$
  2. Predykcja:  $d_{j+1} = c_j^{(o)} - P\{c_j^{(e)}\}$
  3. Uaktualnienie:  $c_{j+1} = c_j^{(e)} + U\{d_{j+1}\}$
- (11.38)

Odwrotną LWT wyznacza się przez odwrócenie kolejności działań wykonanych w transformacji prostej, co pokazano na rys. 11.17:

1. Odwrócenie uaktualnienia:  $c_j^{(e)} = c_{j+1} - U\{d_j\}$
2. Odwrócenie predykcji:  $c_j^{(o)} = d_{j+1} + P\{c_j^{(e)}\}$
3. Łączenie:  $c_j^{(e)}, c_j^{(o)} \rightarrow c_j$



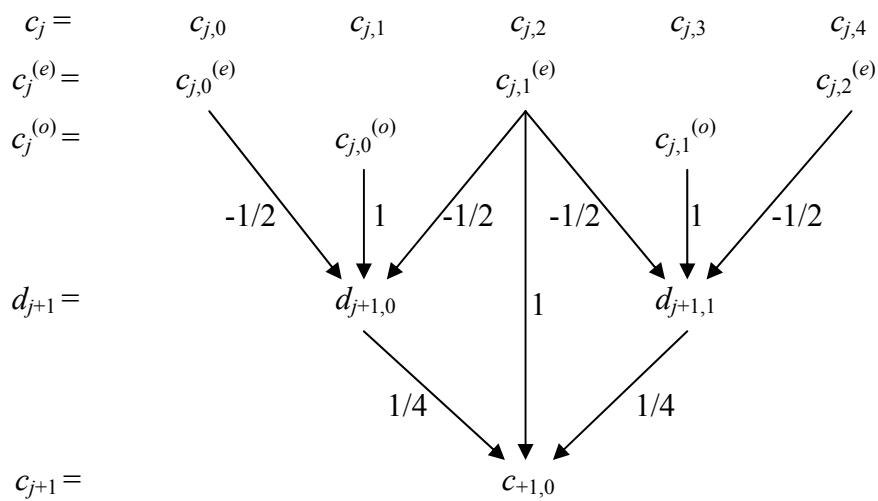
Rys.11.17 Schematy blokowe prostej i odwrotnej LWT,  $P$  – predykcja,  $U$  – uaktualnienie.

Zdefiniowana równaniami (11.38) i (11.39) LWT jest zawsze odwracalna niezależnie od tego, jakie filtry (funkcje)  $P$  i  $U$  zostaną użyte, w szczególności mogą to być zależności nieliniowe. Implementacja DWT w wersji LWT umożliwia obliczenia z podstawianiem, tzn. nie potrzeba dodatkowej pamięci na przechowywanie pośrednich wyników obliczeń. Obliczenia LWT mają niższą złożoność obliczeniową niż DWT liczona w decymowanym zespole filtrów z rys.11.16.

Tab.11.1 podaje przykładowe współczynniki filtra predykcji  $P$ . Współczynniki uaktualnienia  $U$  wyznacza się jako  $P/2$ , np. dla  $P=\{1/2, 1/2\}$ ,  $U=\{1/4, 1/4\}$ . W układzie LWT z rys.11.17 długości filtrów  $P$  i  $U$  nie muszą być takie same.

Tab. 11.1 Współczynniki filtra predykcji  $P$ .

{1}
{1/2,1/2}
{-1/2^4,9/2^4,9/2^4,-1/2^4}
{3/2^8,-25/2^8,150/2^8,150/2^8,-25/2^8,3/2^8}



Rys.11.18 Rozpływ sygnału w LWT z rys.11.17 dla  $P = \{1/2, 1/2\}$  i  $U = \{1/4, 1/4\}$ .

W celu pokazania równoważności obliczeń pomiędzy układem LWT z rys.11.17, a decymowanym zespołem filtrów z rys.11.16 rozważmy układ LWT z  $P=\{1/2, 1/2\}$  i  $U=\{1/4, 1/4\}$ . Rys.11.18 przedstawia rozpływ sygnału w LWT dla  $P=\{1/2, 1/2\}$  i  $U=\{1/4, 1/4\}$ . Sygnał detali  $d_{j+1}$  wyznaczany jest w wyniku przesuwania po sygnale  $c_j$  filtru o wagach  $\tilde{h}_1 = \{-1/2, 1, -1/2\}$  co drugą próbki. Jest to równoważne z filtracją sygnału  $c_j$  filtrem  $\tilde{h}_1$ , po którym następuje decymacja, tak jak na rys.11.16. Analogicznie sygnał aproksymacji  $c_{j+1}$  wyznacza się w wyniku przesuwania po sygnale  $c_j$  filtru o wagach  $\tilde{h} = \{-1/8, 1/4, 1-1/8-1/8, 1/4, -1/8\} = \{-1/8, 1/4, 6/8, 1/4, -1/8\}$ , co drugą próbki. Jest to równoznaczne z filtracją sygnału  $c_j$  filtrem  $\tilde{h}$ , po którym następuje decymacja, tak jak na rys.11.16. Filtry  $\tilde{h} = \{-1/8, 1/4, 6/8, 1/4, -1/8\}$  i  $\tilde{h}_1 = \{-1/2, 1, -1/2\}$  stanowią układ biortogonalny oznaczany jako 5/3 (gdzie liczby 5 i 3 są długościami filtrów).

### 11.2.1.1 Problem brzegowy

W cyfrowym przetwarzaniu sygnałów zawsze mamy do czynienia z sygnałami o skończonym czasie trwania (tzn. sygnał cyfrowy składa się ze skończonej liczby próbek). Sytuacja taka powoduje powstawanie problemu brzegowego, który objawia się tym, że w wyniku filtracji cyfrowej sygnału zawierającego  $N$  próbek filtrem o długości  $M$  otrzymuje się  $N+M-1$  próbek (a nie  $N$ ). W przypadku DWT problem brzegowy można rozwiązać przez zastosowanie periodyzacji sygnału lub odbicia lustrzanego na brzegach sygnału. Dla LWT dodatkowo istnieje możliwość przełączania długości filtrów  $P$  i  $U$  na brzegach sygnału.

Dla predyktora o długości 1, tj.  $P=\{1\}$  problem brzegowy nie występuje. Można, więc rozpocząć analizę sygnału na lewym brzegu filtrem o długości 1, a następnie zwiększać długość filtra w miarę jak dostępne stają się lewe, skrajne próbki. Analogicznie, w miarę jak zaczyna brakować próbek na prawym brzegu można przełączać się na krótsze filtry.

### 11.2.1.2 Predykcyjna transformacja falkowa w wersji całkowitoliczbowej

Każda LWT może zostać zaimplementowana w postaci całkowitoliczbowej, przez wprowadzenie operatorów zaokrąglenia, w sposób pokazany na rys.11.19:

Analiza:

- |   |  |
|---|--|
| 1. Podział:<br>2. Predykcja:<br>3. Uaktualnienie: | $c_j \rightarrow c_j^{(e)}, c_j^{(o)}$<br>$d_{j+1} = c_j^{(o)} - \lfloor P\{ c_j^{(e)} \} \rfloor$<br>$c_{j+1} = c_j^{(e)} + \lfloor U\{ d_{j+1} \} \rfloor$ |
|---|--|
- (11.40)

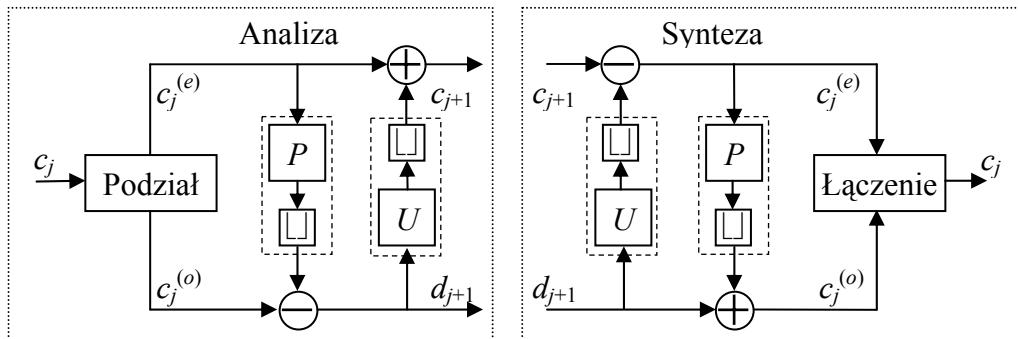
Synteza:

- |  |  |
|--|--|
| 1. Odwrócenie uaktualnienia:<br>2. Odwrócenie predykcji:<br>3. Łączenie: | $c_j^{(e)} = c_{j+1} - \lfloor U\{ d_j \} \rfloor$<br>$c_j^{(o)} = d_{j+1} + \lfloor P\{ c_j^{(e)} \} \rfloor$<br>$c_j^{(e)}, c_j^{(o)} \rightarrow c_j$ |
|--|--|
- (11.41)

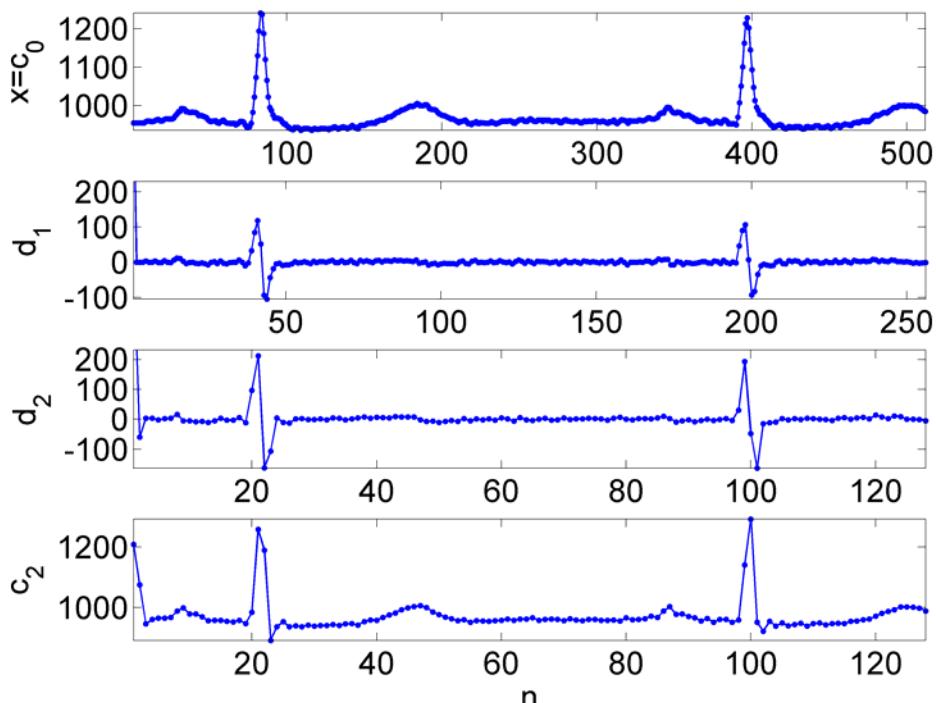
Symbol „ $\lfloor \rfloor$ ” w (11.40) i (11.41) oznacza zaokrąglenie do najbliższej liczby całkowitej. Obliczenia (11.40) i (11.41) odbywają się na liczbach zmiennoprzecinkowych. Implementację całkowitoliczbową LWT należy rozumieć w ten sposób, że dla sygnału wejściowego w postaci liczb całkowitych współczynniki falkowe są również liczbami całkowitymi. Dzięki temu błąd rekonstrukcji, tzn. różnica pomiędzy sygnałem i jego rekonstrukcją ze współczynników falkowych, wynosi zero. W przypadku transformat takich jak np. DFT czy DWT, błąd rekonstrukcji nie jest równy zeru, a jego wartość zależy od precyzji obliczeń.

Wersja całkowitoliczbowa LWT odgrywa ważną rolę w algorytmach kompresji bezstratnej.

Rys.11.20 przedstawia dwupoziomową dekompozycję falkową sygnału EKG. Analizowany sygnał  $x[n]$  może być interpretowany jako sygnał aproksymacji na zerowym poziomie dekompozycji, tj.  $x[n]=c_0[n]$ . Po dwóch etapach dekompozycji sygnał  $x[n]$  jest reprezentowany przez współczynniki aproksymacji  $c_2[k]$  i współczynniki detali  $d_2[k]$  i  $d_1[k]$ . Sygnał  $x[n]$  zawiera dużą składową stałą, która jest obecna również w sygnale aproksymacji  $c_2[k]$ . Sygnały detali nie zawierają składowej stałej.



Rys.11.19 Schematy blokowe prostej i odwrotnej LWT w wersji całkowitoliczbowej,  $P$  – predykcja,  $U$  – uaktualnienie, „└” – zaokrąglenie do najbliższej liczby całkowitej.



Rys.11.20 Dwupoziomowa dekompozycja falkowa sygnału EKG,  $P=\{1/2, 1/2\}$  i  $U=\{1/4, 1/4\}$ .



## 12. Kodowanie entropijne

Entropia, jako miara ilości informacji źródła bez pamięci, została zdefiniowana przez Shannona w następujący sposób:

$$H(p_1, \dots, p_N) = H(S) = -\sum_{n=1}^N p_n \log_2 p_n . \quad (12.1)$$

gdzie  $S=\{x_1, \dots, x_N\}$  jest zbiorem  $N$  niezależnych zdarzeń, a  $P=\{p_1, \dots, p_N\}$  zbiorem prawdopodobieństw ich zajścia (rozkładem prawdopodobieństwa). Dla kodowania komunikatów entropia jest ograniczeniem dolnym na średnią liczbę bitów przypadających na jeden symbol wejściowy, tj. ograniczeniem na średnią długość słowa kodu używanego przy kodowaniu tych symboli.

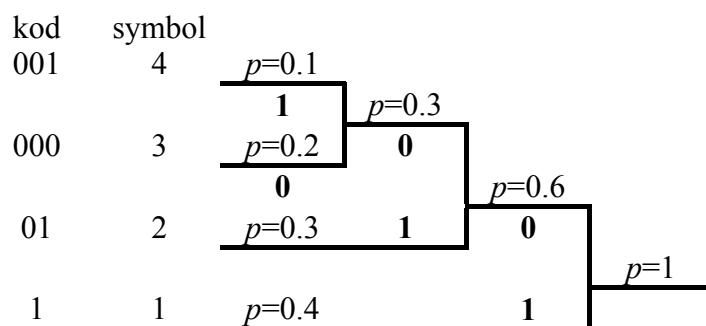
Jako przykład wyznaczmy entropię komunikatu  $x=[1,3,1,2,1,2,1,3,4,2]$ . Sygnał  $x$  zawiera cztery symbole  $S=\{1,2,3,4\}$ , a prawdopodobieństwa ich zajścia wynoszą odpowiednio  $P=\{0.4,0.3,0.2,0.1\}$ . Entropia sygnału  $x$  wynosi  $H=1.8464$ . Dla źródła bez pamięci komunikat  $x$  można przesłać na  $10H=18.464$  bitach. Można zauważyć, że gdyby symbole były jednakowo prawdopodobne to entropia wyniosłaby  $H=2$  i do przesłania komunikatu potrzebne byłoby 20 bitów.

Kodery entropijne są to kodery, dla których średnia długość słowa kodowego jest bardzo bliska długości optymalnej, wyznaczonej na podstawie entropii źródła (12.1). Przykładami koderów entropijnych są: koder Huffmmana, koder arytmetyczny oraz koder słownikowy LZW.

### 12.1 Kodowanie Huffmmana

Rozważmy komunikat  $x=[1,3,1,2,1,2,1,3,4,2]$ , dla którego  $S=\{1,2,3,4\}$ ,  $P=\{0.4,0.3,0.2,0.1\}$ . Pierwszym etapem kodowania Huffmmana jest określenie tabeli kodowej. Tabela ma kształt drzewa binarnego i jest budowana w następujący sposób:

1. symbole i ich prawdopodobieństwa są umieszczane w węzłach (liściach),
2. spośród wszystkich symboli  $S$  wybierane są dwa o najniższym prawdopodobieństwie, przypisywane są im wartości 0 i 1, ich prawdopodobieństwo jest sumowane i tworzą one nowy węzeł,
3. krok 2 jest powtarzany, aż pozostałą tylko dwa węzły,
4. w wyniku powyższej procedury powstaje drzewo binarne, w którego liściach znajdują się symbole; kod dla każdego symbolu jest odczytywany przez poruszanie się po drzewie od korzenia do danego liścia. Powyższy algorytm dla sygnału  $x$  przedstawiony jest na rys.12.1.



Rys.12.1 Tabela kodowa dla  $S=\{1,2,3,4\}$ ,  $P=\{0.4,0.3,0.2,0.1\}$ .

Na rys.12.1 przyjęto konwencję, że kod 1 otrzymuje węzeł o niższym prawdopodobieństwie. Kompresja sygnału polega na tym, że symbole, które występują częściej otrzymują krótszy kod. Kod na rys.12.1 jest kodem przedrostkowym, tzn., że kody krótsze nigdy nie są początkiem kodów dłuższych, np. kod symbolu 1 wynosi 1 i nie jest to początek kodu dla żadnego z pozostałych symboli.

Kodowanie komunikatu odbywa się przez zastąpienie każdego z symboli odpowiadającym mu ciągiem bitów. Dla  $x=[1,3,1,2,1,2,1,3,4,2]$  i tabeli kodowej z rys.12.1 otrzymujemy następujący strumień 19 bitów [1,0,0,0,1,0,1,1,0,1,1,0,0,0,0,1,0,1], podczas gdy długość strumienia bitów wyliczona z entropii wynosi  $H=18.464$ .

Koder Huffmmana przeznacza całkowite wielokrotności bitu na jeden symbol. Powoduje to, że średnia długość słowa kodowego może być znacznie większa (nawet o kilkanaście procent) od optymalnej. W przypadku, gdy w komunikacie występuje dużo symboli o małym prawdopodobieństwie, którym przyporządkowane byłyby bardzo długie kody, można usprawnić kodowanie przez wprowadzenie tzw. kodów ucieczki lub grupowanie symboli. Po symbolu ucieczki wysyłany jest niezakodowany symbol o małym prawdopodobieństwie wystąpienia w komunikacie. Grupowanie polega na łączeniu pojedynczych symboli w słowa i kodowanie słów. Wymaga to określenia prawdopodobieństw występowania słów. Operacja grupowania znacząco zwiększa zbiór  $S$ , co prowadzi do spowolnienia kodowania.

Głównym parametrem oceny wydajności kompresji jest współczynnik kompresji  $c_r$  (*compression ratio*) definiowany jako:

$$c_r = N_o / N_c , \quad (12.2)$$

gdzie  $N_o$  - liczba bitów zajmowana przez sygnał oryginalny,  $N_c$  - długość strumienia bitów po kompresji. W rozważanym przykładzie, gdyby symbole sygnału zajmowały po 8 bitów (np. zmienna typu *char* lub *int*) współczynnik kompresji wyniósłby  $c_r=(10\cdot8)/19=4.21$ .

W celu zdekodowania strumienia bitów niezbędna jest znajomość tabeli kodowej. Jest ona przesyłana wraz z zakodowaną informacją, co dla krótkich komunikatów może w znacznym stopniu obniżyć współczynnik kompresji, lub może być uzgodniona *a priori* przez koder i dekoder dla pewnej klasy sygnałów, powoduje to jednak utratę optymalności kodowania.

Algorytm Huffmmana może być implementowany w wersji dynamicznej. Dynamiczny koder Huffmmana buduje tabelę kodową w trakcie kodowania i nie wymaga znajomości rozkładu prawdopodobieństwa symboli przed rozpoczęciem kodowania. Kodowanie może więc przebiegać w miarę jak napływają kolejne próbki (symbole) np. podczas rejestracji sygnału EKG w aparacie Holtera. Tabela kodowa nie jest przesyłana, ponieważ dekoder odbudowuje ją w trakcie dekodowania.

Dzięki temu, że kod jest przedrostkowy, dekodowanie strumienia bitów polega na wyszukiwaniu kodu symboli w kolejnych fragmentach strumienia.

Poniższy program, napisany w Matlabie, buduje tabelę kodową oraz przeprowadza kodowanie i dekodowanie komunikatu metodą Huffmmana. Komunikat testowy jest generowany w linii 3. Następnie wyznaczany jest rozkład prawdopodobieństwa symboli w komunikacie i generowana jest tabela kodowa. W linii 8 sprawdzany jest błąd toru koder-dekoder. Koder Huffmmana jest bardzo prosty w implementacji, linie 31-36, co jest jego zaletą. Dekodowanie strumienia bitów, linie 37-52, również nie jest skomplikowane.

```

1 function start
2 clc
3 x = abs(round(10*randn(1,100)));
4 [symbole, p]=zlicz(x);

```

```

5 [symbole, tabela] = tabela_huffmanna(symbole, p);
6 bity = huffman_koder(x, symbole, tabela);
7 y = huffman_dekoder(bity, symbole, tabela);
8 blad_rekonstrukcji=sum(abs(x(:)-y(:)))
9 function [symbol tabela] = tabela_huffmanna(symbol, prw);
10 for k=1:length(symbol);
11     st{k}=k; %symbole tymczasowe
12     tabela{k}=[];
13 end
14 p=prw(:);
15 Np = length(p)+1;
16 while Np>2
17     Np = length(find(p<10));
18     [p, I] = sort(p, 'ascend');
19     st = st(I);
20     for k=1:length(st{1})
21         tabela{st{1}(k)}=[1 tabela{st{1}(k)}];
22     end
23     for k=1:length(st{2})
24         tabela{st{2}(k)}=[0 tabela{st{2}(k)}];
25     end
26     st{1}=[st{1} st{2}];
27     st{2}=[];
28     p(1) = p(1)+p(2);
29     p(2) = 10;
30 end
31 function bity = huffman_koder(x, symbole, tabela);
32 bity=logical([]);
33 for k=1:length(x);
34     I = find(symbole==x(k));
35     bity = [bity logical(tabela{I})];
36 end
37 function y=huffman_dekoder(bity, symbole, tabela);
38 Nb=length(bity);
39 I1=1;
40 y = [];
41 while I1<=Nb
42     for k=1:size(tabela,2)
43         tab = tabela{k}; It=length(tab);
44         if I1+It-1<=Nb
45             if tab == bity(I1:I1+It-1);
46                 y=[y symbole(k)];
47                 I1=I1+It;
48                 break
49             end
50         end
51     end
52 end
53 function [symbol, prw]=zlicz(x);
54 % x - komunikat - liczby całkowite
55 symbol=[];
56 prw=[];
57 x = x(:);
58 N = length(x);
59 k = 1;
60 Iz = zeros(size(x));
61 while sum(Iz)<N;
62     Is=find(x==x(k));
63     symbol = [symbol x(k)];
64     prw = [prw length(Is)/N];
65     Iz(Is)=1;
66     Ik=find(Iz==0);
67     if length(Ik)>0
68         k=Ik(1);
69     else
70         break
71     end
72 end

```

## 12.2 Kodowanie arytmetyczne

Wyjściem kodera arytmetycznego jest liczba z przedziału  $[0,1) = \{y : 0 \leq y < 1\}$ . Kodowanie kolejnych symboli powoduje sukcesywne zwiększanie precyzji tej liczby (tj. zwiększanie liczby bitów niezbędnych do jej reprezentacji). Dzięki takiemu rozwiązaniu na jeden kodowany symbol może przypadać niecałkowita krotność bitu. Koderem arytmetycznym można więc skutecznie zakodować komunikat zawierający tylko dwa symbole, np. 0 i 1, co jest niemożliwe w przypadku kodera Huffmana, ponieważ każdy z symboli otrzymałyby kod o długości jednego znaku.

Niech  $S=\{x_1,\dots,x_n\}$  będzie alfabetem symboli do zakodowania, a  $P=\{p_1,\dots,p_n\}$  zbiorem prawdopodobieństw ich zajścia. Liczbę reprezentującą zakodowany komunikat określa się na podstawie prawdopodobieństw ze zbioru  $P$ .

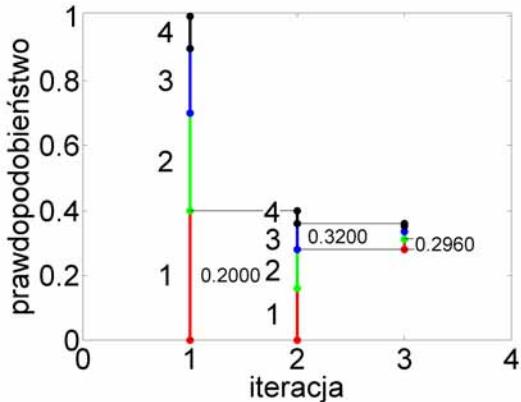
Rozważmy komunikat  $x=[1,3,1,2,1,2,1,3,4,2]$ , dla którego  $S=\{1,2,3,4\}$  i  $P=\{0.4,0.3,0.2,0.1\}$ . Proces kodowania przedstawia rys.12.2. W pierwszej iteracji przedział prawdopodobieństw wynosi  $[0,1)$ . Pierwszym symbolem komunikatu jest 1, którego prawdopodobieństwo wystąpienia wynosi 0.4 i symbol ten jest reprezentowany przez przedział o długości 0.4, w przykładzie  $[0,0.4)$ . Przedział ten nazwijmy przedziałem bieżącym. Wynikiem kodowania jest liczba z przedziału bieżącego, np. 0.2. W drugim kroku przedziały prawdopodobieństw wystąpienia poszczególnych symboli są przeskalowane tak, aby mieściły się one w zakresie przedziału bieżącego, co pokazano na rys.12.2. Kolejny w komunikacie jest symbol 3 należący do nowego przedziału bieżącego  $[0.28, 0.36)$ . Zakodowane do tej pory symbole tj. 1 i 3 są reprezentowane przez liczbę z aktualnego przedziału bieżącego np. 0.32. W trzecim kroku wykonywane jest kolejne przeskalowanie przedziałów prawdopodobieństw, a wynikiem kodowania jest liczba z nowego przedziału bieżącego, itd. W miarę kodowania liczba znaczących miejsc po przecinku w zakodowanej wartości jest coraz większa. W przykładzie dla sygnału  $x$ , tj. po 10 iteracjach otrzymano liczbę 0.29474639892578125, która w reprezentacji binarnej zajmuje 17 bitów  $[0,1,0,0,1,0,1,1,0,1,1,0,1,0,0,1]$ . Liczba ta określa kumulatywne prawdopodobieństwo wystąpienia komunikatu  $x$ .

Dekodowanie sygnału polega na wybieraniu symboli przypisanych danym przedziałom prawdopodobieństwa i takim skalowaniu przedziałów prawdopodobieństw, jak w procesie kodowania. W pierwszym kroku dekodowania liczba 0.2947... należy do przedziału prawdopodobieństwa symbolu 1. W drugim kroku liczba ta należy do przedziału prawdopodobieństwa symbolu 3, który wynosi  $[0.28, 0.36)$ , itd.

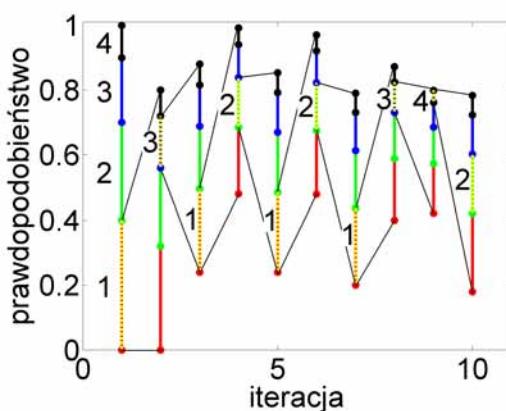
Powyższy algorytm ilustruje ideę kodowania arytmetycznego, nie ma on jednak zastosowania w praktyce, ponieważ nawet dla krótkich komunikatów precyzja reprezentacji liczb zmiennoprzecinkowych jest zbyt mała. Rozwiązaniem tego problemu jest skalowanie przedziału bieżącego, tak, aby jego długość pozostawała bliska 1.

Rys.12.3 przedstawia kodowanie sygnału  $x$  kodera arytmetycznym ze skalowaniem przedziału prawdopodobieństw. W kolejnych krokach przedział bieżący jest najpierw zwiększany a dopiero później skalowane są przedziały prawdopodobieństw odpowiadające poszczególnym symbolom.

Poniżej przedstawiono, napisaną w Matlabie, implementację kodera i dekodera arytmetycznego ze skalowaniem przedziału. Komunikat testowy generowany jest w linii 3. W linii 5 wyznaczany jest rozkład prawdopodobieństwa symboli w komunikacie. Argumentami funkcji `arithmetic_koder` są: komunikat, zbiór symboli i zbiór ich prawdopodobieństw. Przy dekodowaniu podawana jest dodatkowo długość komunikatu. W linii 8 wyznaczany jest błąd rekonstrukcji, który powinien wynosić zero.



Rys.12.2 Ilustracja zasady kodowania arytmetycznego, komunikat  $x=[1, 3, 1, \dots]$ ,  $S=\{1, 2, 3, 4\}$ ,  $P=\{0.4, 0.3, 0.2, 0.1\}$ .



Rys.12.3 Ilustracja zasady kodowania arytmetycznego z przeskalowaniem przedziału, komunikat  $x=[1,3,1,2,1,2,1,3,4,2]$ ,  $S=\{1, 2, 3, 4\}$ ,  $P=\{0.4, 0.3, 0.2, 0.1\}$ .

```

1 function start
2 clc
3 x = abs(round(10*randn(1,100))),
4 N = length(x);
5 [symbole, prw]=zlicz(x);
6 bity = arithmetic_koder(x, symbole, prw);
7 y = arithmetic_dekoder(bity, symbole, prw, N);
8 blad_rekonstrukcji=sum(abs(x(:)-y(:))),
```

Kodowanie rozpoczyna się z przedziałem bieżącym  $[0,1]$ . Następnie przedział ten jest dzielony na przedziały o długościach odpowiadających prawdopodobieństwom występowania symboli. Nowy przedział bieżący jest przedziałem, do którego należy kodowany symbol i jest on wyszukiwany w linii 16. W linii 19 wykonywana jest funkcja kluczowa dla całego algorytmu, która skaluje bieżący przedział.

Kodowanie kończy się dodaniem bitów przez funkcję koniec\_kodowania.

```

9 function bity = arithmetic_koder(x, symbole, prw);
10 % kodowanie arytmetyczne wersja ze skalowaniem
11 przedzialBiezacy=[0 1];
12 przedzialy = zeros(length(prw),2);
13 licznikBitow=0;
14 bity=logical([]);
15 for k=1:length(x);
16     I=find(x(k)==symbole);
17     przedzialy = podziel_przedzial(przedzialy, przedzialBiezacy, prw);
18     przedzialBiezacy = przedzialy(I,:);
```

```

19 [bit,przedzialBiezacy,licznikBitow]=zwroc_bity(przedzialBiezacy,licznikBitow);
20      bity=[bity logical(bit)];
21 end
22 [bit]=koniec_kodowania(przedzialBiezacy,licznikBitow);
23 bity=[bity logical(bit)];
24 function [przedzialy]=podziel_przedzial(przedzialy,przedzialBiezacy,prw);
25 k1=przedzialBiezacy(1);
26 for k=1:length(prw)
27     przedzialy(k,1)=k1;
28     k1 = k1+(przedzialBiezacy(2)-przedzialBiezacy(1))*prw(k);
29     przedzialy(k,2)=k1;
30 end

```

Jeżeli bieżący przedział zawiera się w przedziale  $[0,0.5]$ , to wysyłane jest 0. Przedział  $[0,0.5)$  w zapisie binarny to  $[0,0.1)_2$ , tak więc obie granice przedziału, w rozpatrywanym przypadku, mają 0 po przecinku w zapisie binarnym i zero to jest dodawane do kodu. Następnie granice przedziału są mnożone przez 2 w linii 47, czyli w reprezentacji binarnej granice bieżącego przedziału są przesuwane o jedną pozycję w lewo.

Jeżeli bieżący przedział zawiera się w przedziale  $[0.5,1)$ , to jego granice w zapisie binarnym mają 1 po przecinku i jedynka ta jest dodawana do kodu, następnie granice przedziału są obniżane o 0.5, w linii 40, i mnożone przez 2 w linii 47.

W przypadku, gdy bieżący przedział zawiera się w przedziale  $[0.25,0.75)$  jego granice są zmniejszane o 0.25 i następuje zapamiętanie tej operacji w zmiennej licznikBitow. Proces skalowania bieżącego przedziału jest zilustrowany na rys.12.3.

```

31 function [bit, przedzialBiezacy,licznikBitow]=zwroc_bity(przedzialBiezacy,
licznikBitow);
32 bit=[];
33 while 1
34     if przedzialBiezacy(1)>=0 && przedzialBiezacy(2)<0.5
35         bit=[bit 0 ones(1,licznikBitow)];
36         licznikBitow=0;
37     elseif przedzialBiezacy(1)>=0.5 && przedzialBiezacy(2)<1
38         bit=[bit 1 zeros(1,licznikBitow)];
39         licznikBitow=0;
40         przedzialBiezacy=przedzialBiezacy-0.5;
41     elseif przedzialBiezacy(1)>=0.25 && przedzialBiezacy(2)<0.75
42         licznikBitow=licznikBitow+1;
43         przedzialBiezacy=przedzialBiezacy-0.25;
44     else
45         break
46     end
47     przedzialBiezacy=2*przedzialBiezacy;
48 end
49 function [bit]=koniec_kodowania(przedzialBiezacy,licznikBitow)
50 licznikBitow=licznikBitow+1;
51 if przedzialBiezacy(1)<0.25
52     bit=[0 ones(1,licznikBitow)];
53 else
54     bit=[1 zeros(1,licznikBitow)];
55 end

```

Dekodowanie strumienia bitów wykorzystuje fakt, że początkowy fragment tego strumienia jest liczbą wskazującą na przedział prawdopodobieństwa odpowiadający zakodowanemu symbolowi. Początkowy fragment strumienia bitów jest zamieniany na liczbę dziesiętną w linii 70, a następnie liczba ta określa zakodowany symbol. Liczba bitów uwzględnianych przy zamianie na liczbę dziesiętną nie może być zbyt mała i powinna być większa od  $\lceil \log_2(1/(0.25 \min\{P\})) \rceil$ . W przykładzie dobrano rozmiar bufora bitów równy 15, czyli dla najmniejszego prawdopodobieństwa wystąpienia symbolu rzędu  $p=0.0001$ .

```

56 function y = arithmetic_dekoder(bity, symbole, prw, N);
57 %N- długość sygnału zakodowanego y
58 Nb=16; %długość bufora bitów !!!
59 BuforBitow=zeros(1,Nb);
60 k=1;
61 while k<=length(bity) && k<=Nb
62     BuforBitow(k)=bity(k);
63     k=k+1;
64 end
65 y = [];
66 przedzialBiezacy=[0 1];
67 przedzialy = zeros(length(prw),2);
68 for n=1:N
69     przedzialy = podziel_przedzial(przedzialy,przedzialBiezacy,prw);
70     liczba = buf2decy(BuforBitow);
71     I = find(liczba>=przedzialy(:,1) & liczba<przedzialy(:,2));
72     przedzialBiezacy=przedzialy(I,:);
73     y=[y symbole(I)];

```

W trakcie dekodowania bieżący przedział jest skalowany w pętli `while` (linie 74-93). Równocześnie zmieniana jest wartość bufora bitów przez binarne odjęcie 0.5 w linii 78 i 0.25 w linii 81. Kolejne bity kodu są przesuwane przez bufor bitów w liniach 87-92.

```

74     while (1)
75         if przedzialBiezacy(1)>=0 && przedzialBiezacy(2)<0.5
76             %
77             elseif przedzialBiezacy(1)>=0.5 && przedzialBiezacy(2)<1
78                 BuforBitow=decy2buf(buf2decy(BuforBitow)-0.5,Nb);
79                 przedzialBiezacy=przedzialBiezacy-0.5;
80             elseif przedzialBiezacy(1)>=0.25 && przedzialBiezacy(2)<0.75
81                 BuforBitow=decy2buf(buf2decy(BuforBitow)-0.25,Nb);
82                 przedzialBiezacy=przedzialBiezacy-0.25;
83             else
84                 break
85             end
86             przedzialBiezacy=2*przedzialBiezacy;
87             if k<=length(bity)
88                 BuforBitow = [BuforBitow(2:Nb) bity(k)];
89                 k=k+1;
90             else
91                 BuforBitow = [BuforBitow(2:Nb) 0];
92             end
93         end
94     end
95 function liczba = buf2decy(BuforBitow);
96 %zamiana liczby binarnej w postaci 0.101010 na liczbę dziesiętną
97 wagi = 2.^-[1:length(BuforBitow)];
98 liczba = sum(BuforBitow.*wagi);
99 function BuforBitow = decy2buf(liczba,Nb);
100 %zamiana liczby dziesiętnej na binarną w postaci 0.101010
101 wagi = 2.^-[1:Nb];
102 for k=1:Nb;
103     if liczba-wagi(k)>=0;
104         liczba=liczba-wagi(k);
105         BuforBitow(k)=1;
106     else
107         BuforBitow(k)=0;
108     end
109 end
110 function [symbol, prw]=zlicz(x);
111 % x - komunikat - liczby całkowite
112 symbol=[];
113 prw=[];
114 x = x(:);
115 N = length(x);
116 k = 1;

```

```

117 Iz = zeros(size(x));
118 while sum(Iz)<N;
119   Is=find(x==x(k));
120   symbol = [symbol x(k)];
121   prw = [prw length(Is)/N];
122   Iz(Is)=1;
123   Ik=find(Iz==0);
124   if length(Ik)>0
125     k=Ik(1);
126   else
127     break
128 end
129 end

```

## 12.3 Kodowanie słownikowe

Kodowanie słownikowe jest stosowane np. w programach archiwizujących pliki dyskowe takich jak ZIP czy ARJ. Istnieje kilka modyfikacji podstawowego algorytmu LZ77, z których najbardziej zaawansowany jest LZW nazwany tak od nazwisk autorów Lempel, Ziv, Welch. Ogólna zasada konstrukcji koderów słownikowych polega na zastosowaniu dwóch buforów: słownikowego i z danymi do zakodowania. Bufor słownikowy zostaje zainicjowany, a następnie jest rozbudowywany dynamicznie w trakcie kodowania. Kodowanie polega na tym, że w buforze słownikowym wyszukiwane są jak najdłuższe ciągi symboli z buforu danych i wysyłane są indeksy do buforu słownikowego. Różnice pomiędzy algorytmami LZ77, LZ78, LZW polegają na sposobie konstruowania i indeksowania słownika. Słownik jest kluczowym elementem algorytmów z rodzinny LZ jego organizacja oraz efektywne przeszukiwanie stanowi o wydajności kompresji.

Tab.12.1 Kodowanie komunikatu  $x=[1,3,1,2,1,2,1,3,4,2]$  metodą LZW.

koder		słownik	
wejście	wyjście	indeks	hasło
		1	1
		2	3
		3	2
		4	4
1,3	1	5	1,3
3,1	2	6	3,1
1,2	1	7	1,2
2,1	3	8	2,1
1,2,1	7	9	1,2,1
1,3,4	5	10	1,3,4
4,2	4	11	4,2
2	3		

Jako przykład rozważmy zakodowanie komunikatu  $x=[1,3,1,2,1,2,1,3,4,2]$  metodą LZW. Proces kodowania przedstawia tab.12.1. Przed rozpoczęciem kodowania inicjowany jest słownik w postaci wszystkich symboli występujących w komunikacie, tzn.  $S=\{1,2,3,4\}$ . Każdy z symboli otrzymuje indeks w słowniku. Kodowanie polega na wyszukaniu w słowniku jak najdłuższego bieżącego fragmentu komunikatu. Pierwsze kroki kodowania są następujące. Wejściem jest symbol 1, ponieważ symbol ten znajduje się w słowniku (na pozycji 1), więc wczytywany jest kolejny symbol tj.3. Hasła {1,3} nie ma jeszcze w słowniku, więc zostaje ono dodane na pozycji 5 i wysyłany jest indeks symbolu 1. Symbol 3 jest w słowniku, więc wczytywany jest kolejny symbol 1, ponieważ hasła {3,1} nie ma w

słowniku, więc zostaje ono dodane, a wysyłany jest indeks symbolu 3, itd. Podczas dekodowania słownik jest odbudowywany przez dekoder w taki sam sposób.

Poniższy program realizuje kodowanie i dekodowanie metodą LZW wg opisanego algorytmu. Sygnał testowy w linii 3 jest przypadkiem specjalnym dekodowania uwzględnionym w dekoderze w liniach 65-67. Linie 8-12 porównują zmienne tekstowe w celu określenia poprawności kodowania - dekodowania.

```
1 function start
2 clc
3 %x='1212121333433433431' %przypadek specjalny
4 x='1312121342';
5 slownik = slownik_init(x)
6 kod=lzw_koder(x,slownik)
7 y =lzw_dekoder(kod,slownik)
8 if strcmp(y,{x}),
9     'Kompresja-dekompresja poprawna'
10 else
11     'Błąd kompresji-dekompresji'
12 end
13 function kod=lzw_koder(x,slownik)
14 kod=[];
15 ciag_c=x(1);
16 for k=2:length(x)
17     symbol_s=x(k);
18     cs=[ciag_c symbol_s];
19     indeks=jest_w_slowniku(cs, slownik);
20     if indeks
21         ciag_c=cs;
22     else
23         indeks=jest_w_slowniku(ciag_c, slownik);
24         slownik=wlacz_do_slownika(cs, slownik);
25         ciag_c=symbol_s;
26         kod=[kod indeks];
27     end
28 end
29 indeks=jest_w_slowniku(ciag_c, slownik);
30 kod=[kod indeks];
31 function slownik = slownik_init(x);
32 %słownik - 1-sze wystąpienia
33 N = length(x);
34 k = 1;
35 k1= 1;
36 Iz = zeros(size(x));
37 while sum(Iz)<N;
38     Is=find( x==x(k)); Iz(Is)=1;
39     Ik=find(Iz==0);
40     slownik(k1)={x(k)};
41     if length(Ik)>0
42         k=Ik(1);
43         k1=k1+1;
44     else
45         break
46     end
47 end
48 function [indeks]=jest_w_slowniku(symbol, slownik);
49 indeks=0;
50 for k=1:length(slownik);
51     if strcmp(slownik(k),symbol); indeks=k; break, end
52 end
53 function slownik=wlacz_do_slownika(ciag, slownik);
54 slownik(length(slownik)+1)={ciag};
55 function y=lzw_dekoder(kod,slownik)
56 poprzedniCiag=slownik{kod(1)};
57 y(1)=poprzedniCiag;
58 for k=2:length(kod)
59     indeks=kod(k);
```

```
60     if indeks<=length(slownik)
61         C=slownik{indeks};
62         cs=[poprzedniCiag C(1)];
63         slownik=wlacz_do_slownika(cs, slownik);
64     else
65         cs=[C C(1)];
66         slownik=wlacz_do_slownika(cs, slownik);
67         C=slownik{indeks};
68     end
69     y=[y C];
70     poprzedniCiag=C;
71 end
```

## 13. Obrazy cyfrowe

Obrazy cyfrowe, oznaczane jako sygnały 2D (*two dimensional*), są reprezentowane przez macierze dwuwymiarowe. Podstawowe reprezentacje obrazów cyfrowych są następujące:

1. Obrazy binarne (*binary image*) - macierze zawierające elementy 0 i 1, piksel ciemny oznaczany jest przez 0.
2. Obrazy w skali jasności (*intensity image, gray scale image*) - macierze zawierające liczby stałoprzecinkowe lub zmiennoprzecinkowe określające jasność piksela.
3. *True color image, RGB (Red, Green, Blue) image* - obraz składa się z trzech macierzy, które określają jasności barw podstawowych: czerwonej, zielonej i niebieskiej.
4. *Indexed image, pseudocolor image* - macierz obrazu zawiera indeksy do osobnej macierzy kolorów (*colormap*) o rozmiarze  $M \times 3$ , której kolumny określają jasności barw podstawowych.

Omówione dalej techniki przetwarzania obrazów dotyczą ich reprezentacji w skali jasności.

### 13.1 Filtracja 2D

Splot liniowy sygnałów dwuwymiarowych zdefiniowany jest równaniem:

$$y[n_1, n_2] = \sum_{k_1=-\infty}^{\infty} \sum_{k_2=-\infty}^{\infty} x[k_1, k_2] h[n_1 - k_1, n_2 - k_2] = x[n_1, n_2] * h[n_1, n_2]. \quad (13.1)$$

Współczynniki  $k_1$  i  $k_2$  w (13.1) powodują obrócenie odpowiedzi impulsowej filtra  $h[n_1, n_2]$  o  $180^\circ$ , następnie macierz  $h[n_1, n_2]$  jest przesuwana względem macierzy obrazu  $x[n_1, n_2]$ , a wynik splotu  $y[n_1, n_2]$  jest sumą elementów aktualnego fragmentu obrazu ważoną przez współczynniki filtra.

Jako przykład liczenia splotu liniowego 2D sygnałów o skończonych rozmiarach rozważmy:

$$h[n_1, n_2] = \begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{bmatrix} \quad \text{i} \quad x[n_1, n_2] = \begin{bmatrix} 4 & 6 & 9 & 6 & 1 \\ 3 & 1 & 3 & 10 & 8 \\ 3 & 0 & 2 & 7 & 4 \\ 4 & 5 & 9 & 9 & 9 \\ 4 & 9 & 2 & 0 & 7 \end{bmatrix}. \quad (13.2)$$

Splot tych sygnałów wynosi:

$$y[n_1, n_2] = x[n_1, n_2] * h[n_1, n_2] = \begin{bmatrix} 4 & 14 & 33 & 42 & 40 & 20 & 3 \\ 19 & 51 & 104 & 124 & 125 & 87 & 30 \\ 43 & 99 & 193 & 240 & 260 & 191 & 69 \\ 37 & 59 & 113 & 183 & 280 & 261 & 123 \\ 41 & 81 & 158 & 207 & 250 & 208 & 111 \\ 44 & 123 & 216 & 244 & 256 & 188 & 123 \\ 28 & 95 & 122 & 97 & 67 & 56 & 63 \end{bmatrix}. \quad (13.3a)$$

Rozmiar macierzy  $y[n_1, n_2]$  wynosi  $(N_x + N_h - 1) \times (M_x + M_h - 1)$ , gdzie  $N_x \times M_x$  i  $N_h \times M_h$  są rozmiarami obrazu i filtra.

Wzór (13.4b) przedstawia poglądowo jeden krok liczenia splotu 2D (13.1). Pierwsze czynniki w macierzy są sygnałem wejściowym  $x[n_1, n_2]$ , natomiast drugie czynniki są współczynnikami filtra  $h[n_1, n_2]$  obróconymi o  $180^\circ$ . Wynikiem filtracji, dla tego

położenia filtra, jest piksel o wartości 193 w  $y[n_1, n_2]$ . W celu wyznaczenia wszystkich pikseli  $y[n_1, n_2]$  filtr  $h[n_1, n_2]$  jest przesuwany względem  $x[n_1, n_2]$  o pozycję jednego piksela i dla każdego przesunięcia wyznaczana jest suma iloczynów współczynników filtra z aktualnym fragmentem obrazu  $x[n_1, n_2]$ .

$$y[n_1, n_2] = x[n_1, n_2] * h[n_1, n_2] = \begin{bmatrix} - & - & - & - & - & - & - \\ - & 4 \cdot 9 & 6 \cdot 8 & 9 \cdot 7 & - & - & - \\ - & 3 \cdot 6 & 1 \cdot 5 & 3 \cdot 4 & - & - & - \\ - & 3 \cdot 3 & 0 \cdot 2 & 2 \cdot 1 & - & - & - \\ - & - & - & - & - & - & - \\ - & - & - & - & - & - & - \\ - & - & - & - & - & - & - \end{bmatrix} = \begin{bmatrix} - & - & - & - & - & - & - \\ - & - & - & - & - & - & - \\ - & - & 193 & - & - & - & - \\ - & - & - & - & - & - & - \\ - & - & - & - & - & - & - \\ - & - & - & - & - & - & - \\ - & - & - & - & - & - & - \end{bmatrix} \quad (13.3b)$$

Obliczanie splotu 2D wg definicji (13.1) zakłada, że obraz  $x[n_1, n_2]$  ma wartości zerowe poza zadanym zakresem indeksów, co może powodować znaczne zakłócenia na brzegach obrazu. Zakłócenia te mogą być zmniejszone przez lustrzane odbicie sygnału na jego brzegach. Obie metody obliczeń zilustrowane są na rys. 13.1.

a)	b)
$\begin{bmatrix} 0 & 0 & 0 & 0 & - & - & - \\ 0 & 4 \cdot 9 & 6 \cdot 8 & 9 \cdot 7 & - & - & - \\ 0 & 3 \cdot 6 & 1 \cdot 5 & 3 \cdot 4 & - & - & - \\ 0 & 3 \cdot 3 & 0 \cdot 2 & 2 \cdot 1 & - & - & - \\ - & - & - & - & - & - & - \\ - & - & - & - & - & - & - \\ - & - & - & - & - & - & - \end{bmatrix}$	$\begin{bmatrix} 4 & 4 & 6 & 9 & - & - & - \\ 4 & 4 \cdot 9 & 6 \cdot 8 & 9 \cdot 7 & - & - & - \\ 3 & 3 \cdot 6 & 1 \cdot 5 & 3 \cdot 4 & - & - & - \\ 3 & 3 \cdot 3 & 0 \cdot 2 & 2 \cdot 1 & - & - & - \\ - & - & - & - & - & - & - \\ - & - & - & - & - & - & - \\ - & - & - & - & - & - & - \end{bmatrix}$

Rys. 13.1 Problem obliczeń splotu 2D na brzegach sygnału: a) poza obrazem wartości uzupełniane są zerami, b) odbicie lustrzane obrazu.

### 13.1.1 Filtry specjalne

Do filtrów specjalnych stosowanych w przetwarzaniu obrazów można zaliczyć m.in. filtry uśredniające, filtry do detekcji krawędzi i filtr medianowy.

Rys. 13.2 przedstawia odpowiedzi impulsowe  $h[n_1, n_2]$  typowych filtrów uśredniających. Filtry uśredniające są często określane jako filtry wygładzające (*smoothing filters*). Odpowiedzi impulsowe są pokazane w postaci powierzchni trójwymiarowych oraz obrazów 2D (widok z góry), gdzie jasność piksela jest proporcjonalna do wartości współczynnika filtra. Odpowiedzi impulsowe filtrów są unormowane w ten sposób, że suma ich współczynników wynosi 1.

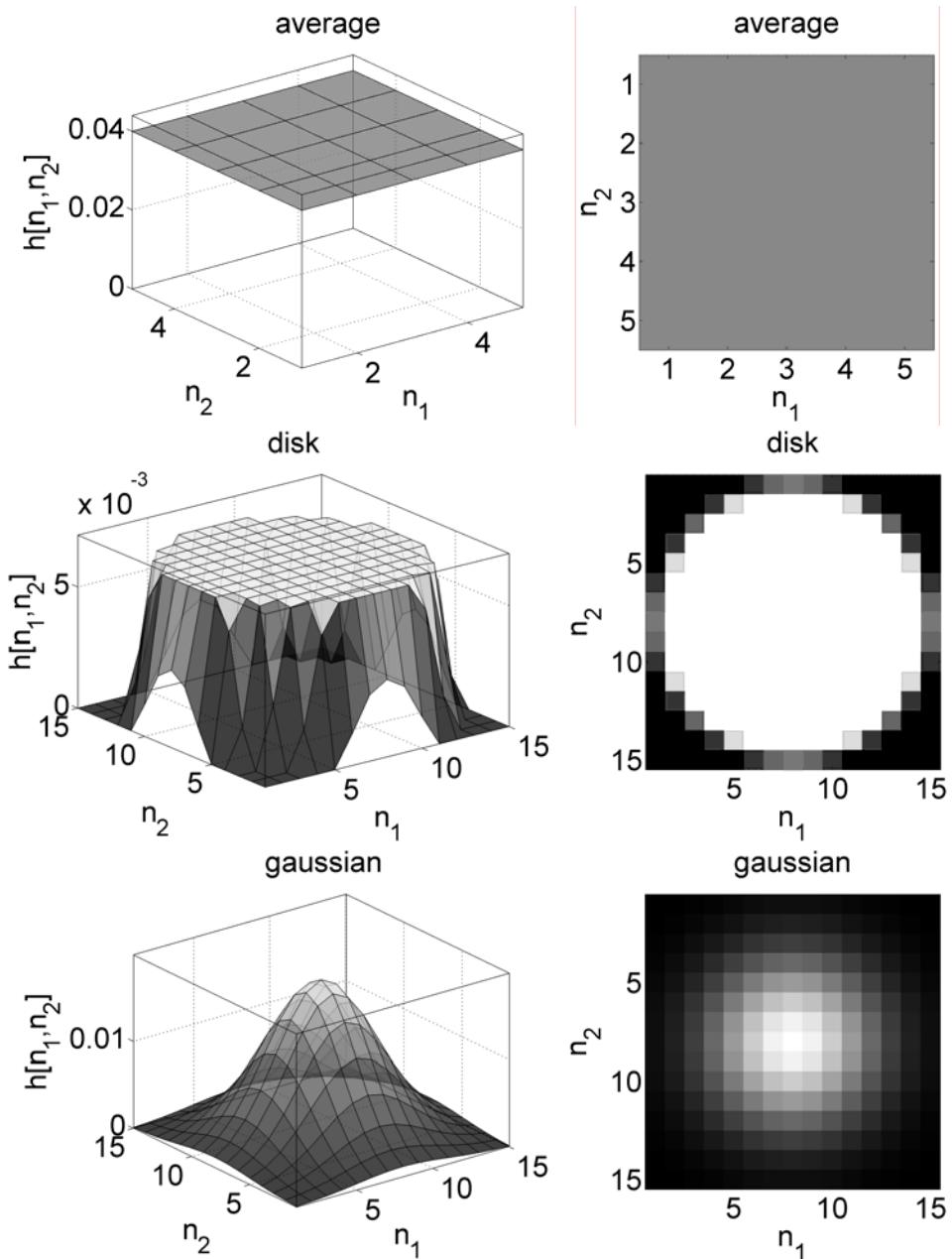
Filtr Gaussa zdefiniowany jest jako:

$$h[n_1, n_2] = h_{Gauss}[n_1, n_2] / \left( \sum_{n_1} \sum_{n_2} h_{Gauss}[n_1, n_2] \right), \quad h_{Gauss}[n_1, n_2] = e^{-(n_1^2 + n_2^2) / 2\sigma^2}, \quad (13.4)$$

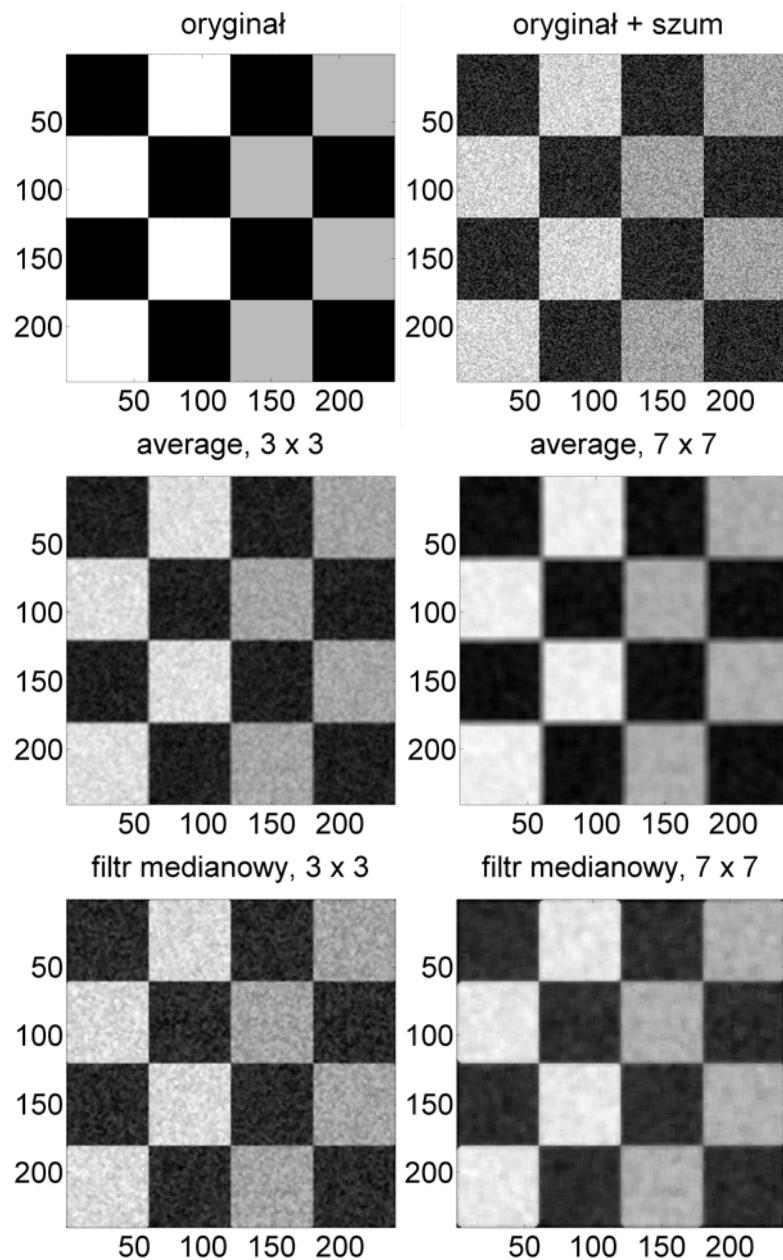
gdzie  $\sigma$  jest odchyleniem standardowym, tj. liczbą dodatnią.

Rys. 13.3 przedstawia zastosowanie filtru uśredniającego do filtracji obrazu zakłóconego szumem addytywnym. Sygnał testowy i sygnał testowy zakłócony szumem są pokazane w pierwszym wierszu rys. 13.3. Wiersz środkowy przedstawia obraz po filtracji filtrem o kwadratowej odpowiedzi impulsowej i rozmiarze  $3 \times 3$  i  $7 \times 7$  pikseli. Jak widać zwiększenie rozmiaru filtra owocuje lepszym usunięciem szumu z obrazu powodując jednak

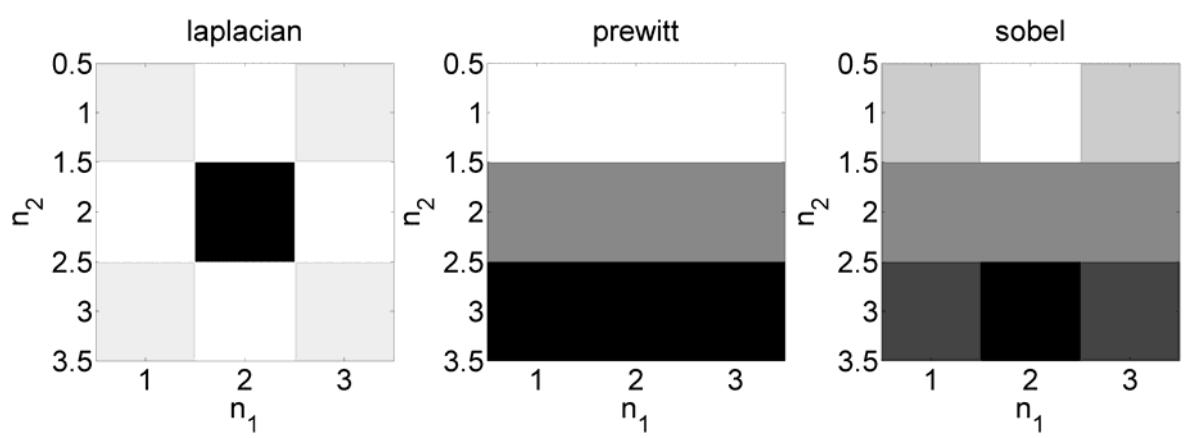
niekorzystne rozmycie krawędzi. Dla porównania rys.13.3 przedstawia również wyniki filtracji medianowej filtrem o tych samych rozmiarach. Filtr medianowy przesuwa się względem obrazu, podobnie jak w przypadku liczenia splotu, z tą różnicą, że wynikiem filtracji jest mediana z aktualnego fragmentu obrazu. Filtr medianowy jest nieliniowy, jego główną zaletą jest usuwanie z obrazu szumu o charakterze impulsowym, przy jednoczesnym zachowaniu ostrzych krawędzi, co widoczne jest na rys.13.3.



Rys.13.2 Odpowiedzi impulsowe filtrów uśredniających.



Rys.13.3 Wyniki filtracji sygnału zakłóconego szumem dla filtra uśredniającego i filtra medianowego.



Rys.13.4 Odpowiedzi impulsowe filtrów do detekcji krawędzi.

Rys.13.4 przedstawia odpowiedzi impulsowe filtrów do detekcji krawędzi, których wartości liczbowe są następujące:

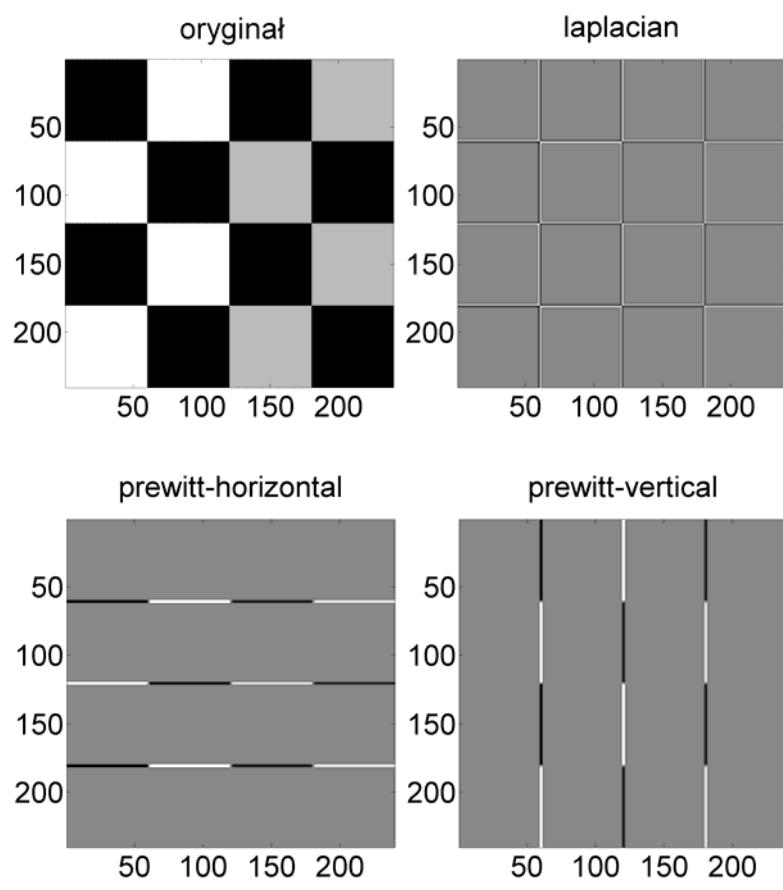
$$h_{laplacian} = \begin{bmatrix} 0.17 & 0.67 & 0.17 \\ 0.67 & -3.33 & 0.67 \\ 0.17 & 0.67 & 0.17 \end{bmatrix}, h_{prewitt} = \begin{bmatrix} 1 & 1 & 1 \\ 0 & 0 & 0 \\ -1 & -1 & -1 \end{bmatrix}, h_{sobel} = \begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix}. \quad (13.5)$$

Filtr  $h_{laplacian}$  aproksymuje gradient po współrzędnych  $n_1$  i  $n_2$ . Filtr ten jest zdefiniowany następująco:

$$h_{laplacian}[n_1, n_2] = \left( (n_1^2 + n_2^2 - 2\sigma^2) h_{Gauss}[n_1, n_2] \right) / \left( 2\pi\sigma^6 \sum_{n_1} \sum_{n_2} h_{Gauss}[n_1, n_2] \right), \quad (13.6)$$

gdzie  $h_{Gauss}$  zdefiniowany jest przez (13.4).

Rys.13.5 przedstawia wyniki filtracji obrazu testowego filtrami do detekcji krawędzi. Filtr  $h_{laplacian}$  wykrywa krawędzie wertykalne i horyzontalne. Filtr  $h_{prewitt}$  pozostawia w obrazie krawędzie horyzontalne jednak obracając go o 90 stopni można również detektować krawędzie wertykalne.



Rys.13.5 Detekcja krawędzi za pomocą filtra laplaciana, prewitta i transponowanego prewitta.

## 13.2 Transformacja Fouriera obrazów

Ciągła transformacja Fouriera dla dyskretnych sygnałów 2D jest zdefiniowana:

$$X(\omega_1, \omega_2) = \sum_{n_1=-\infty}^{\infty} \sum_{n_2=-\infty}^{\infty} x[n_1, n_2] e^{-j\omega_1 n_1} e^{-j\omega_2 n_2}. \quad (13.7)$$

Transformacja odwrotna ma postać:

$$x[n_1, n_2] = \frac{1}{4\pi^2} \int_{\omega_1=-\pi}^{\pi} \int_{\omega_2=-\pi}^{\pi} X(\omega_1, \omega_2) e^{j\omega_1 n_1} e^{j\omega_2 n_2} d\omega_1 d\omega_2. \quad (13.8)$$

Ciągłe widmo dyskretnego sygnału 2D jest okresowe z okresem  $2\pi \times 2\pi$ , tzn.:

$$X(\omega_1, \omega_2) = X(\omega_1 + 2\pi, \omega_2) = X(\omega_1, \omega_2 + 2\pi). \quad (13.9)$$

Większość własności TF dla sygnałów 1D takich jak: liniowość, przesunięcie w czasie, przesunięcie w częstotliwości, odwrócenie w czasie, różniczkowanie w częstotliwości, splot w czasie, splot w częstotliwości i zachowanie energii przenosi się bezpośrednio do przypadku 2D.

Dla sygnałów dwuwymiarowych występuje dodatkowa własność dla ciągów separowalnych. Jeżeli sygnał 2D można przedstawić w postaci iloczynu zewnętrznego (*outer product*) ciągów jednowymiarowych

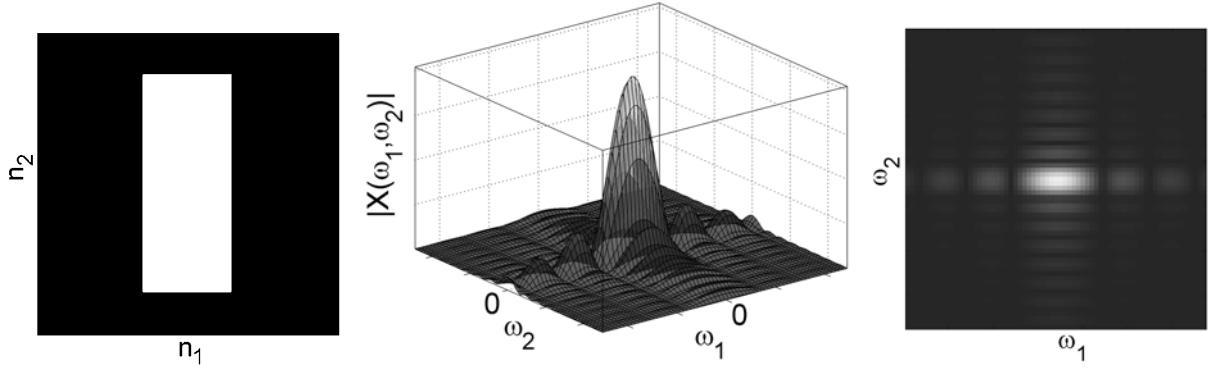
$$x[n_1, n_2] = x_1[n_1] x_2[n_2] = \begin{bmatrix} x_1[1] \\ x_1[2] \\ \vdots \end{bmatrix} \begin{bmatrix} x_2[1] & x_2[2] & \cdots \end{bmatrix} = \begin{bmatrix} x_1[1] x_2[1] & x_1[1] x_2[2] & \cdots \\ x_1[2] x_2[1] & x_1[2] x_2[2] & \cdots \\ \vdots & \vdots & \ddots \end{bmatrix} \quad (13.10)$$

to jest to tzw. sygnał separowalny i jego widmo jest iloczynem zewnętrznym widm sygnałów jednowymiarowych:

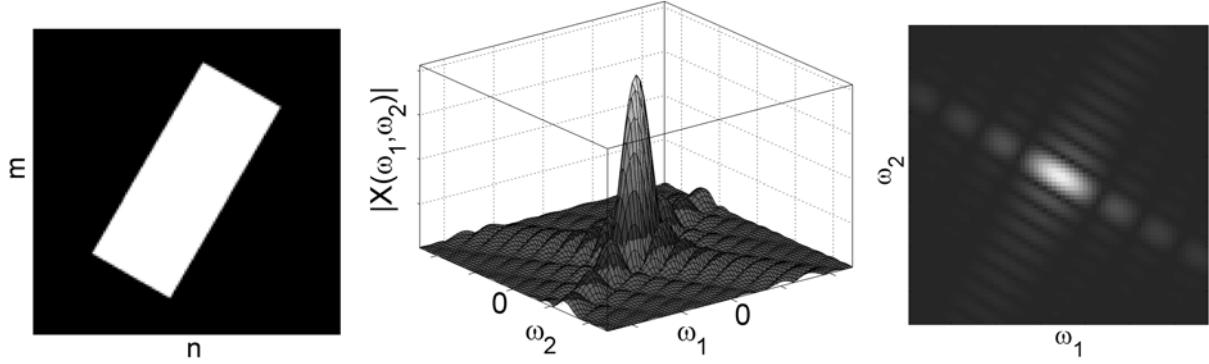
$$X(\omega_1, \omega_2) = X_1(\omega_1) X_2(\omega_2). \quad (13.11)$$

W definicji (13.7) kolejność sumowania jest dowolna, można więc najpierw wykonać sumowanie po  $n_1$  a następnie po  $n_2$ . Sumowanie po  $n_1$  są to 1D transformaty Fouriera wierszy obrazu, a sumowanie po  $n_2$  są to 1D transformaty Fouriera jego kolumn. Widmo obrazu można więc obliczyć stosując jednowymiarową transformację Fouriera najpierw do jego wierszy, a następnie do jego kolumn.

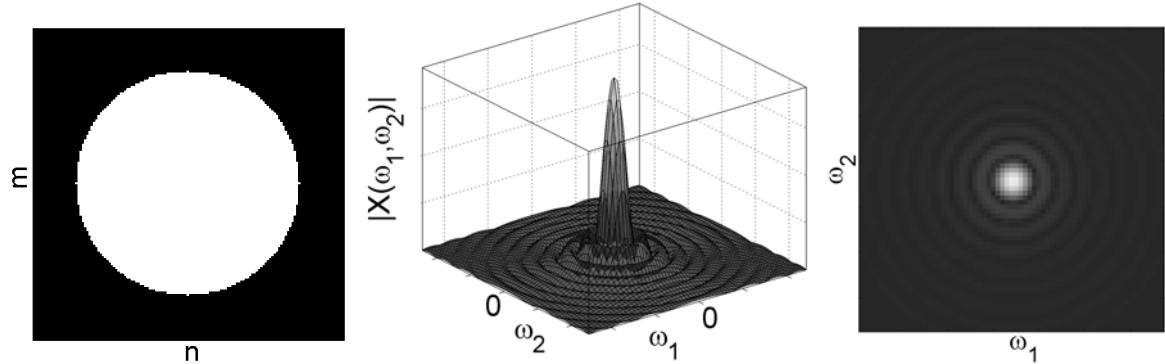
Rys.13.6 przedstawia widmo obrazu o kształcie prostokąta. Jest to przykład sygnału separowalnego. Jak widać z rys.13.7 obrót obrazu powoduje analogiczny obrót jego widma. Rys.13.8 przedstawia widmo obrazu o kształcie cylindra, widmo to ma również kształt cylindryczny.



Rys.13.6 Obraz i jego widmo Fouriera.



Rys.13.7 Obraz i jego widmo Fouriera.



Rys.13.8 Obraz i jego widmo Fouriera.

### 13.3 Projektowanie filtrów 2D metodą okien

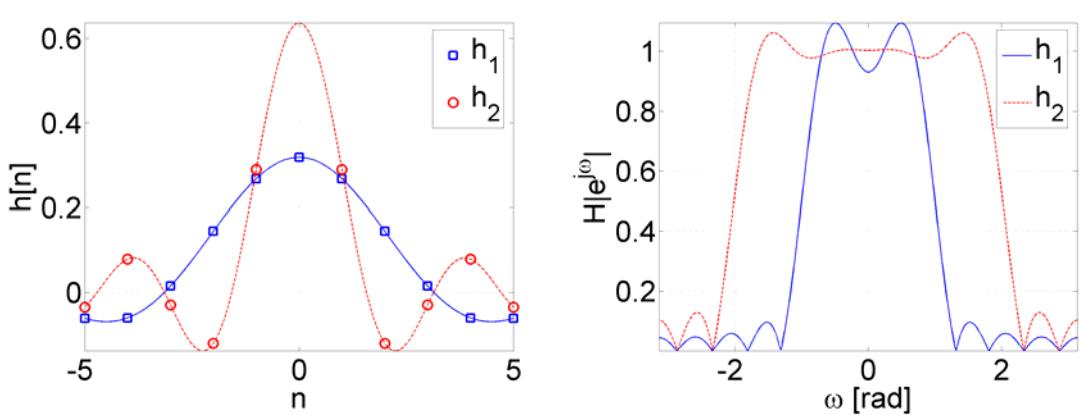
Odpowiedzi impulsowe filtrów FIR 2D mogą być projektowane jako separowalne (prostokątne) lub cylindryczne (*circularly symmetric*).

Jako przykład filtra o separowalnej odpowiedzi impulsowej rozważmy filtr dolnoprzepustowy. Odpowiedź impulsowa idealnego filtra 1D LP jest następująca:

$$h[n] = \begin{cases} \frac{\sin(\omega_c n)}{\pi n}, & n \neq 0, \\ \frac{\omega_c}{\pi}, & n = 0. \end{cases} \quad (13.12)$$

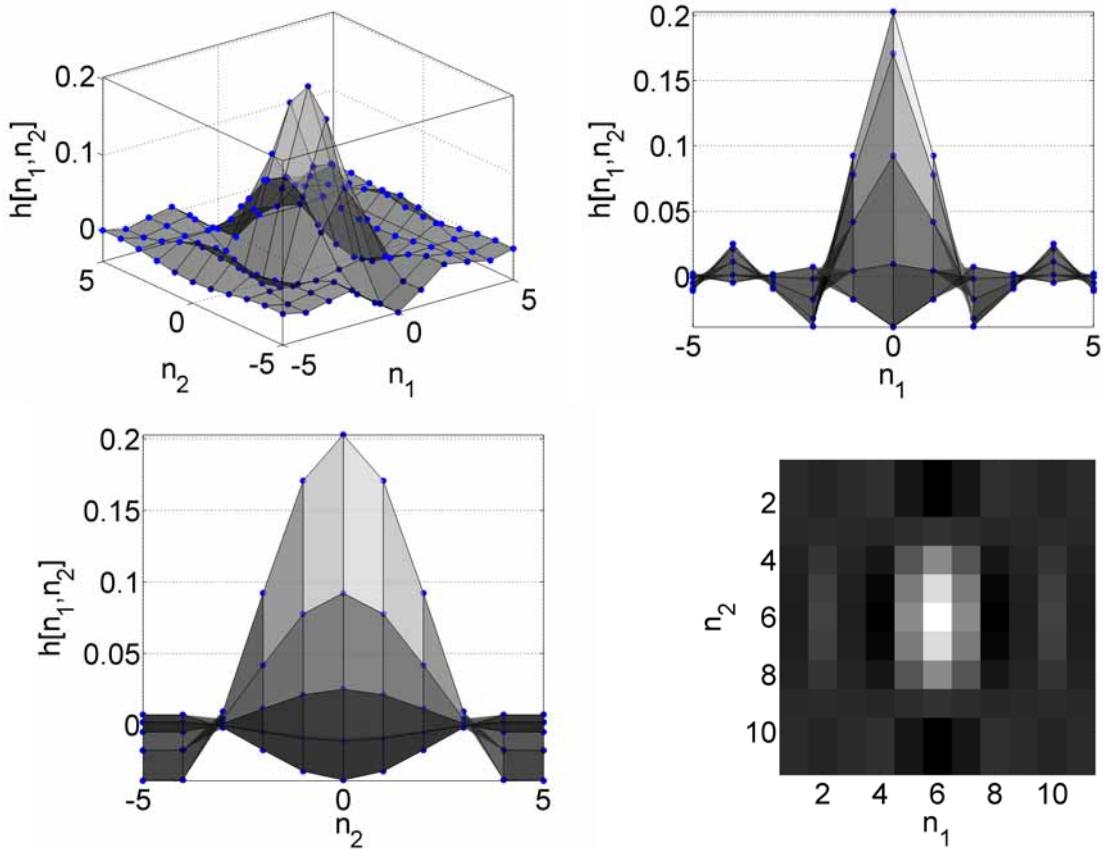
Korzystając z (13.10) współczynniki filtra 2D otrzymuje się w następujący sposób:

$$h[n_1, n_2] = h_1[n_1]h_2[n_2] = \frac{\sin(\omega_{c1}n_1)}{\pi n_1} \frac{\sin(\omega_{c2}n_2)}{\pi n_2}. \quad (13.13)$$

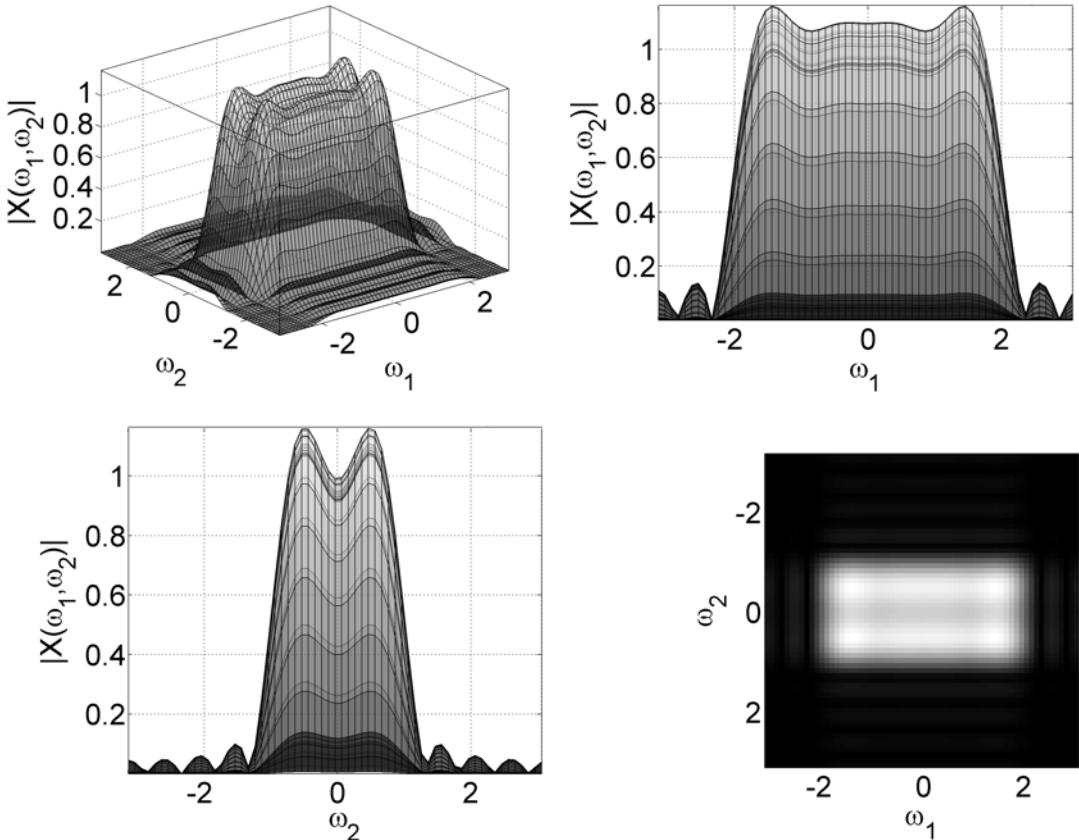


Rys.13.9 Odpowiedzi impulsowe i charakterystyki amplitudowe filtrów 1D.

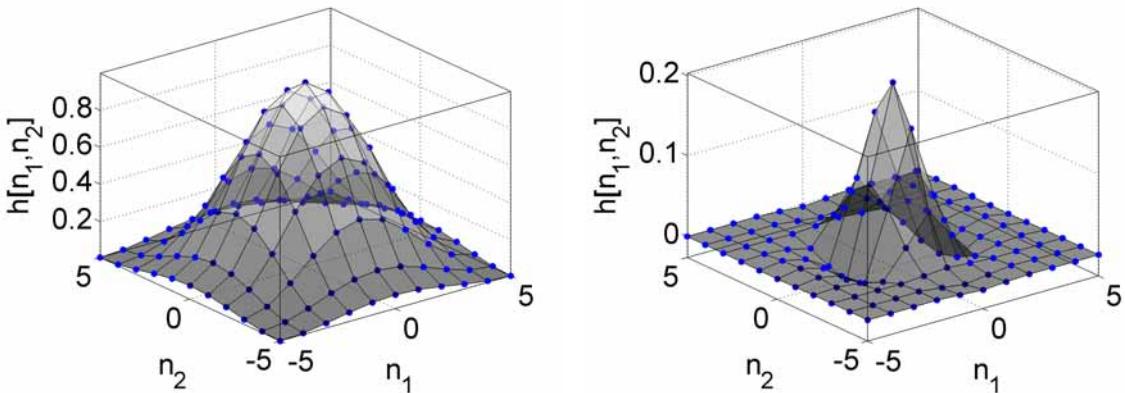
Rys.13.9 przedstawia odpowiedzi impulsowe oraz charakterystyki amplitudowe dwóch filtrów dolnoprzepustowych. W wyniku wymnożenia tych odpowiedzi impulsowych wg (13.10) otrzymujemy współczynniki filtra 2D pokazane na rys.13.10. Charakterystykę amplitudową otrzymanego filtra 2D przedstawia rys.13.11. Widok z góry na tę charakterystykę ma kształt prostokąta.



Rys.13.10 Odpowiedź impulsowa separowalnego filtra 2D, widok ogólny i rzuty wzdłuż osi.



Rys.13.11 Charakterystyka amplitudowa separowalnego filtra 2D, widok ogólny i rzuty wzdłuż osi.

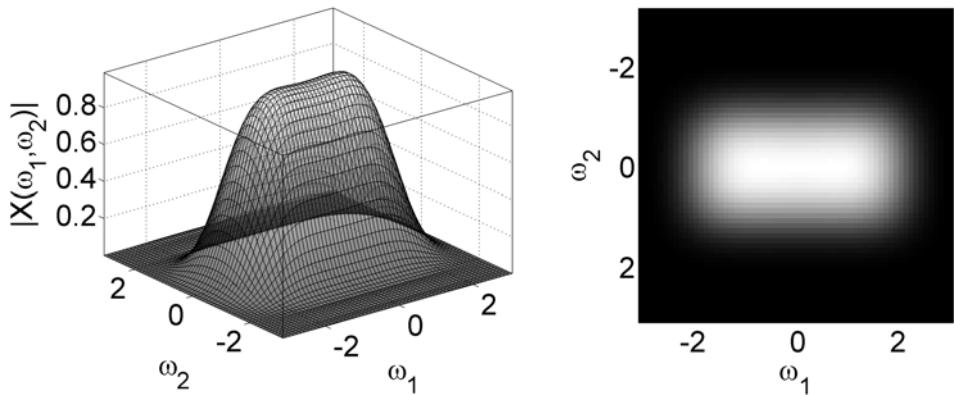


Rys.13.12 Separowalne okno 2D Hamminga i odpowiedź impulsowa filtra z rys.13.10 przemnożona przez to okno.

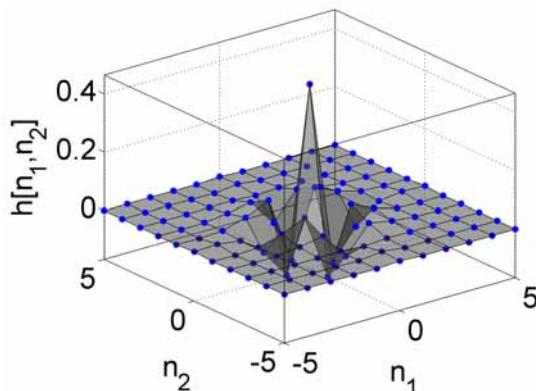
Odpowiedź impulsowa filtra 2D (13.13) może być ograniczona oknem innym niż prostokątne. Separowalne okno 2D otrzymuje się tak samo jak współczynniki filtru, tj. przez wymnożenie dwóch okien 1D wg (13.10). Separowalne okno Hamminga oraz odpowiedź impulsowa filtra przemnożona przez to okno są przedstawione na rys.13.12. Charakterystyka amplitudowa filtra 2D z oknem Hamminga jest pokazana na rys.13.13. W porównaniu do okna prostokątnego jest ona mniej zafalowana, ale też mniej stroma.

Stosując opisaną wyżej procedurę można projektować dowolne filtry pasmowe, tj. HP, BP i BS na podstawie odpowiednich odpowiedzi impulsowych filtrów 1D.

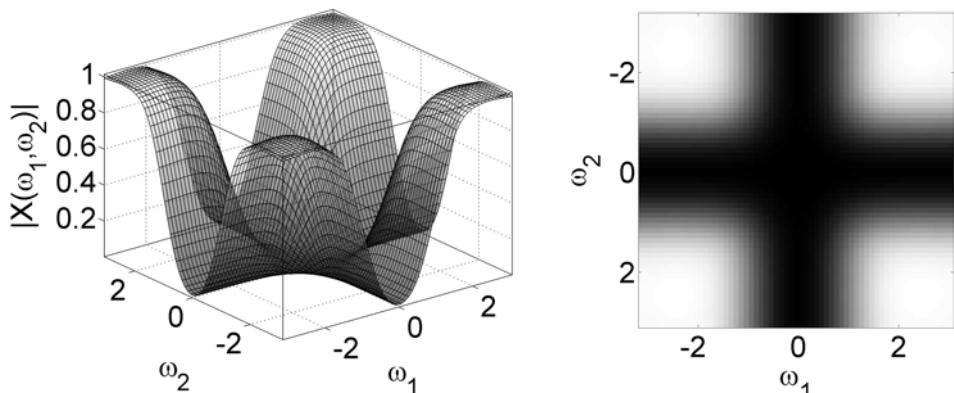
Rys.13.15 przedstawia charakterystykę amplitudową separowalnego filtra HP z oknem Hamminga.



Rys.13.13 Charakterystyka amplitudowa filtra separowalnego z oknem Hamminga.



Rys.13.14 Odpowiedź impulsowa filtra separowalnego HP.



Rys.13.15 Charakterystyka amplitudowa filtra separowalnego HP z oknem Hamminga.

W przypadku filtrów cylindrycznych odpowiedź impulsowa LP jest następująca:

$$h_{FDP}[n_1, n_2] = \begin{cases} \frac{\omega_c J_1(\omega_c r)}{2\pi r}, & r \neq 0 \\ \omega_c^2 / (4\pi), & r = 0 \end{cases}, \quad r = \sqrt{n_1^2 + n_2^2}, \quad -N \leq n_1, n_2 \leq N, \quad (13.14)$$

gdzie  $J_1(\cdot)$  jest funkcją Bessela pierwszego rodzaju i pierwszego stopnia, którą można rozwinać w następujący szereg:

$$J_1(x) = \frac{x}{2} - \frac{x^3}{2^3 1! 2!} + \frac{x^5}{2^5 2! 3!} - \frac{x^7}{2^7 3! 4!} + \frac{x^9}{2^9 4! 5!} - \dots \quad (13.15)$$

Zgodnie z (13.14) jednowymiarowa funkcja Bessela jest obracana względem punktu  $r=0$ , tak więc współczynniki filtra zależą tylko od odległości od tego punktu.

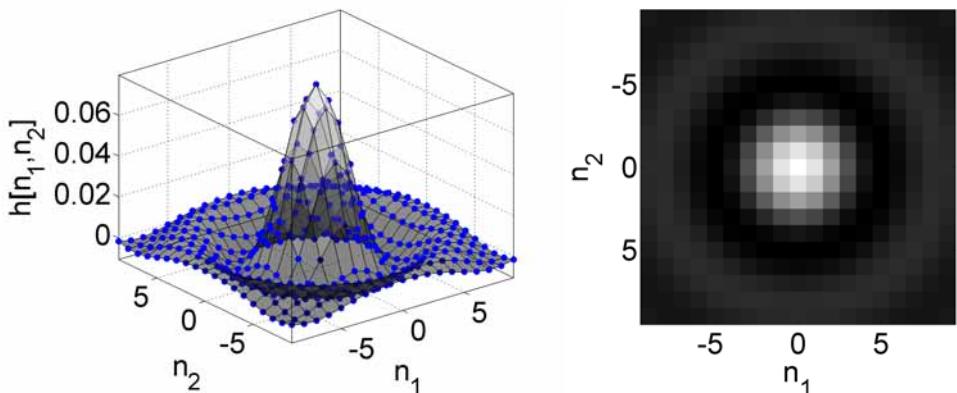
Rys.13.16 przedstawia odpowiedź impulsową filtra cylindrycznego wyznaczoną z (13.14), a rys.13.17 charakterystykę amplitudową tego filtra. Odpowiedź impulsowa filtra może być ograniczona oknem innym niż prostokątne. Stosowane są zarówno okna separowalne jak i cylindryczne. Współczynnik okna cylindrycznego wyznacza się z zależności:

$$w_c[n_1, n_2] = w[r], \quad r = \sqrt{n_1^2 + n_2^2}, \quad (13.16)$$

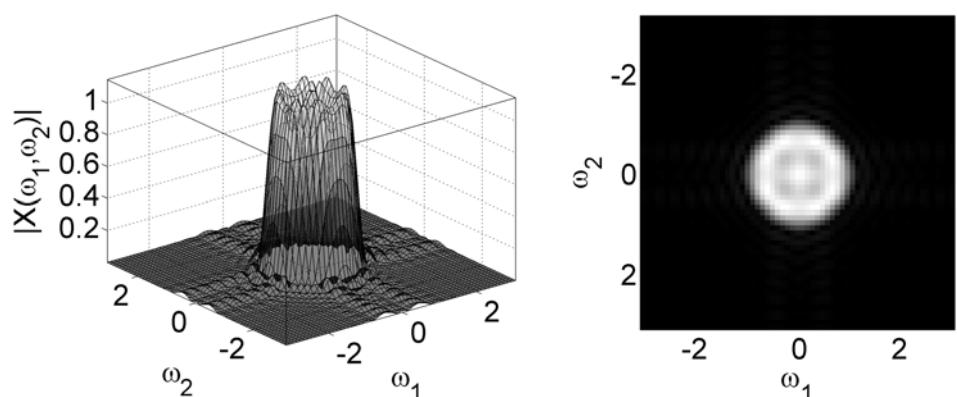
jest to więc obrót jednowymiarowego okna  $w[r]$  wokół jego środka. Okno cylindryczne można wyznaczyć dla każdego okna 1D, np. dla okna Hamminga:

$$w[r] = \begin{cases} 0.54 - 0.46 \cos(2\pi r / M), & 0 \leq r \leq M, \\ 0, & \text{w przeciwnym przypadku.} \end{cases} \quad (13.17)$$

Należy zaznaczyć, że utworzone wg (13.16) okno nie ma tych samych własności, co okno 1D, tzn. jego widmo nie jest obróconym widmem okna 1D.

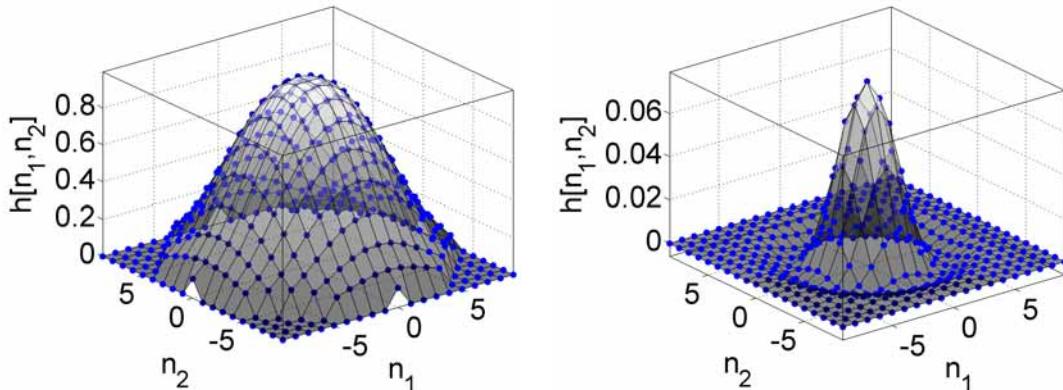


Rys.13.16 Odpowiedź impulsowa filtra cylindrycznego LP z oknem prostokątnym.

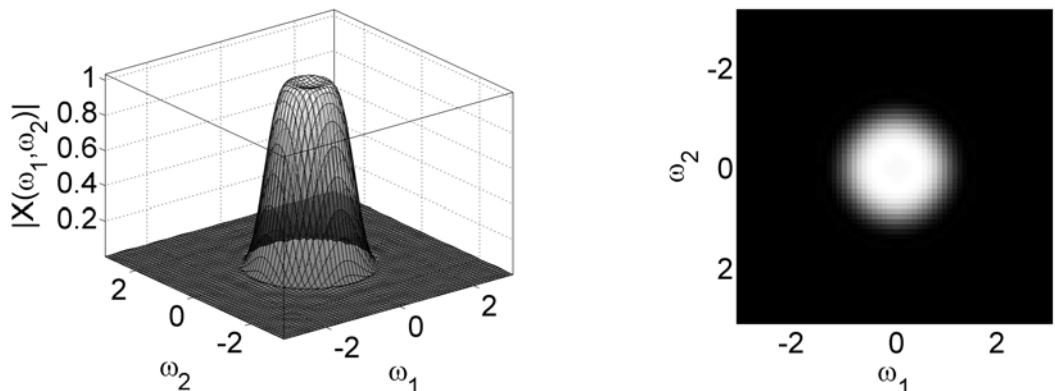


Rys.13.17 Charakterystyka amplitudowa filtra cylindrycznego LP z oknem prostokątnym.

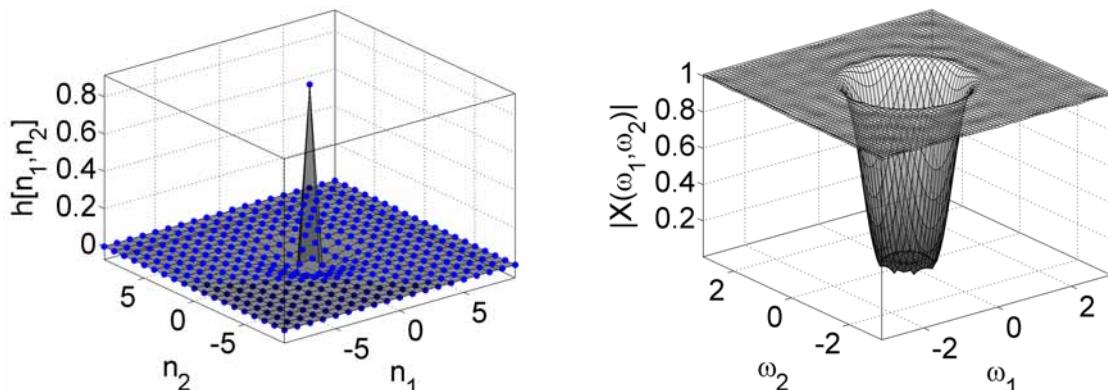
Cylindryczne okno Hamminga jest pokazane na rys.13.18. Widoczna jest różnica pomiędzy oknem cylindrycznym z rys.13.18 i oknem separowalnym z rys.13.12. Zastosowanie okna Hamminga powoduje redukcję zafalowania charakterystyki amplitudowej filtra, co pokazano na rys.13.19.



Rys.13.18 Cylindryczne okno Hamminga i odpowiedź impulsowa filtra cylindrycznego LP z tym oknem.



Rys.13.19 Charakterystyka amplitudowa filtra cylindrycznego LP z oknem Hamminga.



Rys.13.20 Odpowiedź impulsowa i charakterystyka amplitudowa filtra cylindrycznego HP z oknem Hamminga.

Pozostałe filtry pasmowe uzyskuje się przez odpowiednie sumowanie odpowiedzi impulsowych z filtrem wszechprzepustowym  $h_{FALL}[n_1, n_2] = \delta[n_1, n_2]$ , podobnie jak dla filtrów 1D. Na przykład odpowiedź impulsowa filtra HP wynosi:

$$h_{FGP}[n_1, n_2] = h_{FALL}[n_1, n_2] - h_{FDP}[n_1, n_2] = \begin{cases} -\frac{\omega_c J(\omega_c r)}{2\pi r}, & r \neq 0 \\ 1 - \omega_c^2/(4\pi), & r = 0 \end{cases}. \quad (13.18)$$

Odpowiedź impulsową i charakterystykę amplitudową cylindrycznego filtra HP przedstawia rys.13.20.

### 13.4 DFT obrazów

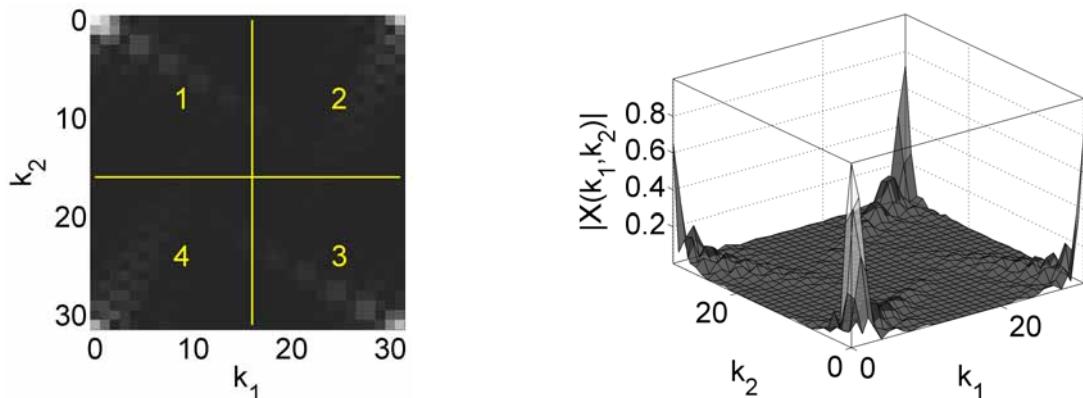
Dla dyskretnego sygnału 2D o rozmiarze  $N_1 \times N_2$  dyskretne przekształcenie Fouriera definiuje się następująco:

$$X[k_1, k_2] = \sum_{n_1=0}^{N_1-1} \sum_{n_2=0}^{N_2-1} x[n_1, n_2] e^{-j\frac{2\pi}{N_1} k_1 n_1} e^{-j\frac{2\pi}{N_2} k_2 n_2}, \quad k_1 = 0, 1, \dots, N_1 - 1, k_2 = 0, 1, \dots, N_2 - 1. \quad (13.19)$$

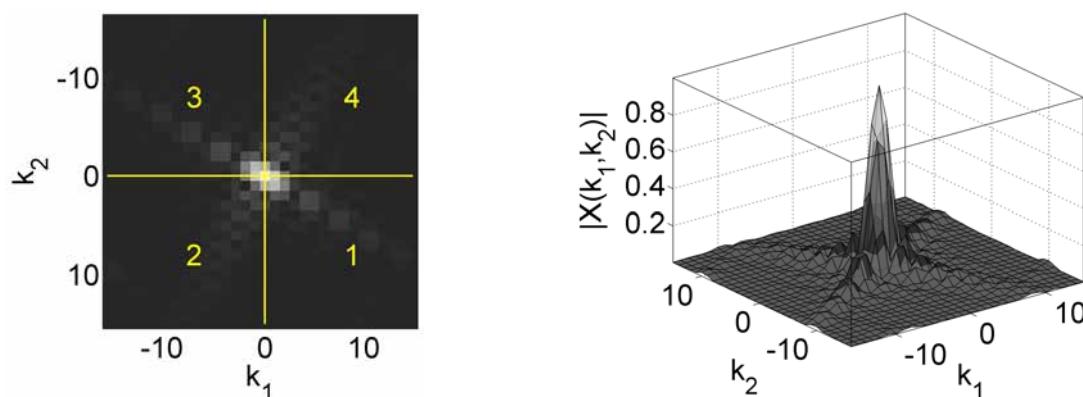
Przekształcenie odwrotne ma postać:

$$x[n_1, n_2] = \frac{1}{N_1 N_2} \sum_{k_1=0}^{N_1-1} \sum_{k_2=0}^{N_2-1} X[k_1, k_2] e^{j\frac{2\pi}{N_1} k_1 n_1} e^{j\frac{2\pi}{N_2} k_2 n_2}, \quad n_1 = 0, 1, \dots, N_1 - 1, n_2 = 0, 1, \dots, N_2 - 1. \quad (13.20)$$

Porównując (13.7) i (13.19) widzimy, że DFT próbkuję ciągłe widmo sygnału dyskretnego w punktach  $\omega_1 = (2\pi/N_1)k_1$ ,  $\omega_2 = (2\pi/N_2)k_2$ . W celu gęstszyego próbkowania widma do obrazu  $x[n_1, n_2]$  można dodać zera, które zwiększą jego rozmiar, tj.  $N_1$  i  $N_2$ .



Rys.13.21 Widmo DFT obrazu.



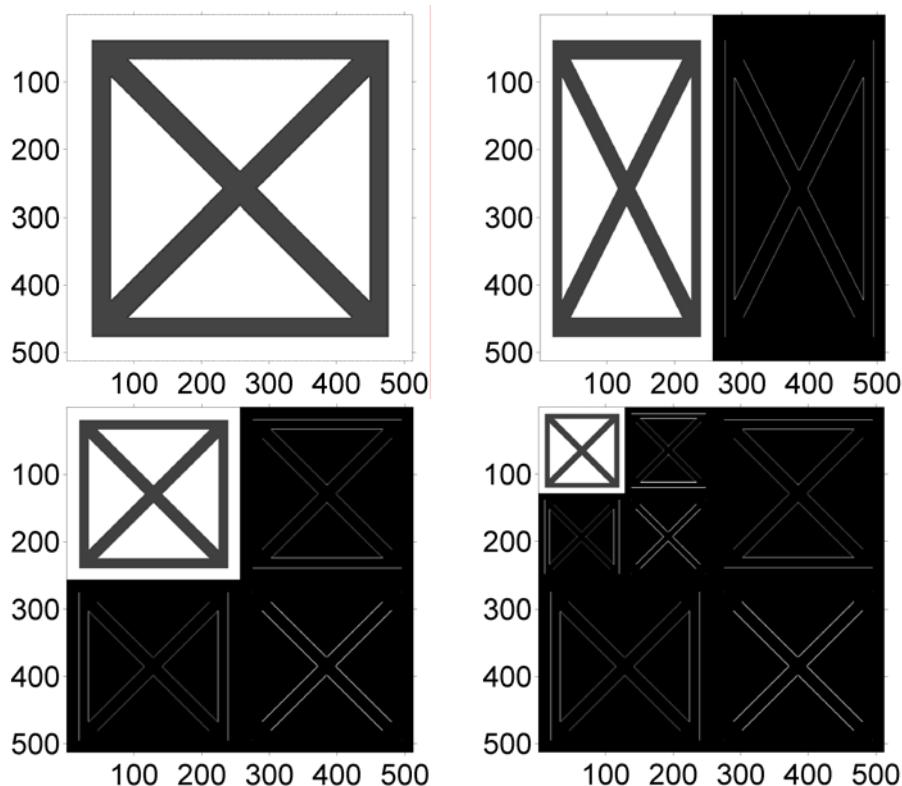
Rys.13.22 Widmo DFT obrazu po zamianie ćwiartek.

DFT 2D (13.19) wyznacza się przez DFT 1D wykonywane na wierszach, a następnie kolumnach obrazu. W obliczeniach stosuje się szybkie algorytmy FFT. Zakres indeksów  $k_1$  i  $k_2$  w (13.19) ograniczony do liczb nieujemnych powoduje, że składowa stała  $X[0,0]$ , znajduje się w lewym, górnym rogu, co pokazano na rys.13.21. Korzystając z symetrii widma można zamienić ćwiartki płaszczyzny częstotliwości w sposób pokazany na rys.13.22 tym samym przesuwając składową stałą do środka płaszczyzny.

### 13.5 DWT obrazów

Podobnie jak w przypadku DFT, transformatę falkową obrazu wyznacza się za pomocą transformacji 1D zastosowanych do wierszy, a następnie kolumn obrazu.

Rys.13.23 przedstawia dwupoziomową dekompozycję falkową obrazu. Wiersze obrazu testowego o rozmiarze  $512 \times 512$  pikseli są dekomponowane na falkowe współczynniki aproksymacji i detali. Obliczenia mogą być przeprowadzane w decymowanym zespole filtrów z rys.11.16, lub w układzie predykcyjnym z rys.11.17. Następnie jednowymiarowej dekompozycji falkowej poddawane są aktualne kolumny obrazu. W wyniku otrzymujemy pierwszy poziom dekompozycji obrazu. Obraz jest reprezentowany przez falkowe współczynniki aproksymacji oraz współczynniki detali horyzontalnych (na prawo od aproksymacji), wertykalnych (poniżej aproksymacji) i diagonalnych. Współczynniki aproksymacji mogą być dalej dekomponowane w ten sam sposób.

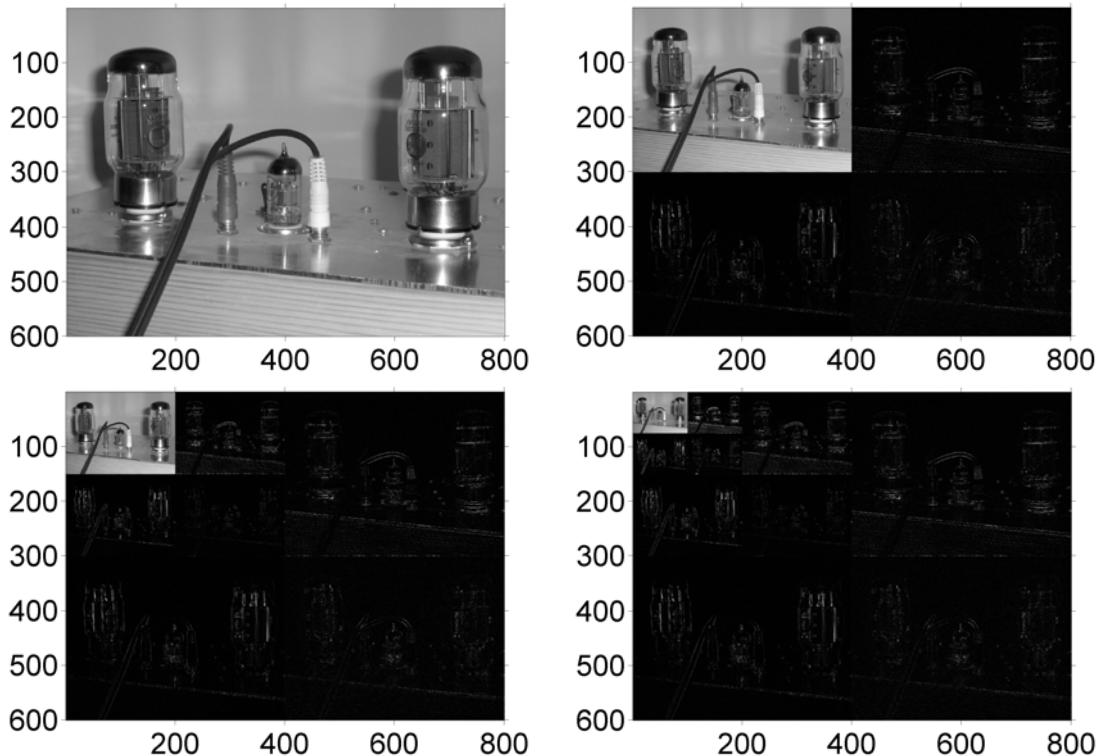


Rys.13.23 Dwupoziomowa dekompozycja falkowa obrazu.

Jak widać z rys.13.23 współczynników falkowych jest tyle samo co pikseli w obrazie.

Rys.13.24 przedstawia trzypoziomową dekompozycję falkową obrazu naturalnego. Na rysunkach, w celu lepszej widoczności, pokazane są moduły przeskalowanych detali. Współczynniki aproksymacji mają wartości w tym samym zakresie, co obraz testowy, tj. 0-255 w prezentowanym przykładzie, natomiast współczynniki detali mogą mieć wartości ujemne i dodatnie przekraczające zakres od -255 do 255. Transformacja falkowa ma bardzo

dobre własności dekorelacyjne, co oznacza, że większość energii we współczynnikach detali zawarta jest w małej liczbie tych współczynników. Pozostałe współczynniki detali mają wartości równe zeru lub bliskie zeru. Dzięki temu możliwa jest wydajna kompresja obrazów w dziedzinie falkowej. Dodatkowo algorytm dekompresji może być skalowalny, tzn. z początkowego strumienia bitów odtwarza się aproksymację sygnału, a następnie w miarę napływania kolejnych bitów aproksymację tę uzupełnia się szczegółami. Własność ta ma duże znaczenie np. w telemedycynie oraz w indeksowaniu multimedialnych baz danych. W przykładzie na rys.13.24 obraz testowy ma rozmiar  $600 \times 800$  pikseli, a jego aproksymacja na trzecim poziomie dekompozycji  $75 \times 100$  pikseli.



Rys.13.24 Trzypoziomowa dekompozycja falkowa obrazu.

### 13.6 Tomografia komputerowa

Tomografia komputerowa oznaczana jako CT (*Computed Tomography*) jest obecnie powszechnym narzędziem nieinwazyjnej diagnostyki medycznej. Znajduje on również szerokie zastosowanie w przemyśle. Słowo *tomography* oznacza przekrój, plaster, czyli CT oznacza obrazowanie na podstawie przekrojów.

Zadanie tomografii komputerowej polega na rekonstrukcji obrazu na podstawie projekcji (rzutowania) danych uzyskanych przez całkowanie wzdłuż różnych kierunków. Rys.13.25 przedstawia równoległe rzutowanie wiązki przez obiekt  $f(x,y)$  pod różnymi kątami  $\Theta$ . Zmierzona przez detektory funkcja  $P_\Theta(t)$  reprezentuje pochłanianie wiązki promieni rentgena przez obiekt:

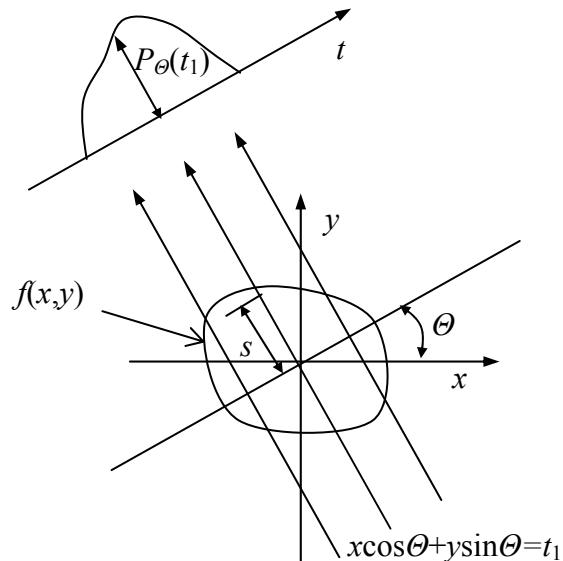
$$P_\Theta(t) = \int_{(\Theta,t)\text{line}} f(x,y) ds . \quad (13.21)$$

Stosując deltę Diraca i podstawienie  $x\cos\Theta + y\sin\Theta = t$  zależność (13.21) można przepisać w postaci:

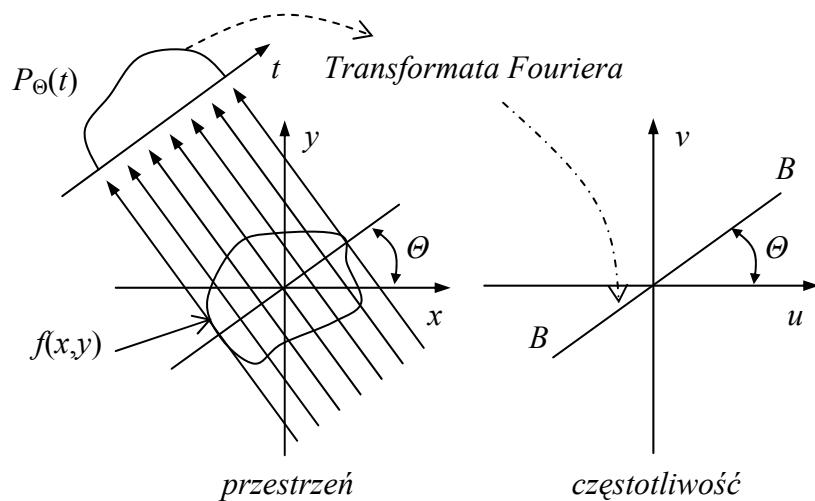
$$P_\Theta(t) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f(x,y) \delta(x \cos \Theta + y \sin \Theta - t) dx dy. \quad (13.22)$$

Funkcja  $P_\Theta(t)$  jest to transformata Radona funkcji  $f(x,y)$ . Tak, więc zadanie tomografii komputerowej polega na wyznaczeniu odwrotnej transformaty Radona, tj. rekonstrukcji obrazu  $f(x,y)$  na podstawie projekcji  $P_\Theta(t)$  pod różnymi kątami  $\Theta$ .

Odwrotna transformata Radona może być wyznaczona przy pomocy twierdzenia Fouriera o przekrojach (*Fourier Slice Theorem*). Twierdzenie to jest następujące. Transformata Fouriera projekcji równoległej obrazu  $f(x,y)$  pod kątem  $\Theta$  zwraca linię w dwuwymiarowym widmie obrazu  $F(u,v)$  pod kątem  $\Theta$  do osi  $u$ . Twierdzenie to jest zilustrowane na rys.13.26.



Rys.13.25 Równoległe rzutowanie wiązki (*parallel beam projection*).



Rys.13.26 Ilustracja twierdzenia Fouriera o przekrojach (*Fourier Slice Theorem*).

Twierdzenie Fouriera o przekrojach można zapisać w następującej postaci:

$$S_\Theta(w) = F(w, \Theta) = F(w \cos \Theta, w \sin \Theta), \quad (13.23)$$

gdzie:

$$S_\Theta(w) = \int_{-\infty}^{\infty} P_\Theta(t) e^{-j2\pi w t} dt, \quad S_\Theta(w) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f(x, y) e^{j2\pi w(x \cos \Theta + y \sin \Theta)} dx dy. \quad (13.24)$$

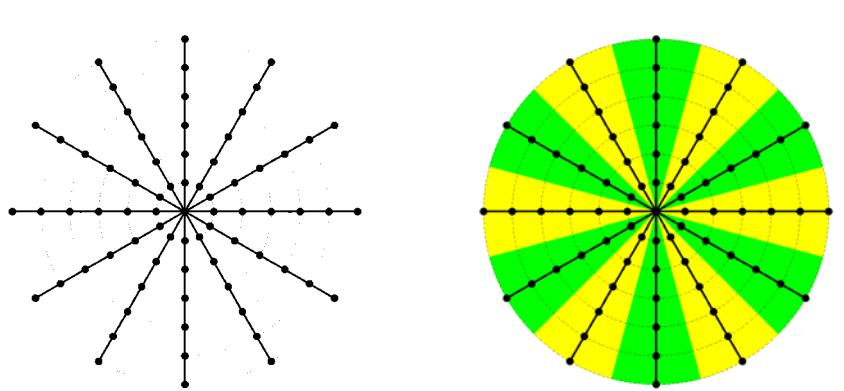
Na podstawie powyższego twierdzenia widmo obrazu  $F(u, v)$  może zostać zrekonstruowane z nieskończoną liczbą projekcji, a następnie obraz  $f(x, y)$  może zostać odtworzony przez odwrotną transformację Fouriera. W praktyce dostępna jest tylko skończona liczba projekcji, co powoduje próbkowanie płaszczyzny  $F(u, v)$  w sposób pokazany na rys.13.27. Wyznaczenie odwrotnej transformaty Fouriera z próbkiowanej płaszczyzny  $F(u, v)$  wymaga więc interpolacji wartości na siatce prostokątnej. Popularnym algorytmem stosowanym w tomografii jest filtrowana propagacja wsteczna (*filtered backprojection*). Rys.13.27 (prawy) przedstawia sposób interpolacji płaszczyzny  $F(u, v)$ , każda projekcja, zaznaczona na rys.13.27 jako linia, jest ważona (mnożona) przez funkcję, której wartości rosną wraz z częstotliwością, tak aby otrzymać zaznaczone obszary (kawałki torta) w sąsiedztwie każdej linii. Dla  $K$  projekcji w przedziale  $180^\circ$  dla danej częstotliwości  $w$ , współczynnik ważący wynosi  $2\pi|w|/K$ . Kształt charakterystyki częstotliwościowej filtra ważującego przedstawia rys.13.28.

Szukany obraz  $f(x, y)$  dany jest równaniem

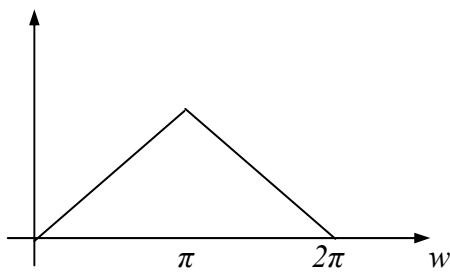
$$f(x, y) = \int_0^{\pi} \left[ \int_{-\infty}^{\infty} F(w, \Theta) |w| e^{j2\pi w t} dw \right] d\Theta. \quad (13.25)$$

W wewnętrznej całce występuje filtracja filtrem o odpowiedzi częstotliwościowej  $|w|$ . Po filtracji sygnał jest rzutowany wstecznie (*backprojected*), tzn. wynik każdej przefiltrowanej projekcji jest dodawany do wszystkich punktów  $f(x, y)$ , które leżą na drodze promienia.

Do testów algorytmów rekonstrukcji tomograficznej wykorzystywany jest syntetyczny model główny opracowany przez Sheppa i Logana pokazany na rys.13.29. Obraz ten jest zbudowany z 10 elips. Rys.13.30 i rys.13.31 przedstawiają transformatę Radona obrazu testowego dla projekcji wykonanych co 5 stopni i co 1 stopień oraz zrekonstruowane obrazy. Dla zbyt małej liczby projekcji w obrazie zrekonstruowanym pojawiają się wyraźne zakłócenia.



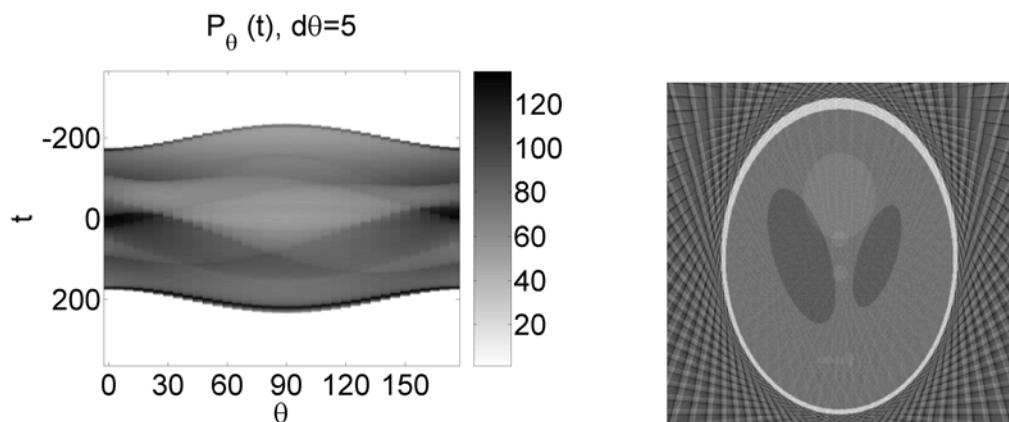
Rys.13.27 Próbkowanie płaszczyzny częstotliwości  $F(u, v)$  przez skończoną liczbę projekcji i interpolacja płaszczyzny częstotliwości.



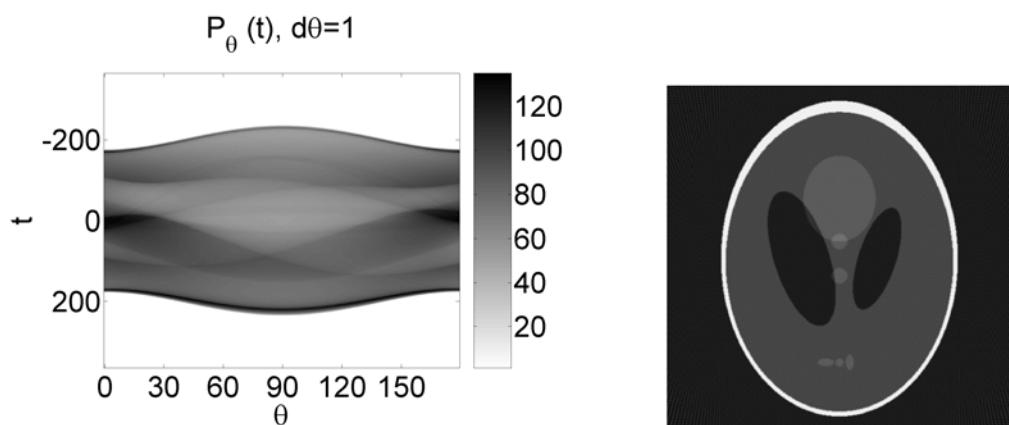
Rys.13.28 Charakterystyka częstotliwościowa filtra ważacego (interpolującego).



Rys.13.29 Model głowy Sheppa i Logana o rozmiarze  $512 \times 512$  pikseli.



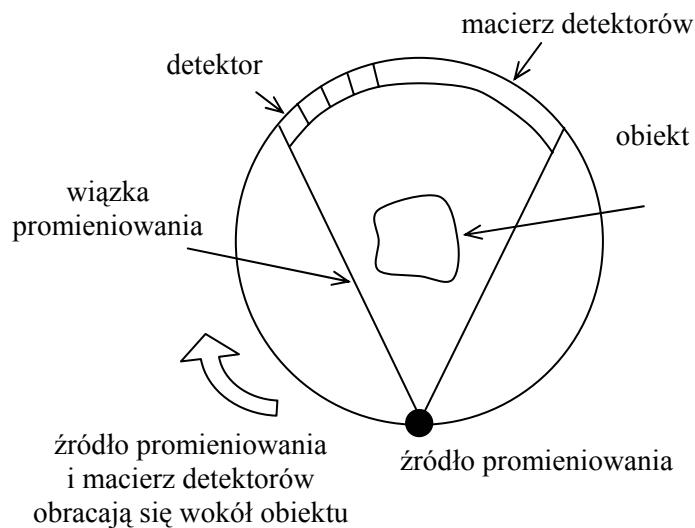
Rys.13.30 Transformata Radona modelu głowy Sheppa i Logana i rekonstrukcja obrazu, dla 36 projekcji wykonanych co 5 stopni.



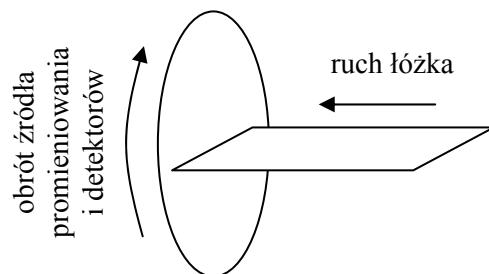
Rys.13.31 Transformata Radona modelu głowy Sheppa i Logana i rekonstrukcja obrazu, dla 180 projekcji wykonanych co 1 stopień.

Ze względu na łatwiejszą realizację techniczną w skanerach medycznych stosowana jest wiązka promieniowania uformowana w wachlarz (*fan-beam projection*), a nie równoległa, co pokazano na rys.13.32. Rys.13.32 przedstawia skaner CT trzeciej generacji, w którym źródło promieniowania i macierz detektorów obracają się wokół obiektu. Rys.13.33 przedstawia skaner CT czwartej generacji. Pacjent (obiekt) leży na łóżku, które przesuwa się względem wirującego źródła promieniowania i macierzy detektorów. Rekonstrukcja danych ze skanerów CT czwartej generacji jest obarczona rozmyciem spowodowanym ciągłym ruchem obiektu. Obiekt jest skanowany po spirali, a nie w równoległych płaszczyznach. Zaletą tej metody jest jednak możliwość skanowania dużych obszarów w krótkim czasie, np. klatki piersiowej na jednym wdechu pacjenta.

Głównym ograniczeniem rozdzielczości medycznych skanerów CT jest dopuszczalna dla pacjenta dawka promieniowania. Na jakość obrazów wpływa również stabilność (brak ruchu) pacjenta w czasie skanowania. Ruchy fizjologiczne takie jak oddychanie czy bicie serca powodują rozmycie zrekonstruowanego obrazu.



Rys.13.32 Skaner CT trzeciej generacji z wiązką promieniowania uformowaną w wachlarz (*fan beam projection*).



Rys.13.33 Spiralny skaner CT czwartej generacji (*helical (spiral) CT scanner*).



## Literatura

- [1]. Kak A. C., Slaney M., *Principles of Computerized Tomographic Imaging*, IEEE Press, 1988.
- [2]. Lim J.S., *Two-Dimensional Signal and Image Processing*, Englewood Cliffs, NJ, Prentice Hall, 1990.
- [3]. Moon T. K., Stirling W. C., *Mathematical Methods and Algorithms for Signal Processing*, Prentice Hall 1999.
- [4]. Oppenheim A. V., Schafer R. W., Buck J. R., *Discrete-Time Signal Processing*, 2nd Edition, Prentice-Hall, Signal Processing Series, 1999.
- [5]. Lyons R.G.: Wprowadzenie do cyfrowego przetwarzania sygnałów, WKŁ, Warszawa 1999.
- [6]. Drozdek A., *Wprowadzenie do kompresji danych*, WNT, Warszawa 1999.
- [7]. Zieliński T.P., *Cyfrowe przetwarzanie sygnałów, od teorii do zastosowań*, WKŁ, Warszawa, 2005.
- [8]. Matlab Signal Processing Toolbox, dostępne na [www.mathworks.com](http://www.mathworks.com).
- [9]. Matlab Filter Design Toolbox, dostępne na [www.mathworks.com](http://www.mathworks.com).
- [10]. Matlab Wavelet Toolbox, dostępne na [www.mathworks.com](http://www.mathworks.com).
- [11]. Matlab Image Processing Toolbox, dostępne na [www.mathworks.com](http://www.mathworks.com).