

Przetwarzanie i analiza sygnałów pomiarowych

CYFROWE PRZETWARZANIE SYGNAŁÓW W SYSTEMACH POMIAROWYCH

ANDRZEJ MAJKOWSKI, REMIGIUSZ RAK

ANALIZA CZĘSTOTLIWOŚCIOWA, ANALIZA CZASOWO-CZĘSTOTLIWOŚCIOWA, ANALIZA FALKOWA

Mniej lub bardziej zaawansowane algorytmy cyfrowego przetwarzania i analizy sygnałów stanowią nieodłączną część modułów programowych każdego wirtualnego przyrządu pomiarowego. W związku z tym większość narzędziowych programów przeznaczonych do projektowania wirtualnych przyrządów pomiarowych, wyposażona jest w biblioteki zawierające funkcje cyfrowego przetwarzania sygnałów. Odpowiadające im panele funkcyjne lub moduły graficzne mają formę przyjazną dla użytkownika i są łatwe w implementacji. Sztuczne problemy mogą wynikać z dowolności w zakresie terminologii i symboliki użytej do ich opisu. Rzeczywiste kłopoty mogą się pojawić dopiero na etapie interpretacji wyników, ponieważ nadawane im przez twórców formy graficzne obarczone są dużą dozą subiektywizmu. W celu uniknięcia kłopotów zarówno na etapie implementacji jak i interpretacji warto zapoznać się z praktycznymi aspektami najczęściej stosowanych metod jak również wykorzystywanych algorytmów numerycznych.

Spis treści

1.	Wstępne przetwarzanie sygnałów	2
1.1.	Usuwanie wartości średniej.....	2
1.2.	Usuwanie trendu i dryftu.....	3
1.3.	Filtracja cyfrowa.....	4
1.4.	Odszumianie sygnałów	7
2.	Analiza widmowa Fouriera	8
3.	Analiza czasowo-częstotliwościowa	16
3.1.	Krótkoczasowa transformata Fouriera	16
3.1.1.	Transformata Gabora	18
3.1.2.	Właściwości STFT.....	19
4.	Analiza falkowa.....	22
5.	Metody kompresji sygnałów	33
5.1.	Nierównomierna kwantyzacja skalarna.....	35
5.2.	Kwantyzacja wektorowa	36
5.3.	Kodowanie transformacyjne.....	38
5.4.	Alokacja bitów.....	40
5.5.	Kwantyzacja wektorowa współczynników transformat	41
6.	Ćwiczenia do modułu (rozwiązane problemy praktyczne - zadania, projekty).....	44
6.1.	Próbkowanie i kwantyzacja sygnału	44
6.2.	Analiza częstotliwościowa – szybka transformata Fouriera	45
6.3.	Analiza czasowo-częstotliwościowa – krótkoczasowa transformata Fouriera	49
6.4.	Badanie filtrów cyfrowych o skończonej i nieskończonej odpowiedzi impulsowej	53
6.5.	Transformata falkowa.....	60
7.	Pytania kontrolne	64
8.	Literatura	70

1. Wstępne przetwarzanie sygnałów

Naturalne, „surowe” sygnały pozyskane z obiektu pomiarowego często, przed poddaniem ich wnikliwej i szczegółowej analizie, wymagają tzw. przetwarzania wstępnego. To wstępne przetwarzanie może być realizowane zarówno w sposób analogowy, w blokach kondycjonowania sygnału, jak też cyfrowy w komputerze. Podstawowa funkcja, jaką spełniają układy kondycjonowania sygnałów to wzmacnianie. Do wzmacniania sygnałów pomiarowych o niskich poziomach stosuje się dobrej jakości wzmacniacze pomiarowe o programowanym wzmocnieniu (niskoszumne o małym napięciu niezrównoważenia i małych dryftach). Często funkcję wzmacniania sygnałów łączy się z izolacją galwaniczną. Domenę układów kondycjonowania sygnału stanowi, dolnopasmowa (antyaliasingowa) filtracja analogowa. Poza nią do najbardziej popularnych metod kondycjonowania sygnałów zalicza się: usuwanie wartości średniej, usuwanie trendu i dryftu oraz filtrację cyfrową i odszumianie¹.

1.1. Usuwanie wartości średniej

Sygnał - to jedno z podstawowych pojęć metrologii. Sygnałem nazywamy określony w funkcji czasu przebieg dowolnej wielkości fizycznej $x(t)$, np.: napięcia, prądu, natężenia pola elektrycznego, naprężeń przęsła mostu, prędkości obrotowej turbiny itp. Po zamianie dowolnego sygnału fizycznego (wielkości fizycznej) na sygnał elektryczny (wielkość elektryczną), mamy do czynienia z sygnałem elektrycznym.

Sygnały opisane analitycznie (zależność matematyczna) lub w inny równoważny sposób, np. graficznie, nazywają się sygnałami zdeterminowanymi, ponieważ ich wartości są określone z góry dla każdej chwili czasu. Najprostszymi sygnałami zdeterminowanymi są:

- sygnał harmoniczny:

$$x(t) = A \cos(\omega_0 t + \varphi_0), \quad -\infty < t < \infty \quad (1.1)$$

gdzie A jest amplitudą, ω_0 - pulsacją, a φ_0 - fazą początkową, oraz

- sygnał okresowy:

$$x(t) = x(t + mT), \quad -\infty < t < \infty \quad (1.2)$$

gdzie T jest okresem, a m - liczbą całkowitą dodatnią.

Sygnały, których wartości nie można z góry określić, nazywamy sygnałami niezdeterminowanymi (losowymi) i mówimy, że stanowią one realizacje procesów losowych. Na potrzeby analizy teoretycznej wprowadza się pojęcia procesów stacjonarnych oraz procesów ergodycznych. Parametry statystyczne procesów stacjonarnych są niezmiennie w czasie. W przypadku procesów ergodycznych parametry statystyczne są równoważne parametrom czasowym poszczególnych realizacji, np. wartość oczekiwana jest równa wartości średniej w czasie [21].

W przypadku użycia oscyloskopu do obserwacji sygnału, dokonujemy analizy czasowej, chociaż znając parametry podstawy czasu możemy określić wartość okresu, a więc również wartość częstotliwości powtarzania, dla przebiegów okresowych. Drugi znany sposób pomiaru częstotliwości polega na obserwacji

¹ Zdaniem autorów nie warto w tym miejscu wspominać o metodzie linearyzacji charakterystyk czujników pomiarowych, które obecnie realizuje się w trakcie przetwarzania (cyfrowego) sygnału pomiarowego.

krzywych Lissajou. W żadnym z wymienionych przypadków nie możemy jednak określić tzw. *zawartości harmonicznych* dla odkształconych (różnych od sinusoidy) przebiegów okresowych. W tym celu należy zastosować tzw. analizę widmową. Pojęcie „analiza widmowa” określa badanie właściwości sygnałów nie w dziedzinie czasowej lecz częstotliwościowej.

Typowy sygnał pomiarowy może zawierać wartość średnią, której estymator w przypadku sygnału ciągłego definiowany jest jako:

$$\bar{x}_T = \frac{1}{T} \int_0^T x(t) dt \quad (1.3)$$

W przypadku sygnału dyskretnego:

$$\bar{x}_N = \frac{1}{N} \sum_{n=0}^{N-1} x_n \quad (1.4)$$

W celu pozbycia się wartości średniej należy ją od sygnału odfiltrować. Można zastosować technikę sprzężenia zwrotnego zawierającego blok estymacji wartości średniej.

W przypadku, gdy pożądane jest przetwarzanie sygnału w czasie rzeczywistym można użyć jednej z poniżej przedstawionych metod filtracji:

- Sygnały ciągłe mogą być filtrowane za pomocą analogowego filtra dolnopasmowego, co prowadzi w efekcie do całkowania w sposób podobny do opisanego wzorem (1.3).
- Sygnały dyskretne mogą być filtrowane za pomocą cyfrowego filtra dolnopasmowego lub w sposób opisany formułą typu *moving average*.

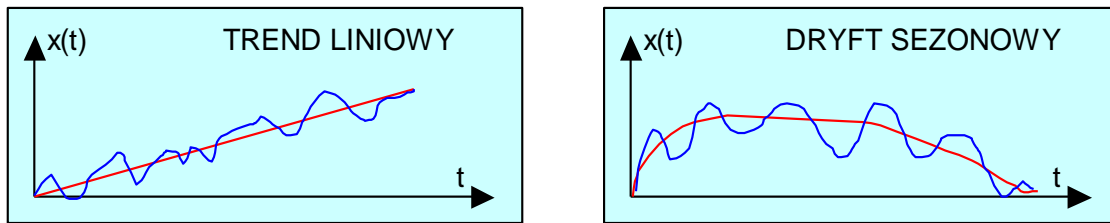
$$\bar{x}_N = \frac{1}{N} \sum_{n=0}^{N-1} x_n = \frac{1}{N} \left[\sum_{n=0}^{N-2} x_n + x_N \right] = \frac{1}{N} \left[(N-1) \bar{x}_{N-1} + x_N \right] \quad (1.5)$$

Trzeba pamiętać, że równanie (1.5) charakteryzuje powolna zbieżność, stąd początkowe estymaty wartości średniej obarczone są dużym błędem. Natomiast dla dużych wartości N, jest ono niewrażliwe na wszelkie niestacjonarne dryfty w zakresie wartości średniej. W związku z tym należy go używać ostrożnie. Warto zauważyć, że transmitancja tak powstałego filtra ma postać (1.6):

$$H(z) = \frac{b_0}{1 - a_1 z^{-1}}, \quad a_1 = \frac{N}{N-1}, \quad b_0 = \frac{1}{N} \quad (1.6)$$

1.2. Usuwanie trendu i dryftu

Istnieje cała gama niestacjonarnych trendów i dryftów, które mogą wystąpić w praktyce i zakłócić podstawowy nurt analizy. Można je z grubsza podzielić na dwie grupy: *trend liniowy* i *dryft sezonowy* (rys. 1.1). Nie istnieje jedna, uniwersalna metoda do usuwania tego typu trendów czy dryftów, ale można zaproponować pewne procedury znane i sprawdzone.



Rys. 1.1. Ilustracja trendu i dryftu sygnału pomiarowego

- Filtracja (analogowa lub cyfrowa): Sygnał jest przepuszczany przez filtr dolnopasmowy o odpowiedniej częstotliwości odcięcia, a następnie odejmowany od „surowego” sygnału. Ta metoda jest przydatna do usuwania trendu liniowego i niskoczęstotliwościowych dryftów sezonowych. Jest oczywiste, że nie można jej stosować w przypadku, gdy najniższa częstotliwość w widmie sygnału leży poniżej częstotliwości odcięcia filtru.
- Adaptacyjna filtracja dolnopasmowa (cyfrowa): Metoda podobna do omówionej poprzednio z tym, że tutaj parametry filtru cyfrowego są zmieniane w sposób adaptacyjny, stosownie do zmian długoterminowych właściwości sygnału (stanu źródła). Metoda przydatna w obydwu przypadkach.
- Filtracja pasmowa: Metoda najczęściej używana, lecz uwarunkowana wymaganiem, aby dolna częstotliwość odcięcia filtru pasmowego (a co za tym idzie dolnego pasma sygnału) znacznie przewyższała częstotliwość odcięcia dryftu lub trendu.
- Filtracja górnopasmowa: Metoda akceptowalna pod tym samym warunkiem, co poprzednia.

1.3. Filtracja cyfrowa

Pod pojęciem filtru cyfrowego rozumie się pewien algorytm, który podobnie jak filtr analogowy zdolny jest zmienić charakterystykę widmową sygnału (dyskretnego). Projektowanie filtrów cyfrowych stanowi sztukę samą w sobie. Czytelnik zainteresowany projektowaniem filtrów cyfrowych zmuszony będzie sięgnąć do literatury, po szczegóły. W niniejszym punkcie podany będzie tylko zarys metod filtracji cyfrowej. Symbole użyte w opisie przedstawione są na rysunku 1.2.



Rys. 1.2. Filtracja cyfrowa: $x(n)$ - ciąg wejściowy, $y(n)$ - ciąg wyjściowy

Działanie filtru cyfrowego w dziedzinie czasu opisuje równanie liniowe różnicowe (1.7).

$$y(n) + a_1 y(n-1) + \dots + a_M y(n-M) = b_0 x(n) + b_1 x(n-1) + \dots + b_{N-1} x(n-N+1) = 0 \quad (1.7)$$

Transmitancję filtru, przy założeniu zerowych warunków początkowych, opisuje się w dziedzinie zmiennej zespolonej (1.8):

$$H(z) = \frac{Y(z)}{X(z)} \quad (1.8)$$

Po konfrontacji z równaniem różnicowym liniowym (1.7) przyjmuje ona postać:

$$H(z) = \frac{b_0 + b_1 z^{-1} + b_2 z^{-2} + \dots + b_{N-1} z^{-N+1}}{1 + a_1 z^{-1} + a_2 z^{-2} + \dots + a_M z^{-M}} \quad (1.9)$$

W przypadku szczególnym, gdy ciąg wejściowy $x(n)$ ma postać:

$$x(n) = \delta(n) = \begin{cases} 1, & \text{dla } n = 0 \\ 0 & \text{dla } n \neq 0 \end{cases} \quad (1.10)$$

ciąg wyjściowy zwany jest *odpowiedzią impulsową* filtru i równoważny jest odwrotnej transformacji Z transmitancji filtru:

$$y(n) = h(n) = Z^{-1}[H(z)] \quad (1.11)$$

Dla liniowego układu przyczynowego, tzn. gdy $h(n)=0$, dla $n<0$:

$$Y(z) = H(z)X(z) \quad (1.12)$$

Zatem:

$$y(n) = x(n) * h(n) = \sum_{m=0}^n x(n)h(n-m) \quad (1.13)$$

Warunkiem koniecznym i wystarczającym stabilności filtru jest, aby:

$$\sum_{n=0}^{\infty} |h(n)| < \infty \quad (1.14)$$

co pociąga za sobą konieczność przynależności biegunów transmitancji $H(z)$ do koła jednostkowego $|z|<1$.

Struktury filtrów cyfrowych można podzielić na dwie podstawowe grupy:

1. Filtry o skończonej odpowiedzi impulsowej: SOI (Finite Impulse Response: FIR),

dla których:

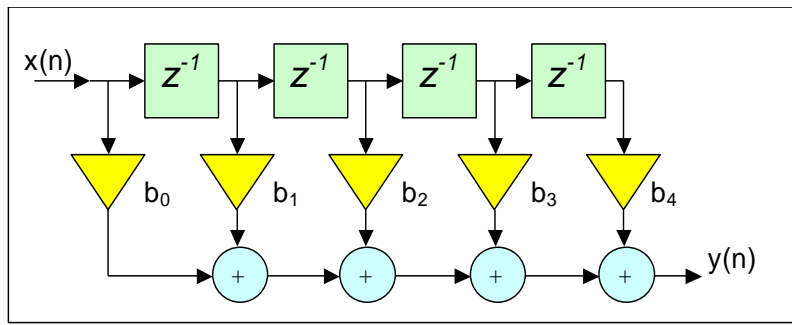
$$a_1 = a_2 = \dots = a_M = 0 \quad (1.15)$$

a co za tym idzie:

$$h(n) = \begin{cases} b_n, & n = 0, 1, 2, \dots, N-1 \\ 0, & \text{dla pozostałych} \end{cases} \quad (1.16)$$

Na rys.1.3 przedstawiono przykładową strukturę filtru SOI o transmitancji:

$$H(z) = b_0 + b_1 z^{-1} + b_2 z^{-2} + b_3 z^{-3} + b_4 z^{-4} \quad (1.17)$$

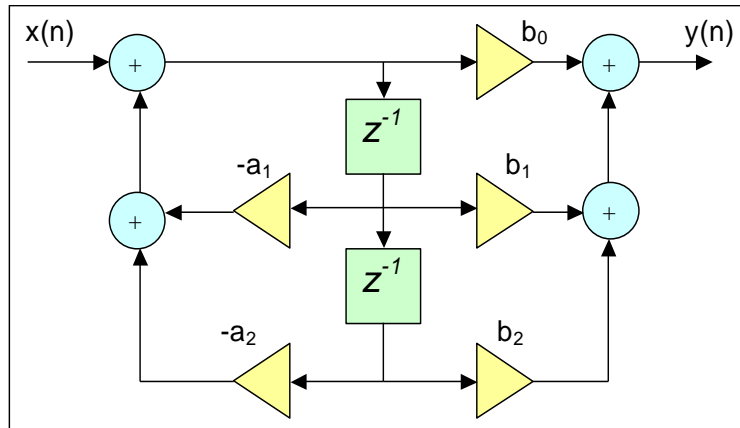


Rys. 1.3. Przykładowa struktura filtru SOI

2. Filtry o nieskończonej odpowiedzi impulsowej: NOI (Infinite Impulse Response: IIR), dla których wielomian w mianowniku (1.9) jest różny od jedności.

W formie przykładu, na rys.1.4 przedstawiona została struktura kanoniczna tzw. *sekcji bikwadratowej* NOI o transmitancji:

$$H(z) = \frac{b_0 + b_1z^{-1} + b_2z^{-2}}{1 + a_1z^{-1} + a_2z^{-2}} \quad (1.18)$$



Rys. 1.4. Struktura kanoniczna sekcji bikwadratowej NOI

Charakterystyki częstotliwościowe filtru otrzymuje się zakładając poruszanie się po okręgu jednostkowym $z=e^{j\theta}$:

$$H(\theta) = \left. \frac{Y(z)}{X(z)} \right|_{z=e^{j\theta}} \quad (1.19)$$

gdzie symbol θ oznacza pulsację unormowaną.

Projektowanie filtru cyfrowego zasadniczo sprowadza się do dwu metod:

1. Przekształcenie transmitancji filtru analogowego $G(s)$ do postaci dyskretniej

$$H(z) = Z[H(s)] \quad (1.20)$$

2. Bezpośrednie zaprojektowanie filtru cyfrowego $G(z)$ z odpowiednią charakterystyką częstotliwościową.

Jako przykłady niech posłużą następujące filtry cyfrowe:

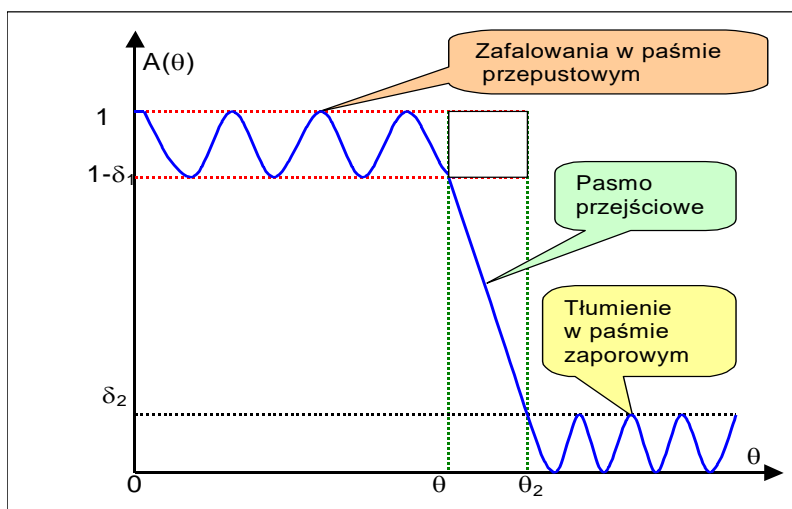
- Filtr dolnopasmowy: $y_n = \frac{1}{2}(x_n + x_{n-1}) \Rightarrow H(z) = \frac{1}{2}(1 + z^{-1})$
- Filtr górnopasmowy: $y_n = (x_n - x_{n-1}) \Rightarrow H(z) = (1 - z^{-1})$

- Filtr środkowopasmowy: $y_n = \frac{1}{2}(x_n - x_{n-2}) \Rightarrow H(z) = \frac{1}{2}(1 - z^{-2})$

Tak opisane filtry (niskiego rzędu) mają oczywiście nienajlepsze charakterystyki częstotliwościowe, szczególnie w sąsiedztwie częstotliwości odcięcia i mogą być traktowane wyłącznie w charakterze przykładów akademickich.

Projektowanie filtrów cyfrowych jest zadaniem bardzo ambitnym. Na szczęście dostępne są bardzo efektywne narzędzia do projektowania filtrów, o charakterze inżynierskim. Należy do nich przede wszystkim pakiet programowy MATLAB.

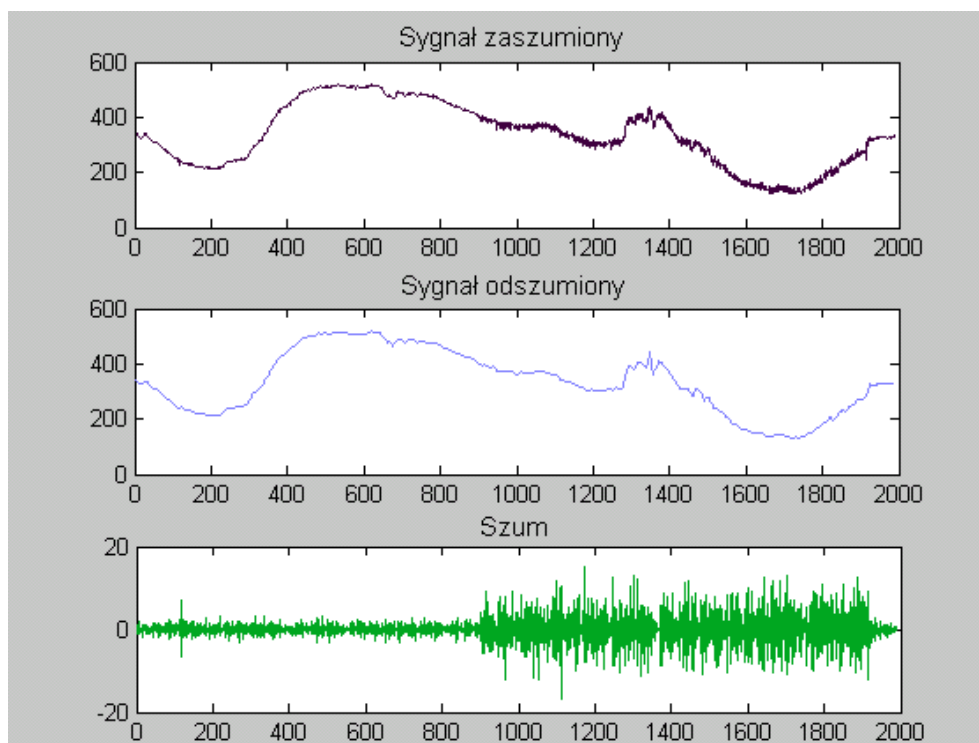
Projektowanie filtrów w tym środowisku polega na umiejscowieniu charakterystyki w unormowanym układzie współrzędnych (rys.1.5), a następnie, po wywołaniu stosownych funkcji, wyznaczenie najpierw rzędu filtru (parametry N i M), a następnie składowych wektorów współczynników \mathbf{a} i \mathbf{b} . Podobne narzędzia dostępne są w zintegrowanych pakietach oprogramowania narzędziowego do projektowania wirtualnych przyrządów pomiarowych.



Rys. 1.5. Charakterystyka amplitudowa filtra w unormowanym układzie współrzędnych

1.4. Odszumianie sygnałów

Właściwie należałoby w tym miejscu również powiedzieć, że odszumianie sygnałów stanowi sztukę samą w sobie. Stosuje się bowiem w tym zakresie bardzo wyrafinowane techniki, z których najbardziej popularne polegają na zastosowaniu *transformaty falkowej* oraz *transformaty składowych głównych*. W każdym przypadku, najpierw realizuje się przekształcenie proste, potem eliminację selektywnie wybranych współczynników i wreszcie przekształcenie odwrotne. Przykład odszumiania z zastosowaniem metody falkowej zamieszczono na rys.1.6. Widać tam sygnał oryginalny (zszumiony), sygnał odszumiony oraz wyeliminowany sygnał szumowy.



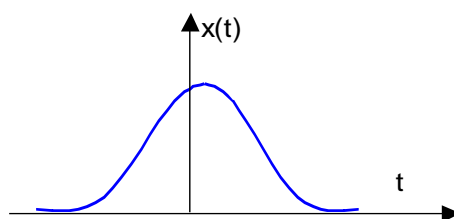
Rys. 1.6. Przykładowe zobrazowanie procesu odzuszumiania

2. Analiza widmowa Fouriera

Analiza widmowa Fouriera sygnałów jako superpozycja funkcji *sinus* i *cosinus* jest obecnie niemal wszechobecna w dziedzinie identyfikacji i analizy sygnałów pomiarowych. Użyteczność transformaty Fouriera zawiera się w jej zdolności do analizy przebiegu czasowego sygnału pod kątem jego „zawartości częstotliwościowej”. Zastosowana w tym celu transformacja sygnału polega na przetworzeniu funkcji opisanej w dziedzinie czasu na funkcję opisaną w dziedzinie częstotliwości. Dopiero wtedy, sygnał może być analizowany pod kątem jego właściwości częstotliwościowych. W przypadku transformaty dyskretnej, współczynniki transformaty reprezentują udział każdej ze składowych przebiegów typu *sinus* i *cosinus* w funkcji ich częstotliwości.

Na potrzeby analizy widmowej niezbędne jest wprowadzenie klasyfikacji sygnałów na sygnały o ograniczonej energii (rys.2.1):

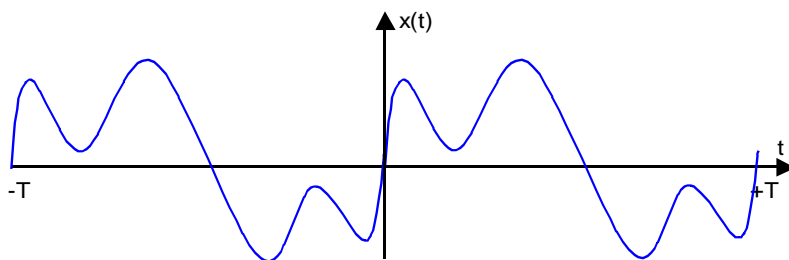
$$\int_{-\infty}^{+\infty} x^2(t) dt < \infty, \quad (2.1)$$



Rys. 2.1. Przykład sygnału o ograniczonej energii

oraz sygnały okresowe o ograniczonej mocy średniej (rys.2.2):

$$\int_{t_0}^{t_0+T} x_p^2(t) dt < \infty \quad (2.2)$$



Rys. 2.2. Fragment sygnału okresowego o ograniczonej mocy średniej

Dla ciągłego sygnału analogowego $x(t)$ o ograniczonej energii definiuje się pojęcie widma $X(\omega)^2$ za pomocą tzw. *ciągłej transformaty Fouriera*:

$$X(\omega) = \int_{-\infty}^{+\infty} x(t)e^{-j\omega t} dt, \quad (2.3)$$

określanej też mianem *transformaty całkowej*. Na odtworzenie sygnału z jego widma pozwala transformata odwrotna:

$$x(t) = \frac{1}{2\pi} \int_{-\infty}^{+\infty} X(\omega)e^{j\omega t} d\omega. \quad (2.4)$$

W obydwu wzorach $\omega = 2\pi f = 2\pi/T$ oznacza pulsację.

W przypadku sygnału okresowego $x_p(t)$ (periodycznego, stąd symbol p) wyznaczenie widma jest bardziej proste i polega na rozwinięciu jednego jego okresu w zespolony szereg Fouriera:

$$x_p(t) = \sum_{k=-\infty}^{+\infty} X_{pk} e^{jk\omega_0 t} \quad (2.5)$$

gdzie współczynniki tego rozwinięcia X_{pk} stanowią dyskretne widmo sygnału, i można je wyliczyć za pomocą całki Fouriera zastosowanej do pojedynczego okresu:

$$X_{pk} = \frac{1}{T} \int_{t_0}^{t_0+T} x_p(t) e^{-jk\omega_0 t} dt \quad (2.6).$$

W obydwu wzorach $\omega_0 = 2\pi f_0 = 2\pi/T$ stanowi pulsację sygnału okresowego.

W ogólności składowe widma są liczbami zespolonymi, a więc można im przyporządkować pewną amplitudę i fazę. Oczywiście można je również zapisać w postaci trygonometrycznej. Na rys.2.3 zamieszczono zestaw przebiegów wyjaśniających zasadę syntezy fali prostokątnej, z przebiegów harmonicznymi o zerowej fazie

² Podana definicja dotyczy widma amplitudowego. Do analizy częstotliwościowej sygnałów losowych często stosuje się funkcję widmowej gęstości mocy. Jest ona definiowana jako transformata Fouriera funkcji autokorelacji [22].

początkowej. Jest to proces odwrotny w stosunku do tego, co obserwujemy w trakcie analizy widmowej. Do syntezy wykorzystano, w sposób stopniowy następujące sygnały:

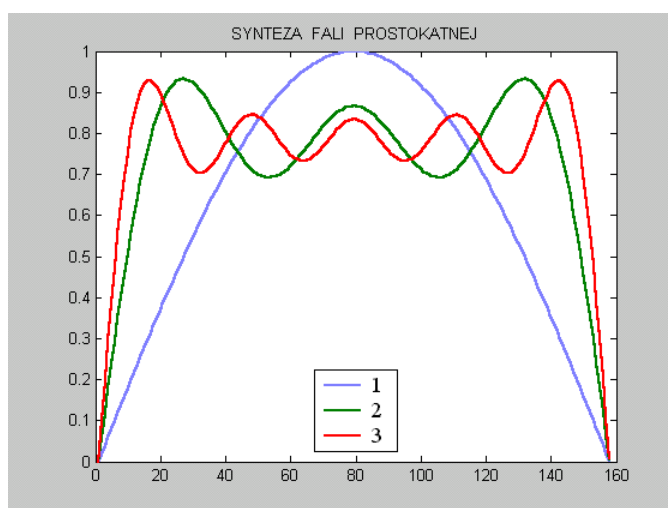
$$x_1(t) = \sin(\omega_0 t) \quad (2.7)$$

$$x_3(t) = \sin(\omega_0 t) + \frac{1}{3} \sin(3\omega_0 t) \quad (2.8)$$

$$x_5(t) = \sin(\omega_0 t) + \frac{1}{3} \sin(3\omega_0 t) + \frac{1}{5} \sin(5\omega_0 t) \quad (2.9)$$

$$x_7(t) = \sin(\omega_0 t) + \frac{1}{3} \sin(3\omega_0 t) + \frac{1}{5} \sin(5\omega_0 t) + \frac{1}{7} \sin(7\omega_0 t) \quad (2.10)$$

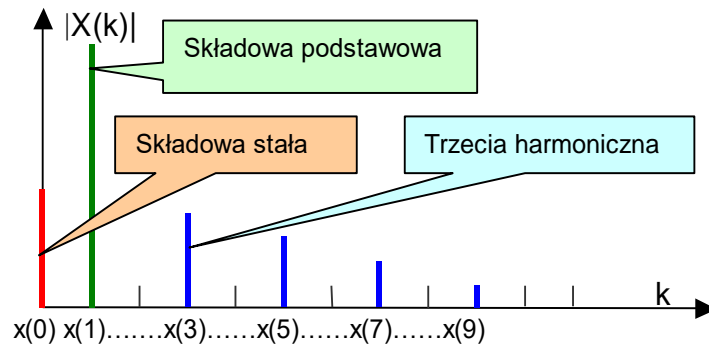
Jak widać sygnały te otrzymano drogą sumowania nieparzystych harmonicznych o amplitudach proporcjonalnych do $1/n$, gdzie n oznacza numer harmonicznej. Na rysunku 2.3 widać wyraźnie jak liczba składowych (nieparzystych harmonicznych) wpływa na kształt przebiegu wynikowego.



Rys. 2.3. Przykład kolejnego sumowania nieparzystych harmonicznych: 1 – $x_1(t)$, 2 – $x_3(t)$, 3 – $x_5(t)$

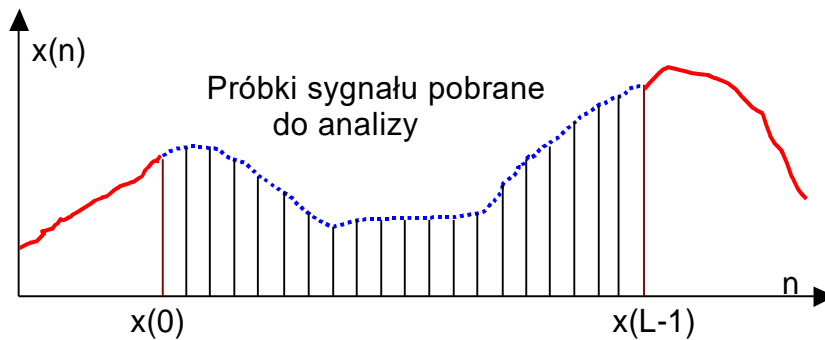
Poszczególnym fragmentom widma, tak zdefiniowanego sygnału, przypisuje się pewne szczególne nazwy i znaczenie zgodne z interpretacją zjawisk fizycznych (rys.2.4). Prążek znajdujący się na pozycji zerowej, *prążek zerowy*, jest określany mianem *składowej stałej* przebiegu. Prążek z nim sąsiadujący nosi nazwę *podstawowej harmonicznej*, zaś wszystkie pozostałe określane są mianem *wyższych harmonicznych*.³

³ W ogólnym przypadku składowe widma nie muszą występować w związku harmonicznym między sobą.



Rys. 2.4. Przykładowe widmo sygnału

Wyznaczanie ciągłego widma sygnału (całkowa transformata Fouriera), czyli analiza widmowa, wymaga stosunkowo skomplikowanych obliczeń matematycznych [22]. Na szczęście, po przejściu do przypadku dyskretnego, obliczenia te mogą być wykonywane za pomocą komputera. Przedtem jednak należy wybrany fragment realizacji sygnału wczytać do pamięci komputera. Wymaga to uprzedniego przetworzenia tego sygnału do postaci cyfrowej, a więc próbkowania i kwantyzacji. W praktyce pomiarowej, gdy mamy do czynienia z sygnałami niezdeterminowanymi, napotykamy pewne trudności, których podstawową przyczyną jest właśnie tzw. *skończony czas obserwacji sygnału*. Do analizy widmowej, z konieczności, przeznaczony zostanie tylko pewien jego fragment - np. część zaznaczona linią przerywaną na rys. 2.5, zawierająca L próbek.



Rys. 2.5. Obraz fragmentu sygnału przeznaczonego do analizy

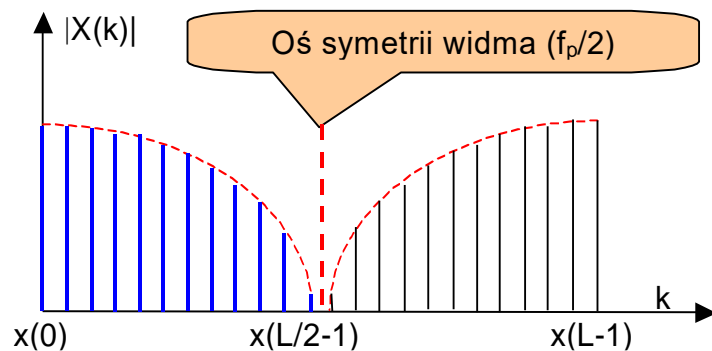
Mówimy w takim przypadku, że na sygnał *nałożone zostało okno czasowe* o kształcie prostokątnym, zgodnie z zależnością (2.11).

$$x_w(n) = x(n)w(n) \quad (2.11)$$

Wyznaczenie widma polega, w takim przypadku, na zastosowaniu algorytmu Dyskretnej Transformaty Fouriera (Discrete Fourier Transform: DFT). Każdy prążek widma wyznacza się z zależności:

$$X(k) = \sum_{n=0}^{L-1} x(n)e^{-j2\pi nk/L} \quad (2.12)$$

Przykładowy obraz dyskretnego widma sygnału rzeczywistego, uzyskany w jednym ze wspomnianych już środowisk inżynierskich (LabWinows/CVI, MATLAB) przedstawiono na rys. 2.6.



Rys. 2.6. Przykładowy obraz dyskretnego widma sygnału rzeczywistego

Zastosowanie algorytmu odwrotnej Dyskretnej Transformaty Fouriera (Inverse Discrete Fourier Transform: IDFT), umożliwi odtworzenie ciągu próbek sygnału (2.13).

$$x(n) = \frac{1}{L} \sum_{k=0}^{L-1} X(k) e^{j2\pi nk/L} \quad (2.13)$$

Przy czym, ponieważ idea Dyskretnej Transformaty Fouriera wymaga „spróbkowania widma”, sygnał odtworzony przyjmuje formę okresową, tzn. następuje powielenie na osi czasu fragmentu sygnału przyjętego do analizy (rys. 2.7).



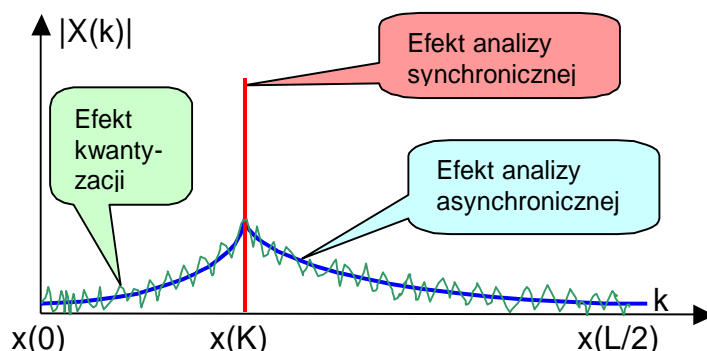
Rys. 2.7. Przykład sygnału odtworzonego z dyskretnego widma

Z rysunku wynika, że przypadkowość polegająca na „wycięciu z kontekstu” fragmentu sygnału (do analizy) objawiać się może w znaczącym odkształceniu przebiegu sygnału na krańcach przedziału po jego odtworzeniu. Znajduje to swoje odzwierciedlenie w widmie sygnału. Skutecznym sposobem ograniczenia niekorzystnego wpływu tego efektu na przebieg analizy jest uprzednie „odkształcenie” tego wycinka sygnału przez zastosowanie okna czasowego o kształcie odmiennym od prostokątnego. Znanych i stosowanych jest wiele różnych kształtów okien czasowych. Zwykle, kształt takiego okna w sposób łagodny tłumaczy do zera amplitudy próbek sygnału na krańcach przedziału. Nie ma recepty na dobór kształtu okna czasowego. Są jednak kryteria pozwalające ocenić właściwości okna w dziedzinie częstotliwości i dobrać okno stosownie do potrzeb. Często, choć nie jest to sposób zalecany, dobiera się je w sposób eksperymentalny, obserwując kształt widma.

W szczególnym przypadku, gdy mamy do czynienia z dyskretnym sygnałem okresowym o okresie N ($L=N$), do analizy możemy przeznaczyć wycinek będący wielokrotnością okresu. Wtedy mamy do czynienia z przypadkiem synchronicznej analizy widmowej, nie wymagającej użycia okien czasowych o wymyślnych kształtach.

Podsumowanie powyższych rozważań stanowi treść rysunku 2.8, na którym zamieszczono przykład analizy widmowej sygnału harmonicznego o częstotliwości $f_h = Kf_p/L$. Jak z niego wynika, w przypadku ogólnym, na

całościowy obraz wyników analizy widmowej składają się trzy wskazane tam elementy: wynik analizy synchronicznej, efekt asynchroniczności, efekt kwantyzacji próbek. Na rysunku, dla przejrzystości, zamieszczono tylko obwiednie fragmentów widma leżących po obydwu stronach prążka głównego.



Rys. 2.8. Wpływ poszczególnych efektów na wyniki analizy widmowej

Obliczenia widma sygnału dokonuje się przy użyciu specjalnego programu do wyznaczania transformaty Fouriera. Program taki można napisać wykorzystując algorytm numeryczny, który powszechnie jest znany pod nazwą Szybkiej Transformaty Fouriera (STF). Częściej używana jest jego nazwa angielska: Fast Fourier Transform (FFT). Algorytm ten został tak skonstruowany, aby obliczenia prowadzone były jak najszybciej, a przetwarzanie sygnału mogło odbywać się *na bieżąco* (czyli w czasie rzeczywistym).

Podstawowy parametr analizy widmowej to *rozdzielczość widma*: Δf . Jest ona równa ilorazowi zakresu częstotliwościowego, (równoważnego f_p) i liczby prążków widma L (rozmiar transformaty) (2.14):

$$\Delta f = \frac{f_p}{L} \quad (2.14)$$

Ta zależność umożliwia skuteczne wyskalowanie osi częstotliwości (zastąpienie indeksów próbek widma wartościami częstotliwości) – co nierzadko, z definicji leży w gestii użytkownika jednego z wymienionych wcześniej programów (i nie zawsze jest dla niego przyjazne)⁴. Dla wygody obliczeń, a ściślej, możliwości wykorzystania algorytmu FFT, przyjmuje się $L=2^n$, np. 128, 256, 512, 1024 itd.

W praktyce bardzo często do zobrazowania widma stosuje się charakterystykę logarymiczną w zakresie amplitud. Dotyczy to przypadków, kiedy w widmie sygnału występują ważne składowe, lecz o bardzo małych amplitudach. Dodatkowo, okazuje się, że duże znaczenie praktyczne ma funkcja (przebieg), która powstaje po wyznaczeniu odwrotnej transformaty Fouriera z logarymicznego widma sygnału. Określa się ją mianem *cepstrum*. Nazwa ta powstała drogą zmiany porządku grup liter w słowie *spectrum*, które oznacza *widmo* w języku angielskim. Jak łatwo zauważyć jest to funkcja czasu lecz okazuje się, że jej przebieg znacznie odbiega od kształtu segmentu sygnału wybranego do analizy i niesie szereg cennych informacji o sygnale.

Celem przygotowania się do dyskusji na temat czasowo-częstotliwościowych aspektów analizy warto przypomnieć, że w zapisie z użyciem iloczynu skalarnego transformaty Fouriera przyjmuje postać:

⁴ Pełny zakres widma (L) równoważny jest wartości częstotliwości próbkowania f_p . Oś symetrii widma leży w połowie tego zakresu – $f_p/2$.

$$X(\omega) = \int_{-\infty}^{+\infty} x(t)e^{-j\omega t} dt = \int_{-\infty}^{+\infty} x(t)e^{j\omega t} dt = \langle x(t), e^{j\omega t} \rangle \quad (2.15)$$

Właściwości transformaty Fouriera weryfikuje się w obszarze takich zagadnień jak:

- Liniowość:

$$\begin{cases} x(t) = ax_1(t) + bx_2(t) \\ X(\omega) = aX_1(\omega) + bX_2(\omega) \end{cases} \quad (2.16)$$

- Przesunięcie i skalowanie w czasie:

$$x_0(t) := x(t - t_0) \longrightarrow X_0(\omega) = X(\omega)e^{-j\omega t_0} \quad (2.17)$$

$$x_a(t) := x(at) \longrightarrow X_a(\omega) = \frac{1}{|a|} X\left(\frac{\omega}{a}\right) \quad (2.18)$$

- Przesunięcie i skalowanie w częstotliwości

$$X_0(\omega) := X(\omega - \omega_0) \Rightarrow x_0(t) = x(t)e^{-j\omega_0 t} \quad (2.19)$$

$$X_a(\omega) := X(a\omega) \Rightarrow x_a(t) = \frac{1}{|a|} x\left(\frac{t}{a}\right) \quad (2.20)$$

- Momenty:

$$M_n := \int_{-\infty}^{+\infty} t^n x(t) dt \Rightarrow M_n = (-j)^{-n} \frac{d^n X(\omega)}{d\omega^n} \Big|_{\omega=0} \quad (2.21)$$

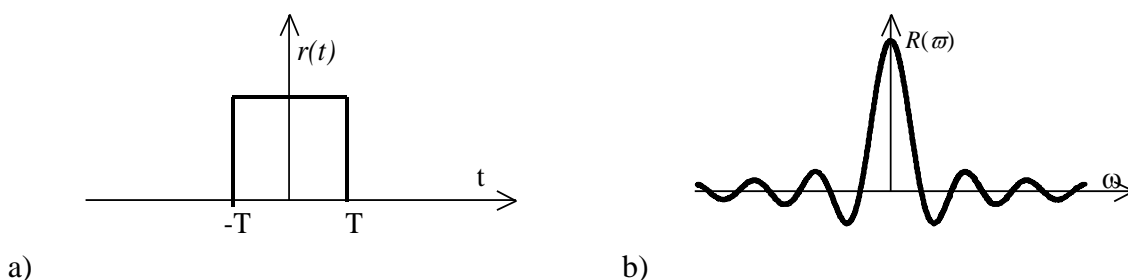
- Splot:

$$x(t) = \int_{-\infty}^{+\infty} x_1(\tau)x_2(t - \tau) d\tau \quad (2.22)$$

$$x(t) = x_1(t) * x_2(t) \Rightarrow X(\omega) = X_1(\omega)X_2(\omega) \quad (2.23)$$

W celu poparcia rozważań teoretycznych dotyczących wyznaczania widma sygnałów o ograniczonej energii, poniżej podano wybrane przykłady, które znajdują zastosowanie w analizie czasowo-częstotliwościowej.

Przykład 1 impuls prostokątny : $r(t) = u(t+T) - u(t-T)$



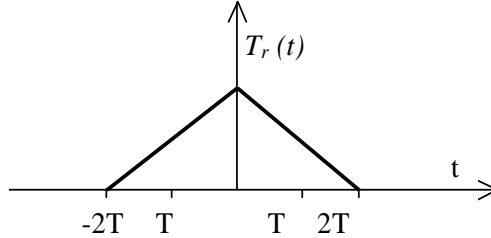
Rys. 2.9. Impuls prostokątny a) i jego widmo b)

$$R(\omega) = \frac{2T \sin(\omega T)}{\omega T} \quad (2.24)$$

Dla $T=0.5$, otrzymuje się funkcję charakterystyczną określaną mianem *B-spline* rzędu I-go

$$r(t) = r(t - 0.5) = \chi_{[0,1)}(t)$$

Przykład 2: Impuls trójkątny: $T_r(t) = r(t) * r(t)$



Rys. 2.10. Impuls trójkątny

Widmo tego impulsu:

$$T_r(\omega) = 4T^2 \frac{\sin^2(\omega T)}{(\omega T)^2} \quad (2.25)$$

W tym przypadku dla $T=0.5$, otrzymuje tzw. *B-spline* rzędu II-go. Łatwo zauważyć, że kolejne funkcje typu *spline* otrzymuje się drogą realizacji splotu w dziedzinie czasu. Ich widma wyznacza się drogą iloczynu widm, a ściślej biorąc podnoszenia widma do kwadratu. Dla przypadku rzędu II-go mamy:

$$t_r(t) = r(t) * r(t), \quad \Rightarrow \quad T_r(\omega) = R(\omega)R(\omega) = [R(\omega)]^2 \quad (2.26)$$

Przykład 3: Impuls Gaussa:

$$g(t) = e^{-\alpha t^2} \quad (2.27)$$

$$\begin{aligned} G(\omega) &= \int_{-\infty}^{+\infty} e^{-\alpha t^2} \cdot e^{-j\omega t} dt = \int_{-\infty}^{+\infty} e^{-\alpha \left[t^2 + j \left(\frac{\omega}{\alpha} \right) t \right]} dt = \\ &= \int_{-\infty}^{+\infty} e^{-\alpha \left[t^2 + j \left(\frac{\omega}{\alpha} \right) t - \left(\frac{\omega^2}{4\alpha^2} \right) \right] - \left(\frac{\omega^2}{4\alpha} \right)} dt = \\ &= e^{-\left(\frac{\omega^2}{4\alpha} \right)} \cdot \int_{-\infty}^{+\infty} e^{-\alpha \left[t + j \left(\frac{\omega}{2\alpha} \right) \right]^2} dt = \sqrt{\frac{\pi}{\alpha}} \cdot e^{-\left(\frac{\omega^2}{4\alpha} \right)} \end{aligned} \quad (2.28)$$



Rys. 2.11. Impuls Gaussa i jego widmo

3. Analiza czasowo-częstotliwościowa

Należy podkreślić, że tradycyjna analiza częstotliwościowa nie nadaje się do obserwacji właściwości sygnałów niestacjonarnych, o parametrach zmiennych w czasie. Wymagana jest tutaj analiza wykorzystująca łączne czasowo-częstotliwościowe reprezentacje sygnałów. Rodzina reprezentacji czasowo-częstotliwościowych jest bardzo duża. Ogólnie mogą być one podzielone na reprezentacje typu czas-częstotliwość oraz czas-skala, a interpretowane jako rodzaj tzw. *krótkoczasowej analizy widmowej* [6, 15,16].

3.1. Krótkoczasowa transformata Fouriera

Jeżeli mamy do czynienia z sygnałem nieokresowym, sumowanie funkcji okresowych typu *sinus* i *cosinus*, nie może dać wiarygodnej reprezentacji sygnału. Sygnał ten można w sposób sztuczny doprowadzić do okresowości, ale wymaga to zadbania o zachowanie ciągłości na krańcach przedziałów. Tak zwana *Krótkoczasowa Transformata Fouriera* (Short Time Fourier Transform – STFT) jest skutecznym narzędziem rozwiązania problemu budowania reprezentacji widmowej funkcji nieokresowej.

Krótkoczasowa Transformata Fouriera umożliwia wydobycie z sygnału informacji o tym, jak zmienia się jego widmo w czasie, czyli jednoczesną obserwację jego właściwości zarówno w dziedzinie czasu jak i częstotliwości. Wycinek sygnału (blok próbek o rozmiarze L) przeznaczony do analizy zostaje podzielony na segmenty, a każdy segment podlega analizie widmowej niezależnie. Podobnie jak w przypadku tradycyjnym, aby usunąć gwałtowne zmiany (cięcia) sygnału na krańcach przedziałów, stosuje się wymyślne okna czasowe w odniesieniu do wspomnianych sekcji. Sukcesywne „przesuwanie okna czasowego” umożliwia lokalizację parametrów widmowych sygnału w czasie.

Ten proces daje się zapisać za pomocą prostego równania:

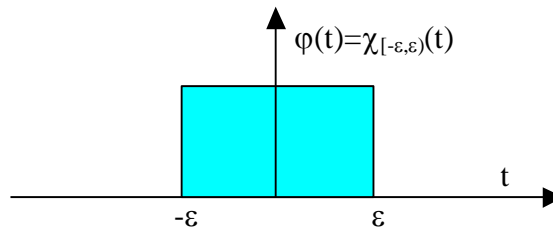
$$X(t, \omega) = \int_{-\infty}^{\infty} x(t) \varphi(t - \tau) e^{-j\omega\tau} d\tau \quad (3.1)$$

w którym $\varphi()$ opisuje funkcję okna czasowego. Przesuwając okno $\varphi()$ w czasie, wzdłuż sygnału, wyznacza się jego zawartość widmową wewnątrz przedziału czasowego, którego długość jest określona szerokością okna. Kształt okna czasowego odgrywa kluczową rolę w przypadku STFT. Iloczyn szerokości okna w dziedzinie czasu i szerokości okna w dziedzinie częstotliwości jest wielkością stałą dla danego okna. Stąd też, poprawiając rozdzielczość w dziedzinie czasu, będziemy ją pogarszać w dziedzinie częstotliwości i odwrotnie. Zatem szerokość okna wybierana jest na drodze kompromisu.

Przechodząc do konkretów, założmy, że $\varphi(t) \in L^2(\mathfrak{R})$, jest funkcją rzeczywistą i opisuje kształt okna czasowego. Efekt zastosowania okna czasowego opisuje zależność:

$$x_{\tau}(t) =: x(t) \varphi(t - \tau) \quad (3.2)$$

Dla przypadku okna prostokątnego (rys.3.1): $\varphi_r(t) = \chi_{[-\varepsilon, \varepsilon]}(t)$



Rys. 3.1. Prostokątne okno czasowe

Efekt działania prostokątnego okna czasowego opisuje zależność (3.3).

$$x_{\tau}(t) = \begin{cases} x(t), & t \in [\tau - \varepsilon, \tau + \varepsilon] \\ 0 & \text{dla pozostałych} \end{cases} \quad (3.3)$$

Teoretycznie, funkcja nie może być jednocześnie ograniczona w dziedzinie czasu i w dziedzinie częstotliwości. Jednak, przy tak zdefiniowanych *promieniach* okien, można osiągnąć jednoczesną ich ograniczoność w dziedzinie czasu i częstotliwości. W takim przypadku $\varphi(t)$ określa się mianem okna *czasowo-częstotliwościowego*. Niestety, okno prostokątne nie jest tutaj dobrym przykładem, gdyż $\Delta_{\varphi} = 3^{-1/2}$, a $\Delta_{\Phi} = \infty$. Wynika stąd jasno, że jest to *idealne okno czasowe* i jednocześnie *najgorsze, wręcz nieakceptowalne okno częstotliwościowe*.

Miarą jakości okna czasowo-częstotliwościowego jest iloczyn promienia w dziedzinie czasu i częstotliwości, czyli „pole” okna położonego na płaszczyźnie t/f. Przy czym zgodnie z *zasadą nieoznaczoności*:

$$\Delta_t \Delta_{\omega} \geq \frac{1}{2} \quad (3.4)$$

W kontekście powyższych rozważań, krótkoczasową transformatę Fouriera (STFT) sygnału $x(t)$, w odniesieniu do okna $\varphi(t)$ rozmieszczonego w pozycji (τ, ξ) na płaszczyźnie t/f zdefiniować można jako:

$$G(\tau, \xi) =: \int_{-\infty}^{+\infty} x(t) \overline{\varphi_{\tau, \xi}(t)} dt \quad (3.5)$$

gdzie: $\varphi_{\tau, \xi}(t) =: \varphi(t - \tau) e^{j\xi t}$.

Dopuszcza się, aby $\varphi(t)$ było funkcją zespoloną i w związku z tym musi spełniać uogólniony warunek zapewniający, że $\Phi(\omega)$ stanowi filtr dolnoprasmowy:

$$\Phi(0) = \int_{-\infty}^{+\infty} \varphi(t) dt \neq 0 \quad (3.6)$$

W odróżnieniu od tradycyjnej transformaty Fouriera, dla której, do wyznaczenia pojedynczej składowej, konieczna jest znajomość funkcji $x(t)$ na całej osi czasu, w tym przypadku wymagana jest znajomość $x(t)$ tylko w przedziale określonym przez położenie $\varphi(t - \tau)$.

Niezwykle interesujące jest podejście przyjmujące, że w STFT okno $\varphi(t)$ użyte jest do modulacji sygnału sinusoidalnego $e^{-j\xi t}$. Wyrażenie opisujące STFT przyjmuje wtedy postać:

$$G(\tau, \xi) =: \left\langle x(t) \varphi(t - \tau) e^{j\xi t} \right\rangle = \left\langle x(t) \varphi_{\tau, \xi}(t) \right\rangle \quad (3.7)$$

Można powiedzieć, że funkcja:

$$\varphi_{\tau,\xi}(t) =: \varphi(t - \tau)e^{j\xi t} \quad (3.8)$$

zachowuje się tak jak *pakiet fal*, gdzie sinusoida oscyluje wewnątrz okna $\varphi(t)$. Każdy *pakiet* traktować można jako pewną *funkcję bazową*. Podejście to pozwala traktować STFT jako rozkład $x(t)$ na składowe należące do tak zdefiniowanej bazy na płaszczyźnie t/f.

Funkcję $x_\tau(t)$ można odtworzyć wyznaczając odwrotną transformatę Fouriera o postaci:

$$x_\tau(t) = \varphi(t - \tau)x(t) = \frac{1}{2\pi} \int_{-\infty}^{+\infty} G(\tau, \xi) e^{j\xi t} d\xi \quad (3.9)$$

Funkcję oryginalną $x(t)$, natomiast otrzyma się dopiero po przebiegnięciu wszystkich pozycji położenia okna:

$$x(t) = \frac{1}{2\pi \|\varphi(t)\|} \int_{-\infty}^{+\infty} d\xi e^{j\xi t} \int_{-\infty}^{+\infty} G(\tau, \xi) \overline{\varphi(t - \tau)} d\tau \quad (3.10)$$

3.1.1. Transformata Gabora

Znany jest pewien przypadek szczególny dla STFT, w którym okno czasowe ma kształt *gaussowski*:

$$\varphi(t) = g_\alpha(t) = \frac{1}{\sqrt{2\pi\alpha}} e^{-\frac{t^2}{4\alpha}}, \quad \alpha > 0 \quad (3.11)$$

Okazuje się, że widmo tego okna ma również kształt *gaussowski*:

$$\Phi(\omega) = G_\alpha(\omega) = e^{-\alpha\omega^2}, \quad \alpha > 0 \quad (3.12)$$

Przypadek ten spełnia warunki okna czasowo-częstotliwościowego o parametrach:

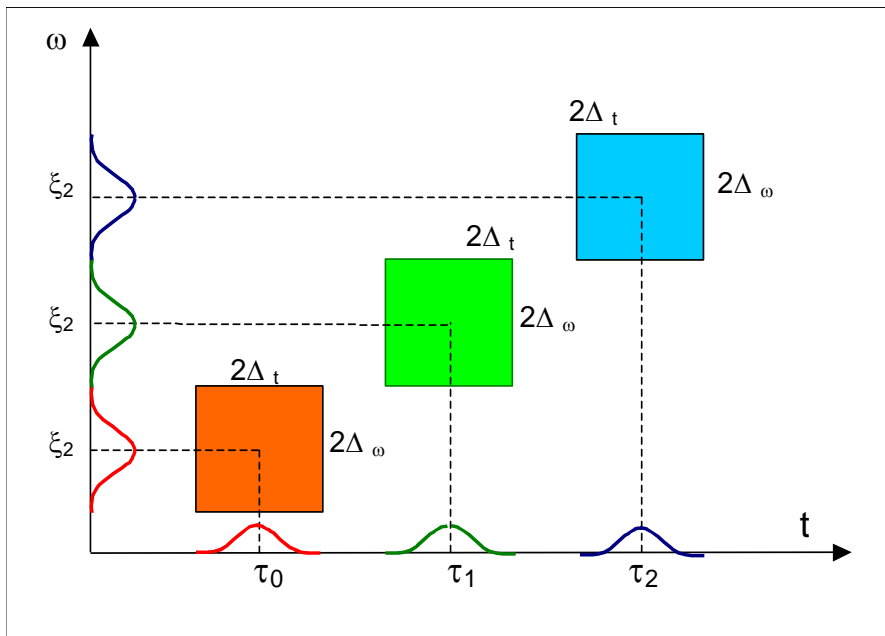
- okno czasowe: $t^* = 0$, $\Delta_t = \sqrt{\alpha}$
- okno częstotliwościowe: $\omega^* = 0$, $\Delta_\omega = \frac{1}{2\sqrt{\alpha}}$

Okno to, od nazwiska pierwszego użytkownika, nosi miano okna *Gabora* [10]. Miara jakości okna *Gabora* wynosi:

$$\Delta_t \cdot \Delta_\omega = \frac{1}{2} \quad (3.13)$$

i osiąga dolną granicę zasady niepewności.

Idea analizy czasowo-częstotliwościowej z użyciem transformaty Gabora, przedstawiona jest na rys.3.2.



Rys. 3.2. Zobrazowanie idei STFT na płaszczyźnie t/f

Analityczny zapis tej transformaty przedstawia zależność (3.14):

$$G(b, \xi) := \int_{-\infty}^{+\infty} x(t) \overline{g_{\alpha}(t-\tau)} e^{-j\xi t} dt = \frac{1}{2\pi\alpha} \int_{-\infty}^{+\infty} f(t) e^{-\frac{(t-\tau)^2}{4\alpha}} e^{-j\xi t} dt \quad (3.14)$$

3.1.2. Właściwości STFT

Właściwości krótkoczasowej transformaty Fouriera ułatwiają zrozumienie istoty analizy czasowo-częstotliwościowej. Poniżej przedstawiono trzy najbardziej istotne:

- liniowość: $x(t) = ax_1(t) + bx_2(t)$

$$G = aG_1(\tau, \xi) + bG_2(\tau, \xi) \quad (3.15)$$

- przesunięcie w czasie: $x_0(t) = x(t - t_0)$

$$\begin{aligned} G_0(b, \xi) &= \int_{-\infty}^{+\infty} x(t - t_0) \overline{\varphi(t - \tau)} e^{-j\xi t} dt = \\ &= \int_{-\infty}^{+\infty} f(t) \overline{\varphi[t - (\tau - t_0)]} e^{-j\xi t} e^{-j\xi t_0} dt = \\ &= e^{-j\xi t_0} G(\tau - t_0, \xi) \end{aligned} \quad (3.16)$$

Oznacza to, że jeżeli położenie oryginalnej funkcji $x(t)$ zostanie przesunięte w czasie o t_0 , to lokalizacja widma na płaszczyźnie t/f zostanie również przesunięta o t_0 w czasie, a lokalizacja częstotliwościowa pozostanie bez zmian. Dodatkowo pojawi się przesunięcie fazowe proporcjonalne do t_0 .

- przesunięcie w częstotliwości: $x_0(t) = x(t) e^{j\omega_0 t}$

$$\begin{aligned}
G(\tau, \xi) &= \int_{-\infty}^{+\infty} x(t) e^{j\omega_0 t} \varphi(t - \tau) e^{-j\xi t} dt = \\
&= \int_{-\infty}^{+\infty} x(t) \varphi(t - \tau) e^{-j(\xi - \omega_0)t} dt = \\
&= \langle x(t), \varphi(t - \tau) e^{-j(\xi - \omega_0)t} \rangle = G(\tau, \xi - \omega_0)
\end{aligned} \tag{3.17}$$

Powyzsza zaleznosc potwierdza, ze w przypadku „przesunięcia” funkcji $x(t)$ w dziedzinie czestotliwosci, widmo STFT zostaje rowniez przesunięte na osi czestotliwosci bez zmiany amplitudy i fazy.

Do celow obliczen numerycznych, jak zwykle wprowadza sie pojecie Dyskretnej Transformaty Gabora, okreslanej w skonczonej zbiorze punktow na plaszczyznie t/f. Wtedy:

$$\begin{aligned}
G(\tau_n, \xi_k) &= \int_{-\infty}^{+\infty} x(t) \overline{g_\alpha(t - \tau_n)} e^{-j\xi_k t} dt = \\
&= \langle x(t), g_\alpha(t - \tau_n) e^{j\xi_k t} \rangle = \langle x(t), \varphi_{n,k}(t) \rangle
\end{aligned} \tag{3.18}$$

Funkcja $x(t)$ moze zostac odtworzona na podstawie zaleznosci okreslanej mianem *rozkładu Gabora*:

$$f(t) = \sum_n \sum_k G(\tau_n, \xi_k) g_\alpha(t - \tau_n) e^{j\xi_k t} \tag{3.19}$$

Współczynnikami tego rozkładu są liczby $G_{n,k} = G(\tau_n, \xi_k)$. W kontekście powyższego warto podać uogólnioną formułę rozkładu Gabora, w której występuje *pakiet fal sinusoidalnych* zmodulowanych za pomocą okna gaussowskiego o symbolu $\varphi_{n,k}(t)$:

$$x(t) = \sum_n \sum_k G_{n,k} \varphi_{n,k}(t) \tag{3.20}$$

Pakiet ten określaný jest mianem *bazy rozkładu Gabora*. Baza ta, z uwagi na gaussowski kształt okna o minimalnym rozmiarze na płaszczyznie t/f, uchodzi za optymalną w sensie budowy reprezentacji sygnału.

Poniżej (rys.3.3) podany jest przykład zastosowania rozkładu STFT do sygnału o postaci [10]:

$$x(t) = \sin 2\pi f_1 t + \sin 2\pi f_2 t + \alpha [\delta(t - t_1) + \delta(t - t_2)] \tag{3.21}$$

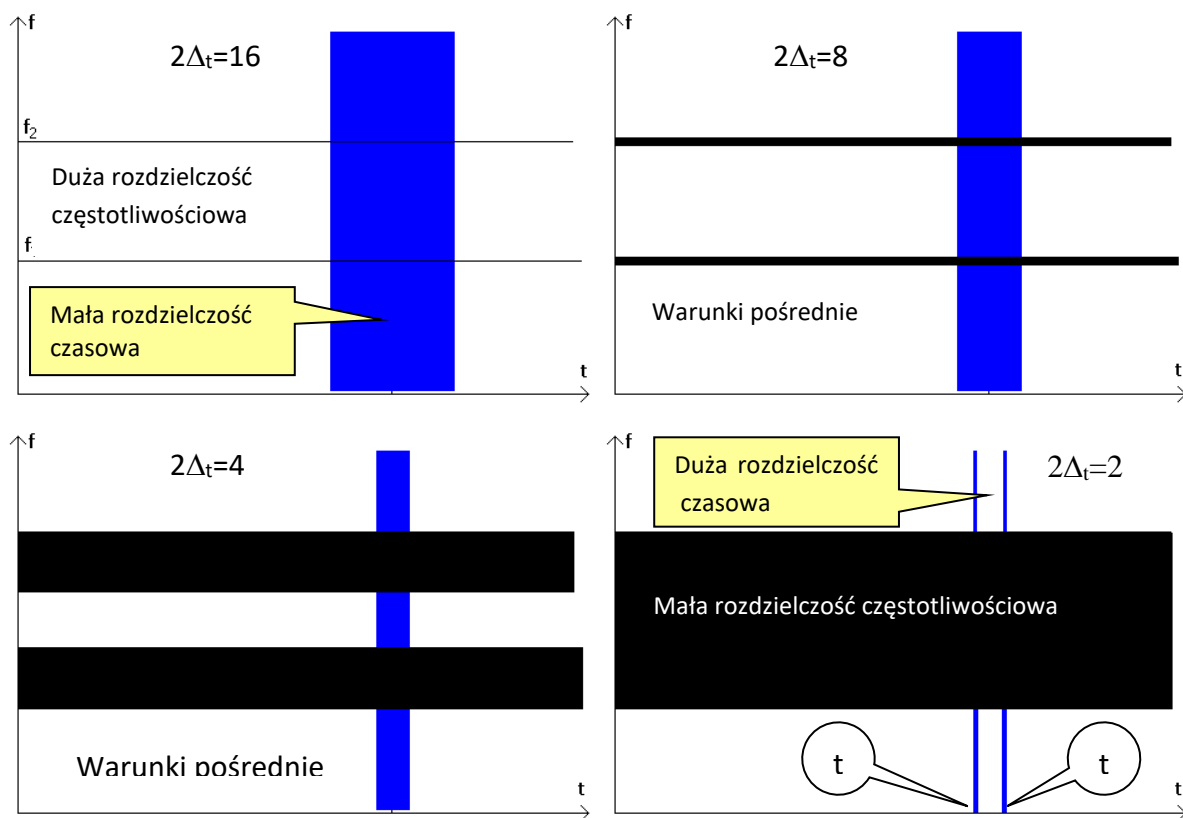
dla różnych szerokości (promieni) Δ_t okna czasowego $\varphi(t)$.

W pierwszym przypadku, dla szerokiego okna czasowego, otrzymuje się bardzo dobrą rozdzielczość czestotliwosciową i bardzo słabą rozdzielczość czasową. W miarę zężenia okna czasowego poprawia się rozdzielczość czasowa kosztem czestotliwosciowej. Dla dobrego zobrazowania wszystkich elementow składowych sygnału, trzeba zmieniać parametry okna czasowo-czestotliwosciowego i za każdym razem, od nowa wyznaczać STFT. Wymaga to dużych nakładow obliczeniowych.

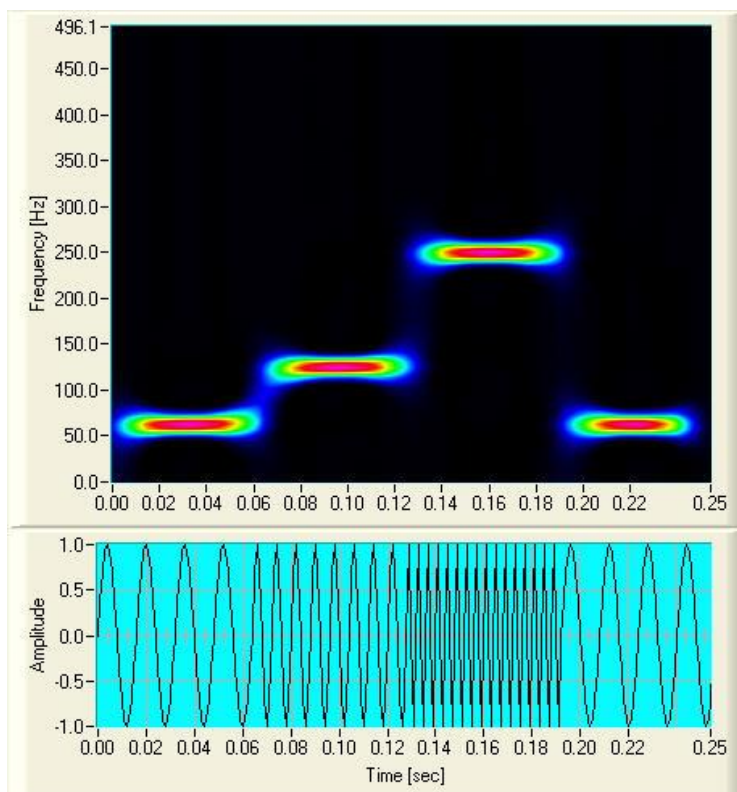
Na zakończenie tego punktu na kolejnych rysunkach pokazano praktyczne wyniki analizy t/f, przeprowadzonej z użyciem transformaty Gabora, za pomocą wirtualnych przyrządow pomiarowych [11]⁵. W obydwu przypadkach użyte zostały bardzo charakterystyczne sygnały, ułatwiające interpretację rozkładow widmowych

⁵ Przyrządy opracowane w Zakładzie Miernictwa Elektrycznego IETiME PW w ramach projektu badawczego

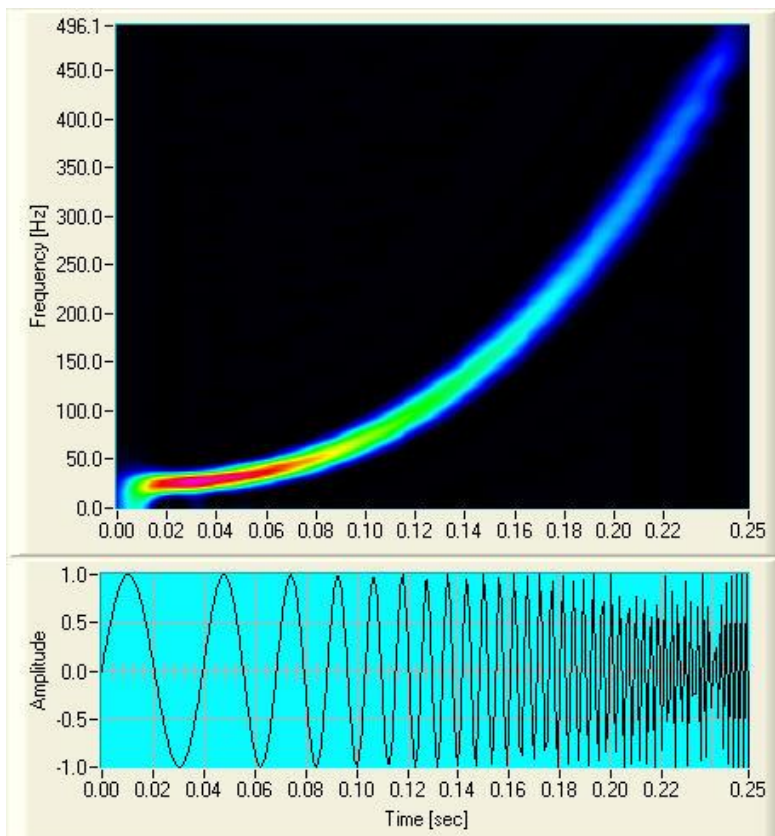
[16]. W pierwszym przypadku (rys.3.4) jest to sygnał złożony z segmentów powstałych w wyniku skokowej zmiany częstotliwości, a w drugim (rys. 3.5) sygnał o zmodulowanej częstotliwości ze zmiennością wykładniczą.



Rys. 3.3. Przykład zastosowania rozkładu STFT dla czterech różnych szerokości okna czasowego



Rys. 3.4. Praktyczny wynik analizy t/f, przeprowadzonej z użyciem transformaty Gabora



Rys. 3.5. Praktyczny wynik analizy t/f, przeprowadzonej z użyciem transformaty Gabora

Podsumowanie właściwości aplikacyjnych STFT:

- Δ_t i Δ_ω - nie zależą od położenia na płaszczyźnie (t/f).
- ustalona rozdzielczość czasowo – częstotliwościowa
- W analizie czasowo-częstotliwościowej wskazane byłoby posługiwanie się oknem *strojonym*:
 - szerokie w czasie (wąskie w częstotliwości) dla małych częstotliwości
 - wąskie w czasie (szerokie w częstotliwości) dla dużych częstotliwości.

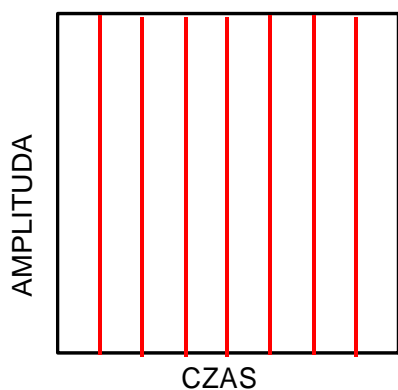
4. Analiza falkowa

Z zasady działania transformaty Fouriera wynika jasno, że rozdzielczość czasowa analizy dokonanej z jej pomocą, nie jest zadowalająca. Innymi słowy, jeżeli analizujemy sygnał, który zmienia się gwałtownie z poziomu $-1V$ do $+1V$ to do jego opisu musimy użyć bardzo dużej liczby współczynników (teoretycznie – nieskończenie wiele), a ponadto nie jesteśmy w stanie określić chwili czasu, w której ta zmiana nastąpiła. Dla jej określenia możemy oczywiście użyć opisanego algorytmu STFT. Jednakże, znaczącą poprawę jakości analizy otrzymamy dopiero po „umiejętnym” zastosowaniu metody *falkowej*.

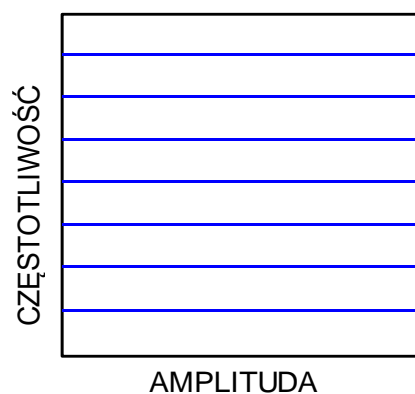
W przypadku transformaty falkowej, najbardziej charakterystyczne jest to, że w odróżnieniu od funkcji sinus i cosinus (charakterystycznych dla transformaty Fouriera), indywidualne funkcje *falkowe* są dobrze zlokalizowane w czasie (lub przestrzeni – dla obrazów) i jednocześnie podobnie jak sinus i cosinus, indywidualne *falki* są dobrze zlokalizowane w częstotliwości, ściśle biorąc tzw. *skali*. Ponadto w odróżnieniu od funkcji *sinus* i *cosinus*, które definiują unikalną transformatę Fouriera, nie ma pojedynczego, unikalnego zbioru *falkowych funkcji bazowych*. Falki różnią się między sobą *zwartością lokalizacji czasowej* oraz *płynnością i gładkością kształtów*. Wynikająca

stąd zdolność falek do opisu sygnałów „z nieciągłościami”, przy ograniczonej liczbie współczynników oraz z lokalizacją w czasie, stanowi o jej przewadze nad transformatą Fouriera. Zaryzykować można stwierdzenie, że analiza falkowa jest „lokomotywą” w dziedzinie cyfrowego przetwarzania sygnałów.

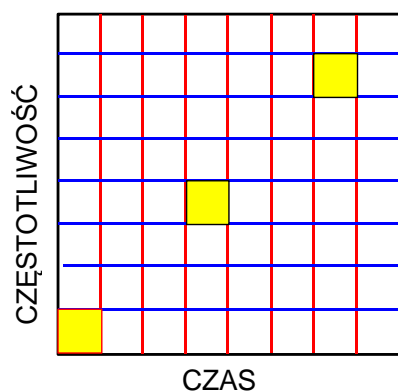
Na rys. 4.1 zilustrowano ideę czterech znanych metod analizy sygnałów: czasową, częstotliwościową, czasowo-częstotliwościową (STFT) oraz falkową. Widać na nim wyraźnie, że w odróżnieniu od metody STFT, gdzie rozdzielczość czasowo-częstotliwościowa jest ustalona na całej płaszczyźnie t/f, w metodzie falkowej rozmiary okna czasowo-częstotliwościowego są funkcją jego położenia na tej płaszczyźnie.



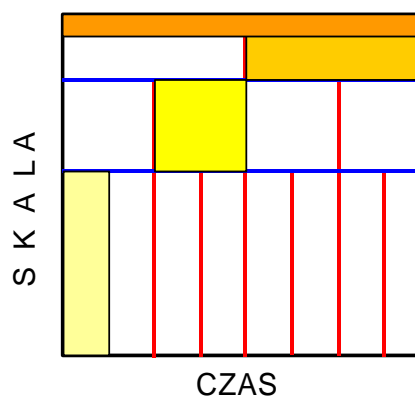
Analiza czasowa



Analiza częstotliwościowa



Analiza STFT



Analiza falkowa

Rys. 4.1. Porównanie metod analizy: **czasowa** (obserwacja właściwości sygnału w dziedzinie czasu), **częstotliwościowa** (obserwacja właściwości sygnału w dziedzinie częstotliwości), **STFT** (obserwacja właściwości sygnału na płaszczyźnie czas-częstotliwość), **falkowa** (obserwacja właściwości sygnału na płaszczyźnie czas-skala)

Swoją niezwykłą efektywność, a zarazem popularność w zakresie analizy sygnałów, transformata falkowa zawdzięcza szybkiemu algorytmowi, opracowanemu przez Mallata w roku 1989, zwanemu *piramidą Mallata* [13]. Algorytm ten wykorzystywany jest do uzyskania dekompozycji sygnału na składowe falkowe z użyciem tzw. *kwadraturowych filtrów lustrzanych*. Zastosowane w nim podejście wielorozdzielcze przenosi metodę falkową w realia kodowania podpasmowego.

Chociaż zakres pozycji literaturowych poświęconych metodzie falkowej jest niesłychanie rozległy [7, 10,13], to ogromna większość spośród nich pisana jest przez specjalistów dla specjalistów. Istnieją pozycje polskojęzyczne [2, 20]. Na szczególną uwagę zasługuje pozycja [22]. Celem tego rozdziału jest przybliżenie czytelnikowi praktycznych aspektów analizy falkowej, co miejmy nadzieję, będzie pomocne w zrozumieniu idei oraz prowadzeniu eksperymentów w tej dziedzinie [16].

Ciągłą (całkową) transformatę falkową (Continuous Wavelet Transform: CWT) funkcji $x(t) \in L^2$ dla pewnej falki $\psi(t)$ definiuje się jako:

$$W(\tau, \sigma) =: \int_{-\infty}^{+\infty} x(t) \overline{\psi_{\tau, \sigma}(t)} dt, \quad \psi_{\tau, \sigma}(t) = \frac{1}{\sqrt{\sigma}} \psi \frac{t - \tau}{\sigma} \quad (4.1)$$

We wzorze tym parametr σ oznacza skalę, zaś τ – przesunięcie, co odpowiada ich funkcjom pełnionym w zapisie wzoru falkowego. Po to, aby funkcja $\psi(t)$ mogła stanowić *funkcję okna*, a także być wykorzystana do odtworzenia $x(t)$ musi spełniać warunek:

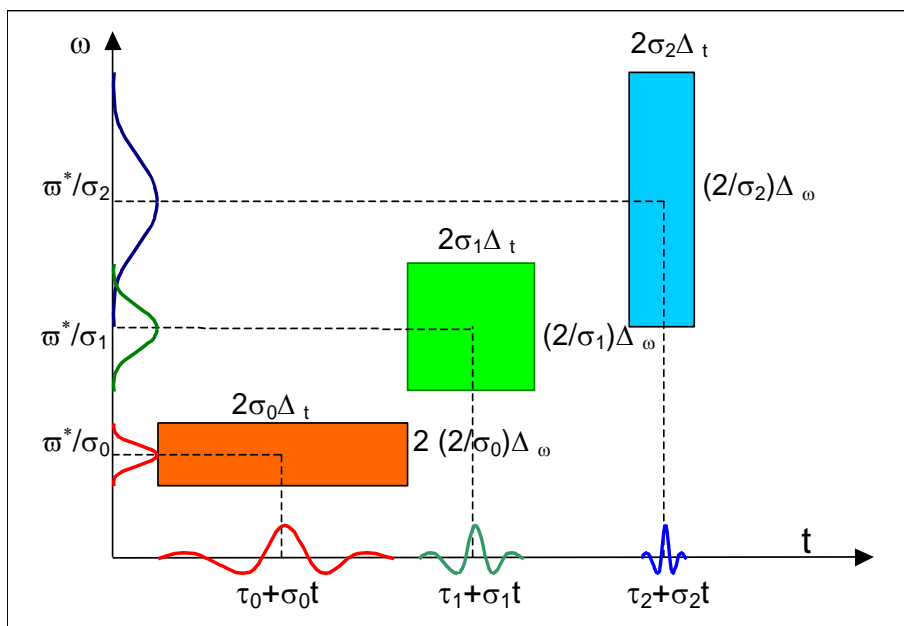
$$\Psi(\omega) \Big|_{\omega=0} = \int_{-\infty}^{+\infty} \psi(t) dt = 0 \quad (4.2)$$

Ściśle biorąc, ciągła transformata falkowa osadzona jest więc w przestrzeni czas-skala (t/s), a nie częstotliwość (t/f). Jednakże, w konkretnym przypadku, po dokonaniu odpowiedniej transformacji można przeliczyć skalę na częstotliwość. Miarą częstotliwości jest $1/\sigma$. Parametr τ symbolizuje lokalizację okna wzdłuż osi czasu. Wyrażenie (4.2) natomiast wymusza *oscylacje falek*, które nie były wymagane w stosunku do okna STFT. Jednakże definitywnie $\psi_{\tau, \sigma}(t)$ zajmuje miejsce $\varphi_{\tau, \xi}(t)$ i zachowuje się tak jak funkcja okna. Natomiast widmo tego okna $\Psi(\omega=0) = 0$ i odwzorowuje filtr pasmowy.

Warto przy tym zwrócić uwagę, że iloczyn promieni okien, czasowego i częstotliwościowego, ma wartość stałą na całej płaszczyźnie:

$$2\sigma\Delta_t \cdot \frac{2}{\sigma}\Delta_\omega = 4\Delta_t\Delta_\omega \quad (4.3)$$

Położenie okna czasowo-częstotliwościowego na płaszczyźnie t/f, dla transformaty falkowej, pokazano na rys.4.2.



Rys. 4.2. Zobrazowanie idei transformaty falkowej na płaszczyźnie t/f

Podobnie jak w przypadkach DFT oraz STFT definiuje się pojęcie dyskretnej transformaty falkowej (Discrete Wavelet Transform: DWT). W tym celu przyjmuje się, że:

$$\sigma = 2^{-s}, \quad \tau = 2^{-s} \cdot l \quad (4.4)$$

co w konsekwencji, po dodatkowym przyjęciu założenia o dyskretyzacji $x(t)$, daje nową formę zapisu transformaty:

$$\begin{aligned} W(l, s) &= W(l2^{-s}, 2^{-s}) = W(l, s) = \\ &= 2^{s/2} \int_{-\infty}^{+\infty} x(t) \psi(2^s t - l) dt = \\ &= 2^{s/2} \sum_n x(n) \psi(2^s n - l) \end{aligned} \quad (4.5)$$

Warto pamiętać, że transformata falkowa nie spełnia warunku *niezmienności względem przesunięcia*, a przesunięcie funkcji w czasie: $x_m(t) = x(t - \tau_m)$, objawia się w dziedzinie falkowej raczej w sposób bardzo nieprzyjazny [10]:

$$\begin{aligned} W_{\psi, x}(l, s) &= 2^{s/2} \int_{-\infty}^{+\infty} x_m(t) \psi(2^s t - l) dt = \\ &= 2^{s/2} \sum_n x(n - m) \psi(2^s n - l) = \\ &= 2^{s/2} \sum_n x(n) \psi[2^s n - (l - m2^s)] = \\ &= W_{\psi, x}[(l - m2^s)2^{-s}, 2^{-s}] \end{aligned} \quad (4.6)$$

Na tym etapie rozważań można się pokusić o sformułowanie wyrażenia na *szereg falkowy*, który istnieje dla dowolnej funkcji $x(t) \in L^2$:

$$x(t) = \sum_s \sum_l w_{l,s} \psi_{l,s}(t), \quad \psi_{l,s}(t) = 2^{s/2} \psi(2^s t - l) \quad (4.7)$$

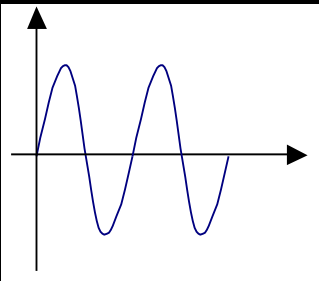
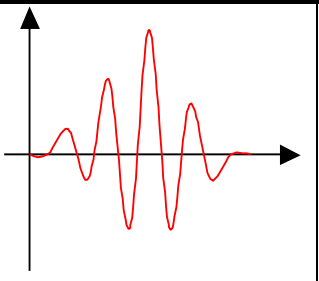
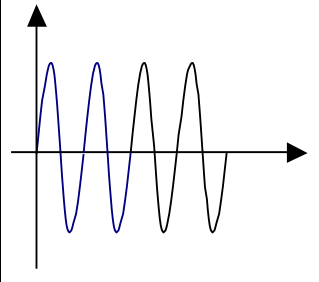
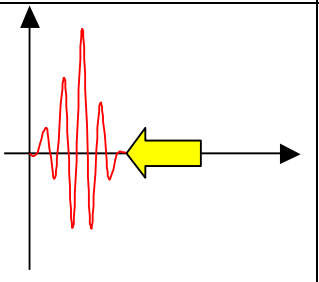
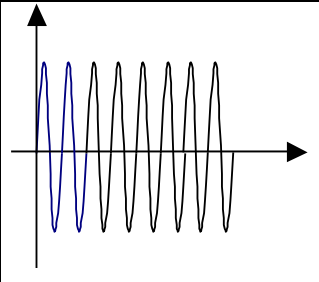
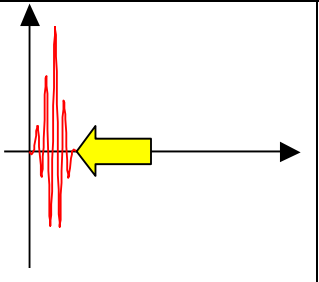
Jeżeli zatem $\{\psi_{k,s}(t)\}$ tworzy ortonormalną bazę w przestrzeni $L^2(\mathfrak{R})$ to podobnie jak w przypadku szeregu Fouriera:

$$w_{l,s} = \langle x(t), \psi_{k,s}(t) \rangle = 2^{s/2} W_{\psi, x}(l, s) \quad (4.8)$$

Okazuje się, że w odróżnieniu od transformaty Fouriera szereg falkowy otrzymuje się po *spróbkowaniu* transformaty ciągłej na płaszczyźnie t/f, w wybranych punktach diadycznych: $(l2^{-s}, 2^{-s})$. Analiza jest utrudniona, gdy baza $\{\psi_{k,s}(t)\}$ nie spełnia warunku ortonormalności. Wymaga to zdefiniowania tzw. falki dualnej i sformułowania bazy: $\{\tilde{\psi}_{l,s}(t)\}$ [10, 13].

W tabeli 4.1 zamieszczono przykłady skalowania funkcji dla przypadku sinusoidy oraz pewnej (typowej) falki (a - *współczynnik skali*). W przypadku sinusoidy istnieje ściśle odwzorowanie skala-częstotliwość, dla falki nie jest ono tak oczywiste. W związku z tym, jak już wspomniano wcześniej, pozostaje się przy pojęciu skalowania.

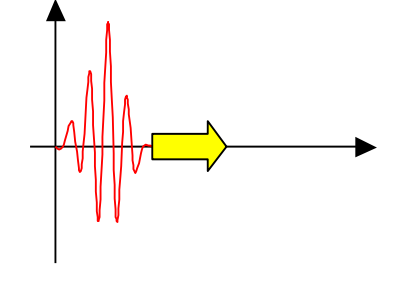
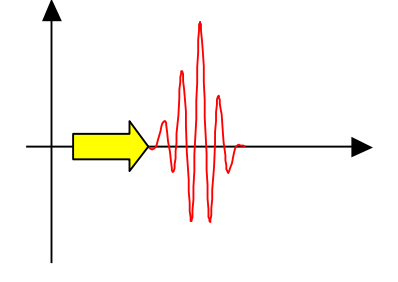
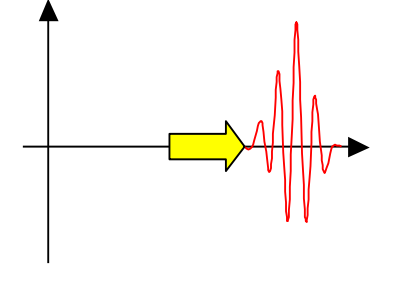
Tabela 4.1 Przykłady skalowania funkcji

	$x(t) = \sin t$ $a = 1$		Wysoka skala $x(t) = \psi(t)$ $a = 1$
	$x(t) = \sin 2t$ $a = 2$		Średnia skala $x(t) = \psi(2t)$ $a = 2$
	$x(t) = \sin 4t$ $a = 4$		Niska skala $x(t) = \psi(4t)$ $a = 4$

Proces skalowania falki może przebiegać w dwu kierunkach, określa się je mianem *kompresji* (ściskania) lub *rozciągania*. W tabeli 4.1 do skalowania falki zastosowano kompresję.

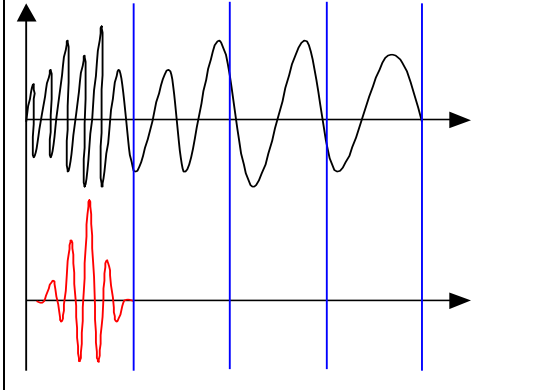
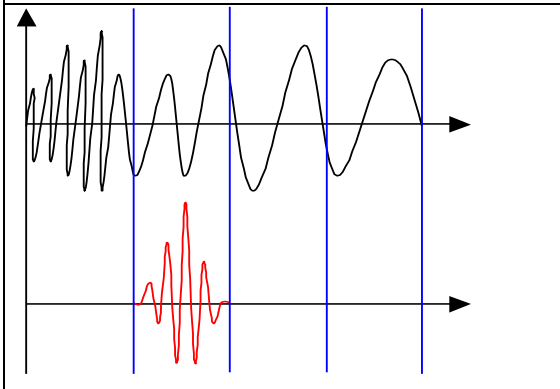
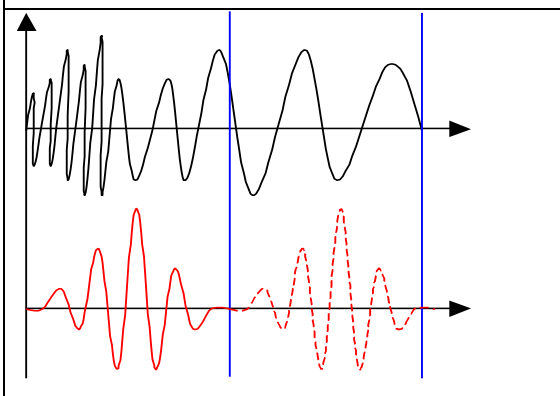
Drugi parametr rozkładu falkowego to przesunięcie. Ilustracja przesuwania w czasie zawarta jest w tabeli 4.2.

Tabela 4.2 Ilustracja przesuwania w czasie falki

		
$\psi(t)$	$\psi(t - \tau)$	$\psi(t - 2\tau)$

Na tle powyższych rozważań można, w sposób opisowy, zdefiniować proces rozkładu falkowego. Zawiera on 5 charakterystycznych kroków (tabela 4.3).

Tabela 4.3 Ilustracja rozkładu falkowego

	<p>1. Wybraną falkę ustawić na początku fragmentu sygnału przeznaczonego do analizy, Wyznaczyć umowną wartość liczbową odpowiadającą korelacji między bieżącą falką i odpowiadającym jej segmentem sygnału. Uwaga! – w przypadku unormowania energii sygnału w aspekcie użytej falki, wspomniana liczba będzie równoważna wartości współczynnika korelacji wzajemnej między falką, a wybranym segmentem sygnału.</p>
	<p>2. Przesunąć falkę o jeden cykl w prawo i powtórzyć działanie opisane w kroku 2. Sekwencję kroków 3,2 powtarzać aż do końca trwania sygnału.</p>
	<p>3. Rozciągnąć falkę i powtórzyć kroki od 1 do 3. 4. Powtórzyć kroki od 1 do 4 aż do wyczerpania wszystkich skal.</p>

Powyższy przykład operuje w obrębie tzw. *diadycznego* charakteru zmian w obrębie skali i przesunięcia charakterystycznych dla dyskretnej transformaty falkowej (DWT). Pod pojęciem ciągłej transformaty falkowej (CWT) kryje się sposób umożliwiający użycie dowolnej, zmienianej w sposób ciągły skali oraz ciągłego przesunięcia w czasie. Oczywiście, w kontekście sygnałów dyskretnych ciągłość, w obydwu wskazanych przypadkach, oznacza zmiany w obrębie jednej próbki sygnału (co jedną próbkę),

Należy zauważyć, że w analizie falkowej wyższa skala równoważna jest bardziej rozciągniętej falce (patrz tabela 4.1). Im bardziej rozciągnięta falka (wyższa skala) tym większa sekcja sygnału, z którą jest porównywana i tym bardziej zgrubne cechy sygnału wyeksponowane za pomocą odpowiadającego jej współczynnika. Podsumowanie tego spostrzeżenia zawarto w tabeli 4.4

Tabela 4.4 Charakterystyka analizy falkowej

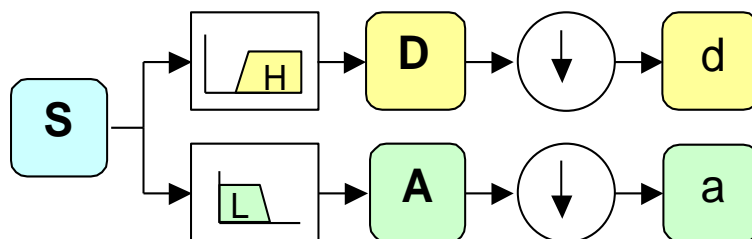
Niska skala analizy falkowej	Ściśnięta falka	Szybkozmiennie detale	Wysoka częstotliwość analizy
Wysoka skala analizy falkowej	Rozciągnięta falka	Wolnozmiennie cechy sygnału	Niska częstotliwość analizy

Fakt, że analiza falkowa nie obrazuje cech sygnału na płaszczyźnie czas-częstotliwość, lecz czas-skala nie stanowi o słabości metody, a wręcz przeciwnie – o jej sile. Okazuje się, że jest to naturalna metoda opisu wielu zjawisk fizycznych odbieranych przez zmysły człowieka. Trzeba się z nią pogodzić i do niej przyzwyczaić.

Zjawiskiem ściśle związanym z metodą falkową jest *analiza wielorozdzielcza* [1]. Jak wskazuje nazwa, umożliwia ona obserwację właściwości sygnału na różnych poziomach aproksymacji lub inaczej *rozdzielczości*. Stosując analizę wielorozdzielczą dokonuje się podziału złożonej funkcji na kilka (lub więcej) prostych składowych i bada je niezależnie.

Bardzo efektywna metoda implementacji tego algorytmu analizy dokonanej z użyciem filtrów opracowana została w 1988 roku przez Mallata [13]. Nawiązuje ona do, znanej z analizy częstotliwościowej, metody kodowania w podpasmach i realizuje tzw. *szybką transformację falkową* (Fast Wavelet Transform: FWT).

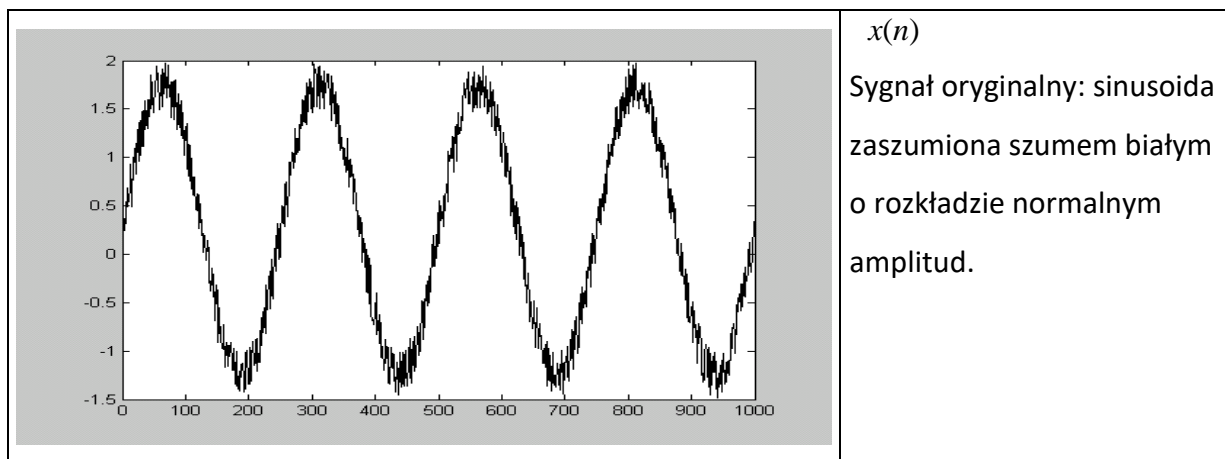
Do analizy falkowej wprowadzono dwa bardzo adekwatne pojęcia: *aproksymacji* i *detalu*. Pod pojęciem *aproksymacji* rozumie się wysoką skalę analizy, niskoczęstotliwościowe składowe sygnału. *Detale* to niska skala i wysokoczęstotliwościowe składowe. Wspomniany proces filtracji, w swoim najniższym poziomie obejmuje dwa filtry: dolnopasmowy (L) i górnopasmowy (H). Filtr dolnopasmowy odtwarza *aproksymację*, a górnopasmowy *detal* sygnału (rys.4.3).

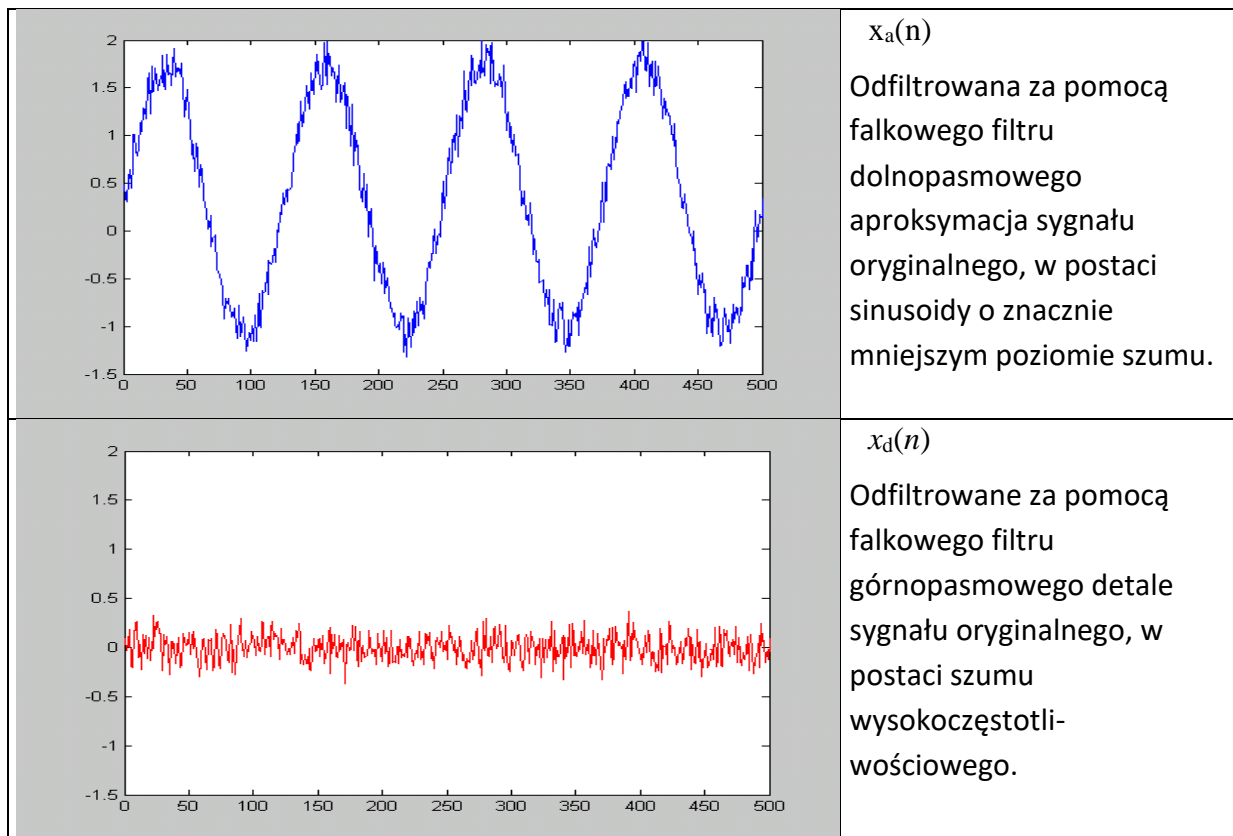


Rys. 4.3. Schemat filtracyjnej dekompozycji falkowej

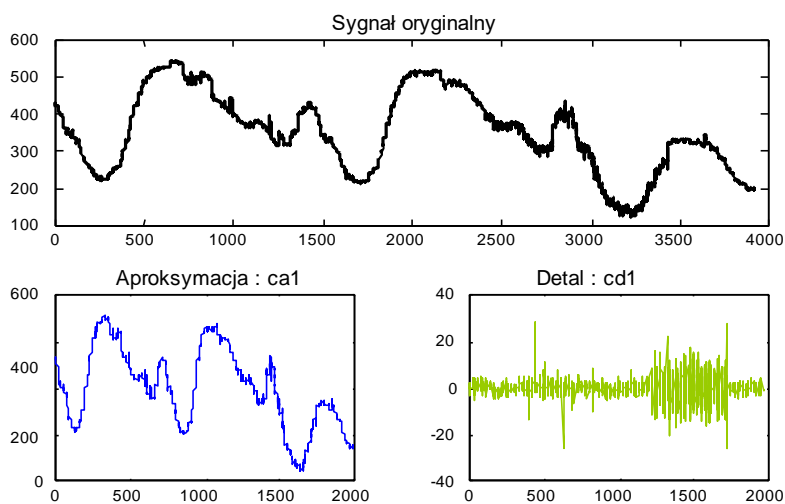
Oryginalny sygnał S przechodzi przez parę komplementarnych filtrów, które rozdzielają go na dwie składowe A (Aproksymacja) i D (Detal). W przypadku filtracji cyfrowej podwaja się liczba danych, przeznaczonych do dalszego przetwarzania. Wygodnym sposobem ograniczenia tej liczby w metodzie falkowej jest *decymacja*, polegająca na odrzuceniu co drugiej próbki danych. Przykład działania filtrów falkowych wraz z interpretacją zawiera tabela 4.5.

Tabela 4.5 Interpretacja działania filtrów falkowych



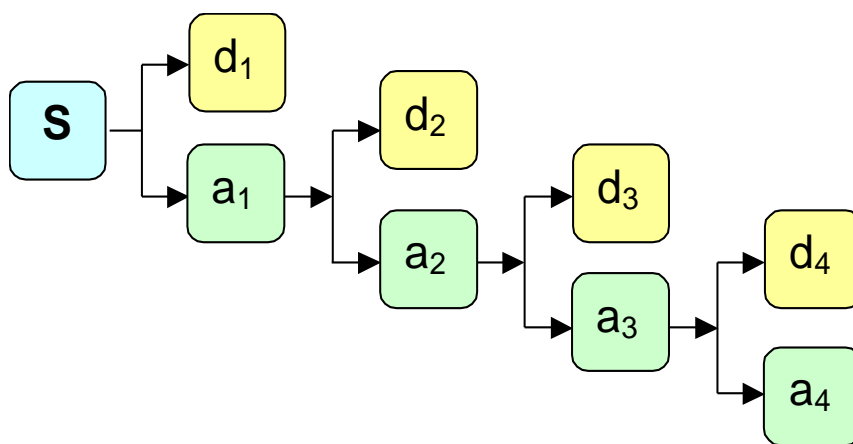


Przykład pierwszego poziomu dekompozycji falkowej pewnego rzeczywistego sygnału pomiarowego zamieszczono na rysunku 4.4.



Rys. 4.4. Przykład pierwszego poziomu dekompozycji falkowej sygnału

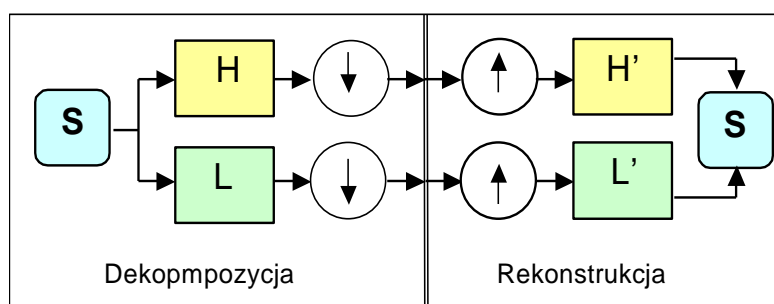
Pełny proces dekompozycji zawiera szereg członów tworzących tzw. *drzewo dekompozycji falkowej*. Przykład takiego drzewa zamieszczono na rysunku 4.5.



Rys. 4.5. Drzewo dekompozycji falkowej: S – sygnał oryginalny, d_i – detal w i-tej skali, a_i –aproxymacja w i-tej skali

Z powodu tego, że proces dekompozycji ma charakter iteracyjny, teoretycznie może on trwać *do nieskończoności*. W rzeczywistości, może być kontynuowany tylko do chwili, aż w kolejnym detalu pozostanie pojedyncza próbka. W praktyce zaś, wybiera się odpowiednią liczbę poziomów, dostosowaną do natury sygnału lub innego kryterium.

Z metrologicznego punktu widzenia można by poprzestać na omówieniu procesu dekompozycji falkowej, zwanego inaczej procesem *analizy*. Nierzadko jednak, np. w zastosowaniu do kompresji sygnałów konieczne jest, możliwie wierne, odtworzenie postaci sygnału oryginalnego. Proces ten, w metodzie falkowej określany jest mianem rekonstrukcji lub *syntezy*. Matematyczny zapis tego procesu obejmuje definicję *odwrotnej dyskretnej transformaty falkowej*. Metoda filtracyjna zaś, polega na użyciu tzw. filtrów rekonstrukcyjnych. Znamienna dla metody falkowej jest konieczność tzw. *rekonstrukcji perfekcyjnej*. Jest ona następstwem systematycznego wprowadzania zniekształceń składowych sygnału występujących w procesie analizy drogą usuwania co drugiej próbki. W procesie rekonstrukcji, *brakujące próbki* zastępowane są wartościami zerowymi. Filtry rekonstrukcyjne zaś dobrane są w taki sposób, aby „*niwelowały*” zgubne następstwa aliasingu. Wspólnie z filtrami użytymi w procesie analizy tworzą one system określany mianem *kwadraturowych filtrów lustrzanych*: $H-H', L-L'$ (rys.4.6).



Rys. 4.6. Złożenie dekompozycji i rekonstrukcji

Dobór charakterystyk filtrów rozkładu falkowego jest jednoznaczny z doбором kształtu falki tego rozkładu. Wydaje się, że z punktu widzenia analizy metrologicznej ten sposób doboru parametrów jest lepszy niż pierwotne skonstruowanie falki, a potem projektowanie dla niej filtru⁶. Zwłaszcza, że niemożliwe jest użycie całkiem dowolnej ograniczonej w czasie funkcji, o zerowej wartości średniej, nazwanej jej falką i użycie w

⁶ W szczególnych przypadkach może się okazać, że łatwiej jest dobrać kształt falki do kształtu sygnału. Przykładem może tu być sygnał elektrokardiogramu.

procesie analizy, jeżeli zamierza się zrekonstruować sygnał w przyszłości. Trzeba to zrobić w kontekście istnienia zespołu kwadraturowych filtrów lustrzanych.

Ściśle rzecz ujmując, kształt falki $\psi(t)$ jest jednoznacznie związany z charakterystyką filtra górnopasmowego wyodrębniającego detal w rozkładzie falkowym. Istnieje jeszcze jedna bardzo charakterystyczna funkcja, która związana jest z niektórymi (nie wszystkimi) zbiorami falek. Jest to tzw. *funkcja skalująca*, oznaczana symbolem $\varphi(t)$. Jej kształt związany jest z charakterystykami dolnopasmowych kwadraturowych filtrów lustrzanych odpowiedzialnych za wyodrębnienie *aproxymacji* sygnału. Kształt funkcji skalującej jest zbliżony do kształtu odpowiadającej jej falki. Definiuje się ją w rekurencyjnym zapisie matematycznym za pomocą równania dylatacyjnego:

$$\varphi(t) = \sqrt{2} \sum_{k=0}^{N-1} h_k \varphi(2t-k) \quad (4.9)$$

W kontekście funkcji skalującej falka zdefiniowana jest za pomocą tego samego równania dylatacyjnego opisanego za pomocą innego zestawu współczynników.

$$\psi(t) = \sqrt{2} \sum_{k=0}^{N-1} g_k \varphi(2t-k) \quad (4.10)$$

Współczynniki $H=\{h_k\}$, oraz $G=\{g_k\}$ są rozumiane jako współczynniki pary kwadraturowych filtrów lustrzanych. W przypadku bazy ortonormalnej związane są zależnością wzajemną: $g_k=(-1)^k h_{N-k}$.

Jak już wspomniano wcześniej, falkowe funkcje bazowe otrzymuje się z falki matki drogą skalowania i przesunięcia:

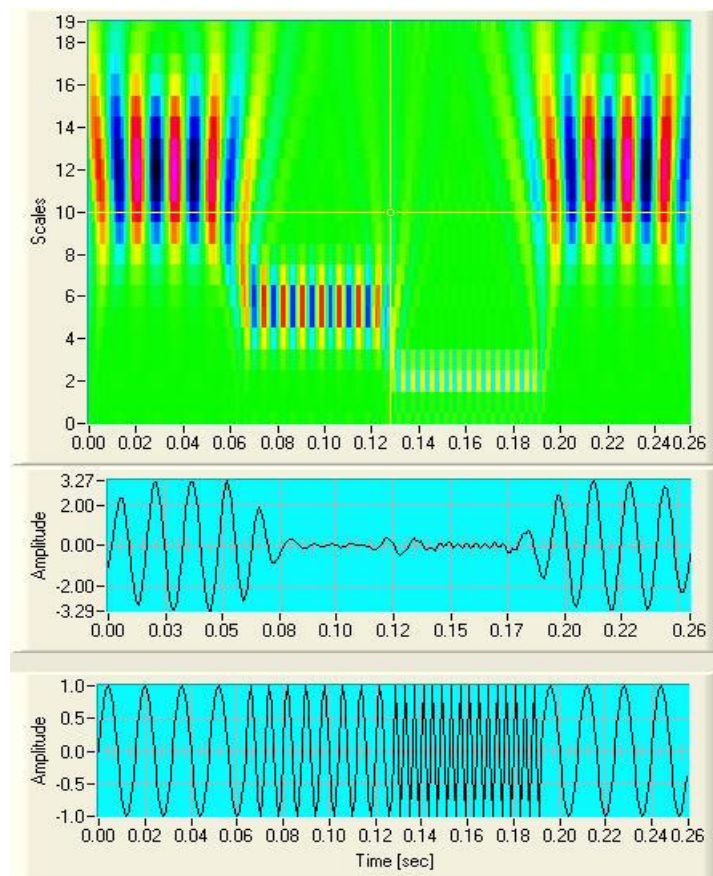
$$\psi_{s,l}(t) = \sqrt{2^s} \psi(2^s t - l), \quad (4.11)$$

Istnieje nieograniczona wręcz liczba możliwych do utworzenia falek jak i tzw. banków filtrów. Która z nich jest najlepsza zależy od konkretnej implementacji, tak jak było to w przypadku doboru okna czasowego w analizie widmowej. Nazwy konkretnych rozwiązań pochodzą zwykle od kształtów lub nazwisk osób, które je po raz pierwszy użyły i opublikowały wyniki. Przykładowe nazwy to: Daubechies, Haar, Coiflets, Symlet, Spline, Battle-Lemarie.

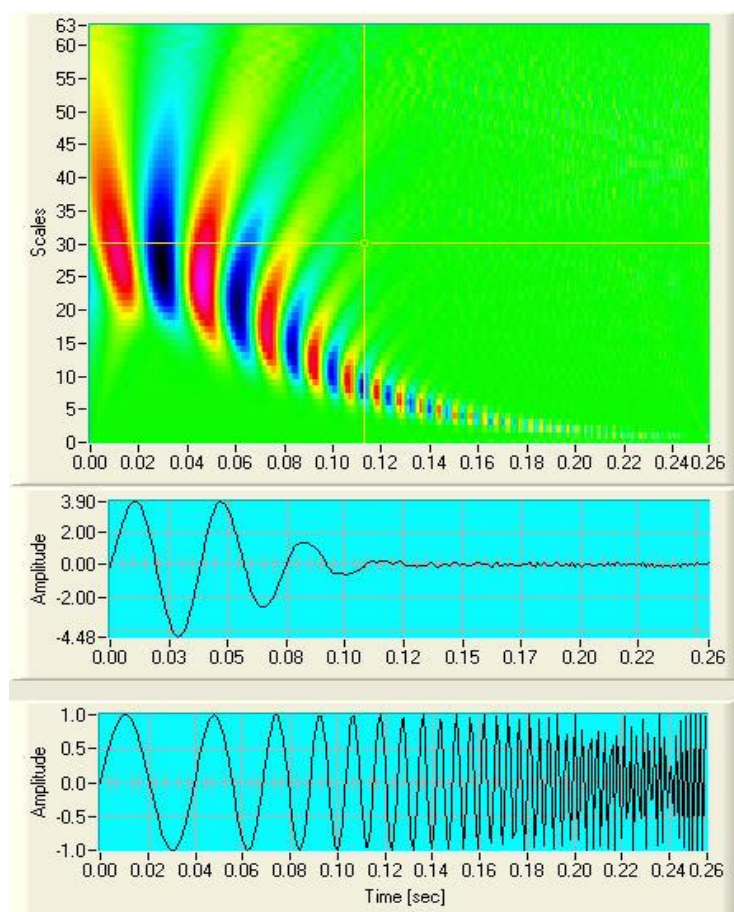
W zasadzie każda dyskusja o falkach może rozpoczynać się od funkcji Haara. Ta falka równoważna jest Daubechies (*db1*). Ingrid Daubechies wprowadziła rodzinę falek, które są często określane mianem *zbioru ortonormalnego o zwartym nośniku* [7].

Dla ustanowienia pełnego obrazu metody falkowej poniżej (rys.4.7 i 4.8) przedstawiono wyniki analizy falkowej (CWT) uzyskane z wykorzystaniem wirtualnego przyrządu pomiarowego [11]⁷. Znaczenie wykresów, idąc od dołu do góry, jest następujące: przebieg czasowy sygnału, zbiór współczynników w wybranej skali - zaznaczonej kursorem na wykresie górnym, rozkład falkowy na płaszczyźnie czas-skala (t/s). Obydwa prezentowane sygnały zostały wygenerowane w sposób sztuczny i były użyte poprzednio do zobrazowania właściwości STFT (rys.4.7 i 4.8). Konfrontacja obydwu par rysunków jest bardzo interesująca. W szczególności dotyczy to wzajemnej relacji między *częstotliwością i skalą*.

⁷ Przyrządy opracowane w Zakładzie Miernictwa Elektrycznego IETiME PW w ramach projektu bad.

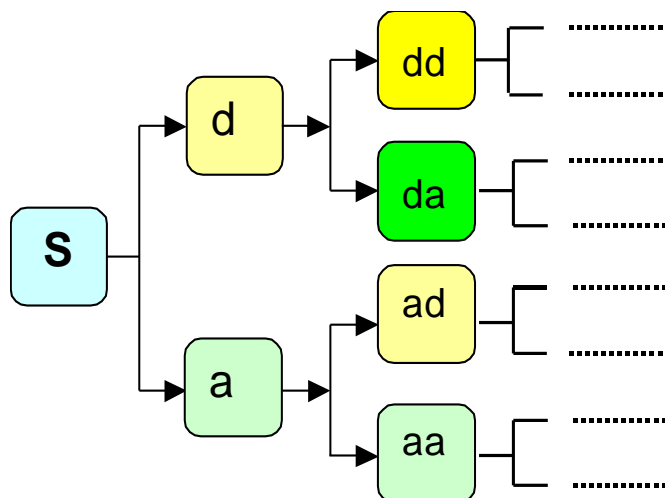


Rys. 4.7. Przykładowy wynik analizy falkowej (CWT): od dołu: przebieg czasowy sygnału, zbiór współczynników w wybranej skali - zaznaczonej kursorem, rozkład falkowy na płaszczyźnie czas-skala (t/s)



Rys. 4.8. Przykładowy wynik analizy falkowej (CWT): od dołu: przebieg czasowy sygnału, zbiór współczynników w wybranej skali - zaznaczonej kursorem, rozkład falkowy na płaszczyźnie czas-skala (t/s)

Na zakończenie opisu metody falkowej trzeba wspomnieć o istnieniu uogólnionej metody rozkładu falkowego, zwanej *pakietami falkowymi*, dostarczającej bogatszego zbioru składowych sygnału. W odróżnieniu od standardu w tej metodzie podlegają dalszemu rozkładowi również detale. Schemat drzewa dekompozycji pakietowej pokazano na rysunku 4.9.



Rys. 4.9. Drzewo falkowej dekompozycji pakietowej: S – sygnał oryginalny, d – detal, a – aproksymacja, dd – detal (detalu) w kolejnym rozkładzie, da – aproksymacja (detalu) w kolejnym rozkładzie, ad – detal (aproksymacji) w kolejnym rozkładzie, aa – aproksymacja (aproksymacji) w kolejnym rozkładzie

5. Metody kompresji sygnałów

Często się zdarza, że w procesie transmisji sygnałów pomiarowych, ze względu na ograniczoną przepustowość kanału transmisyjnego, zachodzi konieczność zastosowania kompresji danych. Wspólną cechą wszystkich metod kompresji danych jest redukcja *średniej bitowej* (*bit rate*, [bps], [bpp]) cyfrowej postaci danych w celu przesłania ich za pośrednictwem cyfrowego kanału komunikacyjnego lub zapamiętania w pamięci cyfrowej. Kompresja danych jest ściśle związana z teorią informacji, a w szczególności z jej działem znanym powszechnie pod nazwą teorii zniekształceń: *Rate-Distortion Theory*⁸[10].

Algorytmy kompresji danych dzieli się na dwie podstawowe grupy: *kompresji bezstratnej* i *kompresji stratnej*. W kompresji bezstratnej nie dopuszcza się żadnych strat informacji. W procesie kompresji stratnej dopuszcza się pewną utratę informacji, kontrolując jednocześnie jakość sygnału odtworzonego. Naturalnym sposobem oceny wierności rekonstrukcji jest rozważenie różnic pomiędzy sygnałem oryginalnym i zrekonstruowanym, czyli zniekształceń powstałych w wyniku zastosowania kompresji. Dla przypadku dyskretnego najbardziej popularną miarą zniekształceń jest błąd średniokwadratowy, definiowany jako średnia arytmetyczna z kwadratów różnic pomiędzy wartościami amplitud poszczególnych elementów ciągu.

Ważną grupę pośród metod kompresji danych stanowią metody służące do kompresji sygnałów. Termin „sygnał” w tym zastosowaniu dotyczy postaci zdyskretyzowanej w czasie oraz poddanej procesowi równomiernej kwantyzacji skalarnej, a więc postaci cyfrowej. Zakłada się przy tym, że na etapie wstępnego

⁸ Tłumaczenie polskie nie jest ścisłe. Określenie *rate* ma charakter wieloznaczny. W tym tekście można go utożsamić z pojęciem *średniej bitowej* (liczby bitów przypadającej na kwant sygnału lub jednostkę czasu). Funkcja RD ma kształt hiperboli.

przetwarzania sygnału było spełnione twierdzenie o próbkowaniu, zaś kwantyzacja w amplitudzie dokonana z odpowiednio dużą rozdzielczością.

Najbardziej znanym, akademickim przykładem kompresji sygnału pomiarowego może być metoda adaptacyjnego próbkowania sygnału, z kontrolą pochodnej. Kolejny przykład to nierównomierne próbkowanie z wykorzystaniem metody aproksymacyjnej za pomocą wielomianów. Obecnie należy się skoncentrować na metodach kompresji realizowanych w sposób cyfrowy. Ogólnie można przyjąć, że każda z wyżej wymienionych metod może być zrealizowana metodą eliminacji nadmiarowych próbek sygnału cyfrowego, uprzednio spróbkowanego i skwantowanego [22].

Dynamiczny rozwój techniki komputerowej doprowadził do stanu, gdy zakres metod przydatnych do praktycznej realizacji znacznie się poszerzył. Powstały metody hybrydowe łączące w sobie całe grupy sposobów kompresji. Zdaniem autorów najbardziej efektywnymi z punktu widzenia kompresji sygnałów są te metody hybrydowe (stratne), które określane są mianem *kwantyzacji wektorowej współczynników transformat*, łączące tradycyjne kodowanie transformacyjne z kwantyzacją wektorową i alokacją bitów.

Ujmując rzecz chronologicznie analizę metod transformacyjnych należałoby rozpocząć od metody kompresji określanej mianem kodowania transformacyjnego z kwantyzacją skalarną. Kodowanie transformacyjne jako rodzaj kodowania blokowego zaliczyć można do grupy metod quasi-wektorowych. Sam proces kwantyzacji tradycyjnie dokonywany jest oddzielnie dla każdego współczynnika transformaty, czyli na skalarach. Transformacja sprawia, że w ramach pojedynczego bloku danych wartość każdego współczynnika zależy od wszystkich danych wejściowych należących do tego samego bloku. Występuje efekt stopniowej, zależnej od rodzaju użytej transformaty, dekorelacji danych. Jednakże, mimo dobrych rezultatów uzyskiwanych przy kodowaniu transformacyjnym, ciągle brak jest ogólnej teorii dowodzącej, że dekorelująca transformacja jest najlepsza w sensie minimalizacji całkowitego błędu powstałego przy kodowaniu z zadaną efektywnością. Ponadto przetworzona w ten sposób informacja jest zwykle znacznie bardziej „upakowana” w wyniku koncentracji energii w obrębie fragmentu otrzymanego zbioru współczynników (własność niezmiernie cenna z punktu widzenia kompresji sygnału).

Biorąc pod uwagę jeden z fundamentalnych wniosków z teorii Shannona mówiący, że *„zawsze lepszą efektywność uzyska się kodując całe bloki - wektory niż pojedyncze próbki, skalary”* oraz fakt występowania bloków (wektorów) danych w przypadku transformat, naturalną konsekwencją stanowi próba połączenia obydwu wspomnianych wyżej metod, co w efekcie doprowadza do powstania wektorowej kwantyzacji współczynników transformaty.

Pozostaje kwestia doboru rodzaju transformaty. Największe znaczenie praktyczne mają: Dyskretna Transformata Fouriera (**DFT**) oraz Dyskretna Transformata Kosinusowa (**DCT**). Obydwie transformaty nie są optymalne w sensie dekorelacji danych, ale mają tę zaletę, że sposób ich obliczania nie zależy od danych wejściowych. Dzięki temu, przy obliczaniu odwrotnej transformaty nie są potrzebne żadne informacje pomocnicze. Drugą przyczyną jest bardziej prozaiczna, DFT jest najbardziej popularną transformatą w dziedzinie Cyfrowego Przetwarzania Sygnału (analiz widmowych w szczególności) oraz istnieje szybki algorytm do jej numerycznego wyznaczenia, który w sposób niemalże bezpośredni można użyć również do obliczeń DCT. Dyskretna Transformata Kosinusowa zawiera cechy transformacji w dziedzinę częstotliwości i jest równoważna DFT po dokonaniu symetryzującego rozszerzenia (podwojenia) zbioru próbek wejściowych. Ponadto ma ona tę przewagę, że daje wynik rzeczywisty. Stąd jej większa popularność w stosunku do DFT.

W dziedzinie kompresji sygnałów, podobnie jak w przypadku analizy czasowo-częstotliwościowej przoduje Dyskretna Transformata Falkowa (DWT). Jej zalety to: wysoka zdolność koncentracji energii (EPE), jednoczesna lokalizacja w czasie i częstotliwości, możliwość dopasowania kształtu falek do kształtu sygnału, zdolność analizy wielorozdzielczej.

Kwantyzacja wektorowa (**VQ**) jest metodą kodowania źródła, która umożliwia zastąpienie wektora (bloku) danych wejściowych za pomocą jednego spośród elementów specjalnego słownika. Prawdopodobnie w doborze właściwych elementów tego słownika stanowi fundamentalne zadanie w procesie projektowania kwantyzatorów wektorowych. Idea kwantyzacji wektorowej usprawiedliwia intuicyjne wysnucie wniosku, że sposób ten jest szczególnie efektywny w przypadku występowania zależności korelacyjnej między poszczególnymi składowymi wektorów. Jest to prawdą, lecz błędnym jest twierdzenie odwrotne jakoby kwantyzacja wektorowa nie wnosila nic w sensie efektywności w przypadku wektorów o składowych nieskorelowanych. Z teorii kodowania źródłowego Shannona wynika jednoznacznie, że kwantyzacja wektorowa ma przewagę nad skalarną zarówno w sensie jakości jak i efektywności nawet w tym przypadku. Za wybitny przykład liniowej transformaty dekorelującej dane może tu posłużyć transformata Karhunen - Loeve'go (KLT). Okazuje się, że zastosowanie kwantyzacji skalarnej w stosunku do współczynników tej transformaty nie jest metodą umożliwiającą osiągnięcie optimum, w sensie efektywności, jakie daje kwantyzacja wektorowa. Tym bardziej stwierdzenie to jest słuszne, w odniesieniu do transformat dających tylko „częściową dekorelację” danych.

5.1. Nierównomierna kwantyzacja skalarna

Ogólna teoria kwantyzacji skalarnej stanowi punkt wyjścia do opracowania metod kwantyzacji wektorowej [9]. Występuje tu istotna współzależność.

N - punktowy kwantyzator skalarny (jednowymiarowy) definiuje się jako odwzorowanie typu:

$$Q: \mathfrak{R} \rightarrow C \quad (5.1)$$

gdzie \mathfrak{R} stanowi nieskończony zbiór liczb rzeczywistych zaś C spełnia relację:

$$C \equiv \{c_1, c_2, c_3, \dots, c_N\} \subset \mathfrak{R}, \quad N < \infty \quad (5.2)$$

i stanowi słownik kwantyzatora o rozmiarze N ($|C| = N$). Elementy c_i tego słownika są odpowiednikami poziomów kwantyzacji.

Podstawowy parametr tego kwantyzatora, rozdzielczość: $r = \log_2 N$, specyfikuje średnią liczbę bitów niezbędnych do zakodowania wartości odpowiadających wszystkim poziomom kwantyzacji.

Konsekwencją rozmieszczenia na osi \mathfrak{R} poziomów kwantyzacji c_i jest powstanie partycji R_i , czyli przedziałów kwantyzacji, o krańcach $[x_{i-1}, x_i]$ zwanych często punktami granicznymi. Przy czym, zgodnie z definicją, R_i określa się jako:

$$R_i = \{x \in \mathfrak{R}: Q(x) = c_i\} \quad (5.3)$$

Ponadto: $\cup R_i = \mathfrak{R}$, oraz $R_i \cap R_j = \emptyset$, dla $i \neq j$. Zatem pełny opis kwantyzatora skalarnego daje równanie:

$$Q = \{c_i, R_i\}, \quad i = 1, 2, 3, \dots, N \quad (5.4)$$

Osobnego potraktowania w analizie teoretycznej wymagają krańcowe punkty skali przetwarzania: x_0 oraz x_N , określające krańcowe przedziały kwantyzacji: R_0 i R_N . Dla przypadku gdy $x_0 = -\infty$ oraz $x_N = \infty$ przedziały te noszą nazwę przepelnionych (w odróżnieniu od ziarnistych). W tym przypadku zakres pracy kwantyzatora definiuje się jako: $B = x_{N-1} - x_1$. Zwykle $x_0, x_N < \infty$, $B = x_N - x_0$.

Kwantyzator nazywamy regularnym, gdy spełniona jest nierówność:

$$x_0 < c_1 < x_1 < c_2 < \dots < c_N < x_N, \quad (5.5)$$

zaś równomiernym gdy ponadto: $(x_n - x_{n-1}) = \text{const.}$

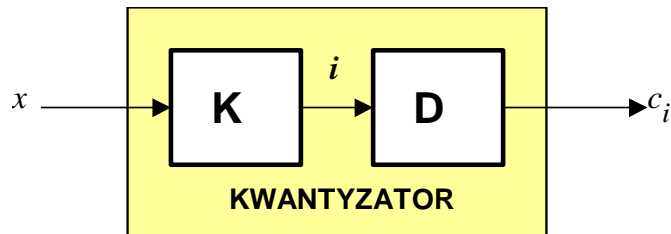
Przypadek $c_n = (x_{n-1} + x_n)/2$ występuje w technice przetwarzania analogowo-cyfrowego powszechnie stosowanego w metrologii i równoważny jest potocznemu rozumieniu kwantyzatora (w ogólności nieoptymalnego).

Zakładając bardzo dużą wartość rozdzielczości kwantyzatora (r) dla sygnału o zerowej składowej stałej (symetria zmienności) otrzymuje się następującą zależność na wartość stosunku sygnału do szumu [9]:

$$SNR \approx 6.02r + 10 \lg(3\beta_x^2) \quad (5.6)$$

Z ostatniego równania wynika „żelazna” reguła, że użycie dodatkowego bitu w procesie kwantyzacji powoduje przyrost stosunku sygnał/szum o ok. 6dB.

Problem konstrukcji optymalnych kwantyzatorów jest niezwykle zajmujący głównie z powodu ich „wyjątkowej” nieliniowości. Dekompozycja układu kwantyzatora na dwie części: koder i dekodek znacznie ułatwia analizę (rys.5.1).



Rys. 5.1 Schemat blokowy kwantyzatora (dekompozycja)

Koder realizuje nieliniową operację (odwzorowanie): $\mathbf{K}: \mathcal{R} \xrightarrow{na} \mathcal{I}$, zaś dekodek: $\mathbf{D}: \mathcal{I} \xrightarrow{na} \mathcal{C}$, gdzie $\mathcal{I} = \{1,2,3,\dots,N\}$ - zbiór indeksów, a $\mathcal{C} = \{c_i\}$ - słownik kwantyzatora. Oznacza to, że:

$$\mathbf{K}(x) = i, \quad \mathbf{D}(i) = c_i \quad \text{oraz} \quad Q(x) = c_i, \quad (5.7)$$

Symbolika ta odpowiada rozumieniu potocznemu procesu kwantyzacji, w którym i oznacza numer przedziału kwantyzacji.

Po przyjęciu założenia, że jeden spośród dwu elementów kwantyzatora ma parametry ustalone, stosunkowo łatwo podać zasady optymalizacji drugiego. Wynika stąd, że proces projektowania może być dokonany w trybie iteracyjnym: dobór optymalnego dekodekera dla kodera oraz dobór optymalnego kodera dla dekodekera. Obydwa procesy mogą być zdefiniowane w sposób rozłączny.

5.2. Kwantyzacja wektorowa

Wektor może być wykorzystany do opisu prawie każdego typu wzorca, jak np. segment sygnału mowy czy obrazu, przez proste uformowanie "wektora próbek" z przebiegu sygnału. Kwantyzacja wektorowa (VQ – Vector Quantization) może być widziana jako forma rozpoznawania struktur (wzorców) wejściowych i aproksymowania ich za pomocą jednego z uprzednio określonych wzorców. Zbiór tych wzorców jest określany mianem słownika.

Kwantyzacja wektorowa może być również postrzegana jako narzędzie do realizacji wielu skomplikowanych zadań z zakresu przetwarzania sygnałów, włączając klasyfikację i transformacje liniowe. W tego rodzaju zastosowaniach VQ może być traktowane jako technika redukcji złożoności, ponieważ redukcja w zakresie bitów może uprościć następne obliczenia numeryczne, pozwalając czasami na zastąpienie skomplikowanych algorytmów cyfrowego przetwarzania sygnałów prostym przeszukiwaniem tablic. Zatem kwantyzacja wektorowa jest czymś daleko więcej niż formalnym uogólnieniem kwantyzacji skalarnej. W ciągu ostatnich kilku lat stała się ona ważną techniką z dziedziny kompresji oraz klasyfikacji sygnałów, zaś jej przydatność i zastosowania ciągle rosną. Z coraz to większym powodzeniem wykorzystywana jest do „odszumiania” sygnałów.

Kwantyzator wektorowy, odwzorowuje wektory należące do k -wymiarowej przestrzeni euklidesowej \mathfrak{R}^k w skończony zbiór C zawierający N elementów wyjściowych (odwzorowań) zwanych wektorami kodowymi. Tak więc:

$$Q: \mathfrak{R}^k \xrightarrow{na} C \quad (5.8)$$

gdzie: $C = \{c_1, c_2, c_3, \dots, c_N\}$, $c_i \in \mathfrak{R}^k$, $i=2, 3, \dots, N$.

Zbiór C nazywany jest słownikiem o rozmiarze N zawierającym N różnych elementów będących wektorami z przestrzeni \mathfrak{R}^k . Rozdzielczość kwantyzatora wektorowego $r = (\log_2 N)/k$ jest miarą liczby bitów przypadających na wektor kodowy użyty do reprezentowania wektora wejściowego, a także wskaźnikiem możliwej do osiągnięcia dokładności. Przy ustalonej wartości wymiaru kwantyzatora jego rozdzielczość jest jednoznacznie określona rozmiarem słownika N , a nie liczbą bitów użytych do specyfikacji numerycznej wektorów kodowych przechowywanych w słowniku. W typowych rozwiązaniach, słownik stanowi tablicę danych przechowywaną w pamięci komputera, zaś liczba bitów precyzji użytych do zapisu poszczególnych składowych każdego wektora nie ma wpływu na rozdzielczość kwantyzatora, a dotyczy jedynie rozmiaru zajętej pamięci oraz problemu adekwatnej precyzji zapisu słownika.

Z każdym k -wymiarowym kwantyzatorem wektorowym ze słownikiem N -wymiarowym jest związany podział przestrzeni \mathfrak{R}^k na N , k -wymiarowych obszarów (przedziałów kwantyzacji) R_i , ($i=1, 2, 3, \dots, N$). Każdy przedział jest opisany równaniem:

$$R_i = \{ \mathbf{x} \in \mathfrak{R}^k: Q(\mathbf{x}) = c_i \} \quad (5.9)$$

Z definicji przedziałów kwantyzacji wynika, że:

$$\cup R_i = \mathfrak{R}^k \text{ oraz } R_i \cap R_j = \emptyset \text{ dla } i \neq j \quad (5.10)$$

Rozróżnia się przedziały otwarte (brzegowe, znajdujące się na krańcach przestrzeni) i zamknięte (ziarniste), stanowiące zamknięty obszar k -wymiarowy.

Ważną własnością zbioru przedziałów przestrzeni \mathfrak{R}^k jest wypukłość. Zbiór S należący do tej przestrzeni jest wypukły, jeżeli dla każdego dwu elementów tego zbioru np. A i B zachodzi zależność: $\alpha A + (1-\alpha)B \in S$ dla $0 < \alpha < 1$.

Kolejną własność kwantyzatorów to regularność. Kwantyzator wektorowy nazywamy regularnym jeżeli każdy przedział R_i stanowi zbiór wypukły oraz $c_i \in R_i$ dla każdego i .

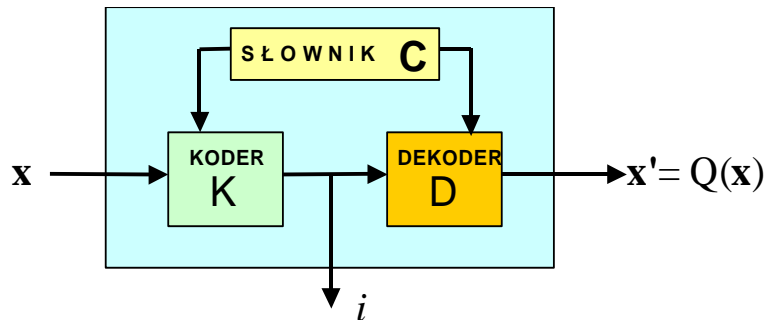
Koder kwantyzatora realizuje zadanie wyselekcjonowania spośród skończonego zbioru (słownika) jednego wektora c_i , który ma reprezentować bieżący wektor wejściowy \mathbf{x} . Indeks i zapisany w postaci binarnej, podobnie jak w przypadku kodera skalarnego, stanowi jedyną informację niezbędną dla dekodera w celu odszukania w słowniku wektora odtworzonego. Dla ułatwienia rozważań analitycznych najwygodniejszy, zdaniem autorów,

jest sposób polegający na dekompozycji strukturalnej kwantyzatora wektorowego na dwa bloki składowe: koder i dekoder, tak jak w przypadku skalarnym.

Koder K odwzorowuje przestrzeń \mathfrak{R}^k w zbiór I symboli $i = 1, \dots, N$, zaś dekoder D odwzorowuje zbiór I w zbiór odtworzeniowy $\{C_i\}$. Proces ten można opisać za pomocą zależności:

$$Q(\mathbf{x}) = D[K(\mathbf{x})] \quad (5.11)$$

Właściwie, często wygodnie jest rozpatrywać kwantyzator wektorowy jako układ dostarczający zarówno indeks i jak i skwantowaną wielkość wyjściową $Q(\mathbf{x})$. Rysunek 5.2 ilustruje zasadę kaskadowego połączenia kodera z dekodernem, definiując w ten sposób kwantyzator wektorowy:



Rys. 5.2 Schemat kwantyzatora wektorowego

Zadaniem kodera będzie przydzielenie każdego wektora wejściowego do jednego z przedziałów k -wymiarowej przestrzeni \mathfrak{R}^k , na zasadzie określenia jego numeru porządkowego i . Zadaniem dekodera, generacja wektora kodowego c_i będącego jej reprezentantem.

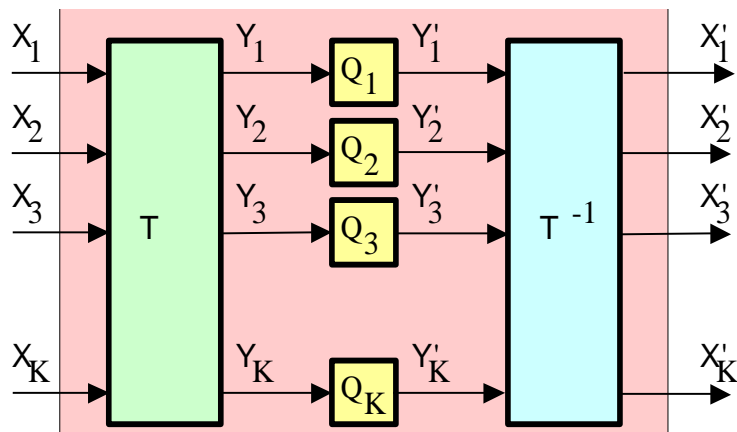
Bardzo ważną rodzinę kwantyzatorów wektorowych, będącą wręcz synonimem kwantyzacji, stanowią układy wykorzystujące regułę najbliższego sąsiada, znaną z opisu kwantyzacji skalarniej. Cechą znamioną jest tutaj zależność zasady podziału przestrzeni \mathfrak{R}^k na przedziały, wyłącznie od słownika oraz użytej miary błędu. Nie występuje konieczność geometrycznego opisu przedziałów w sposób jawny.

5.3. Kodowanie transformacyjne

Kodowanie transformacyjne jest rodzajem kodowania blokowego i z założenia stanowi proces dwustopniowy. Niech \mathbf{X} oznacza wektor próbek sygnału: $\mathbf{X} = (X_1, X_2, \dots, X_N)^t$. Zasada kodowania transformacyjnego polega na poddaniu wektora \mathbf{X} odpowiedniej transformacji liniowej T , w wyniku której powstanie nowy wektor \mathbf{Y} , zwykle o tym samym rozmiarze zwany wektorem współczynników transformaty (próbek transformaty): $\mathbf{Y} = T\mathbf{X}$. Standardowo współczynniki transformaty są kwantowane w sposób niezależny, czyli skalarnie. Istotne własności wektora \mathbf{Y} , ze względu na kompresję sygnału, to:

- Występowanie efektu „wymieszania informacji” zawartych w próbkach wejściowych - każda próbka transformaty zawiera część informacji zawartej we wszystkich próbkach bloku wejściowego.
- Zmniejszona korelacja wzajemna współczynników w porównaniu z korelacją oryginalnych próbek sygnału: możliwość zastosowania alokacji bitów i jak się później okaże kwantyzacji wektorowej z kodem składanym: PCVQ.
- Występowanie efektu „upakowania”, tzn. koncentracja energii w obrębie ograniczonej liczby próbek transformaty $K < N$ (możliwość eliminacji próbek zerowych i mniej znaczących).

Pełny schemat funkcjonalny układu kodowania transformacyjnego z kwantyzacją skalarną przedstawiono na rys.5.3: (symbole Q_k oznaczają kwantyzatory skalarne o różnej liczbie poziomów kwantyzacji, T- kwadratową macierz $K \times K$ realizującą transformację liniową).



Rys. 5.3 Kodowanie transformacyjne z kwantyzacją skalarną

Jak widać, operacja kwantyzacji jest dokonywana oddzielnie dla każdego współczynnika (próbki) transformaty:

$$Y'_k = Q(Y_k) \quad (5.12)$$

Odtworzoną postać X' oryginalnego wektora X otrzymuje się dokonując odwrotnego przekształcenia skwantowanej postaci Y' :

$$X' = T^{-1}(Y') \quad (5.13)$$

Najbardziej wygodną miarą błędu kodowania, którą można zastosować w tym systemie, jest uśredniona statystycznie wartość błędu średniokwadratowego:

$$D_{TC} = E[\|X - X'\|^2] = \sum_{i=1}^K E[|X_i - X'_i|^2] \quad (5.14)$$

Stopień koncentracji energii w dziedzinie transformaty opisywany jest za pomocą tzw. współczynnika efektywności upakowania energii: EPE (Energy Packing Efficiency). Definiuje się go jako stosunek energii zawartej w K pierwszych współczynnikach do energii całkowitej (N współczynników):

$$EPE(K) = \frac{\sum_{k=0}^{K-1} E[Y_k^2]}{\sum_{k=0}^{N-1} E[Y_k^2]} \quad (5.15)$$

Do transformat dotychczas najbardziej popularnych w dziedzinie kompresji sygnału oprócz KLT należą Dyskretna Transformata Fouriera i Dyskretna Transformata Kosinusowa.

W podsumowaniu rozważań dotyczących kodowania transformacyjnego warto podkreślić, iż:

- Badania o charakterze subiektywnym wykazały, że zarówno ucho jak i oko ludzkie charakteryzują się różnymi wrażliwościami na amplitudy poszczególnych współczynników transformat (SNR dla poszczególnych współczynników jest parametrem ważniejszym niż wartość globalnego SNR);

- Operację transformaty opisuje zależność: $\mathbf{Y}=\mathbf{TX}$, gdzie $\mathbf{X}=(X_1, X_2, X_3, \dots, X_K)^t$ jest wektorem transformowanym zaś \mathbf{T} operatorem liniowym;
- Każdy współczynnik transformaty jest zależny od wszystkich składowych wektora transformowanego;
- Współczynniki transformaty są mniej skorelowane i wymagają mniej kompresji dodatkowej;
- Współczynniki transformaty są bardziej upakowane, niektóre można zaniedbać;

Uwaga: Transformata odwrotna może spotęgować błąd kwantyzacji, aby tego uniknąć należy ograniczyć się do transformat ortogonalnych. Przy transformacji ortogonalnej odległość między dwoma punktami pozostaje niezmienna.

5.4. Alokacja bitów

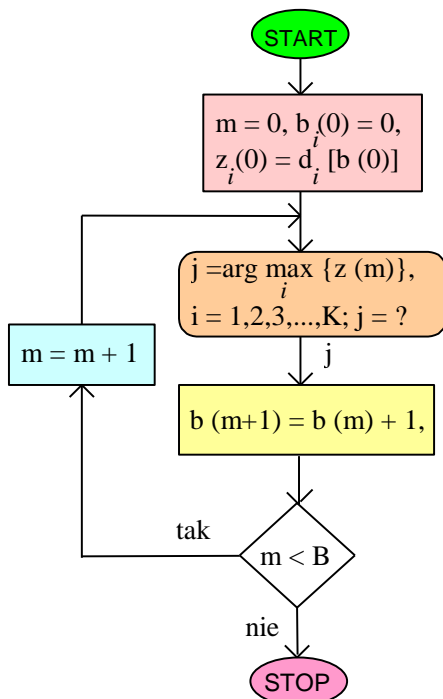
Niech $d_i(b_i)$ oznacza lokalną wartość błędu średniokwadratowego kwantyzacji zmiennej X_i , za pomocą b_i bitów. Wtedy wartość błędu globalnego kwantyzacji bloku wyniesie:

$$D = D(\mathbf{b}) = \sum_{i=1}^K d_i(b_i) \quad (5.16)$$

Zadanie optymalnej alokacji bitów polega na określeniu wszystkich wartości b_i minimalizujących wartość globalnej miary błędu D tak, aby spełniały nierówność:

$$\sum_{i=1}^K b_i \leq B \quad (5.17)$$

Teoretyczne rozważania dotyczące problemu alokacji bitów są tematem szeregu publikacji, w większości których przyjmuje się coraz bardziej rygorystyczne założenia co do parametrów statystycznych sygnału. Zdaniem autorów najprostszym sposobem jest realizacja alokacji bitów w sposób przyrostowy - opcjonalnego dodawania jednego bitu w każdym kroku spośród B iteracji (rys.5.4).



Krok 1: Założenia początkowe

Krok 2: Określenie pozycji bloku w największej potrzebie

Krok 3: Przydzielenie dodatkowych bitów pozycji wybranej w kroku 2

Krok 4: Sprawdzenie czy są jeszcze wolne bity

Rys. 5.4 Algorytm alokacji bitów na potrzeby kwantyzacji wektorowej

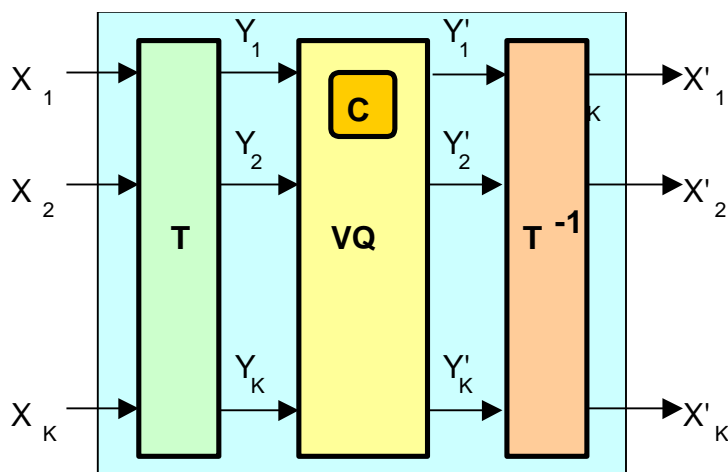
Przyjęcie założenia o dużej rozdzielczości kwantyzacji jak zwykle ułatwia obliczenia - wtedy bowiem bieżąca wartość błędu $d_i(m)$ jest proporcjonalna do wariancji sygnału na i -tej pozycji ze współczynnikiem 2^{-2b} . Wynika stąd, że każdy dodatkowy bit redukuje błąd czterokrotnie.

Praktyczna realizacja algorytmu, szczęśliwie, może być dokonana z użyciem blokowej sekwencji treningowej. Każdy krok iteracji wymaga wówczas wyznaczenia wartości globalnego błędu kwantyzacji (dla całej sekwencji). Błędy przechowywane w tablicy determinują funkcję błędu $d_i(m)$.

5.5. Kwantyzacja wektorowa współczynników transformat

Zastosowanie kwantyzacji wektorowej w odniesieniu do współczynników transformaty jest naturalną konsekwencją ich blokowej struktury. Jednakże, bardziej szczegółowa analiza dostarcza informacji o wielu dalszych, istotnych zaletach, jakie rodzi połączenie tych dwu metod kompresji sygnału.

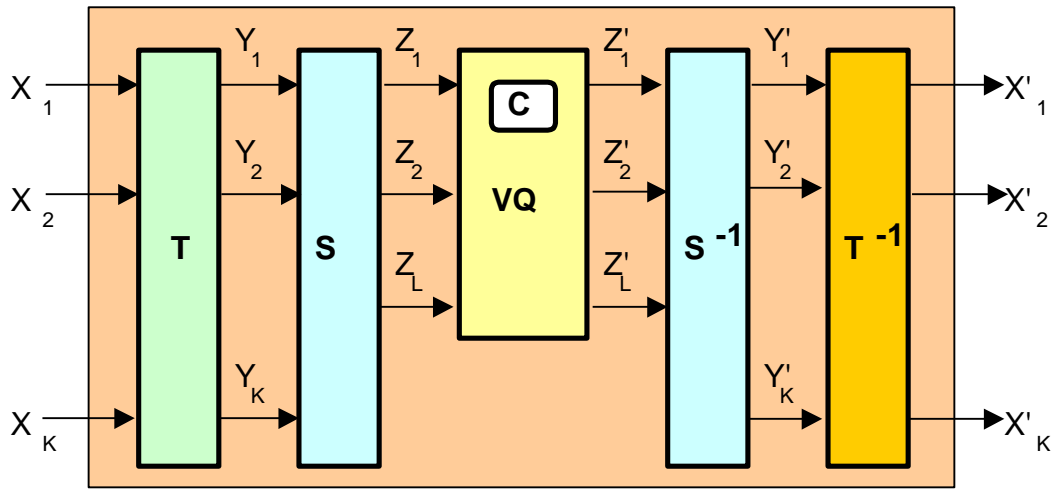
Schemat blokowy najprostszego systemu kwantyzacji wektorowej współczynników transformaty przedstawiono na rys. 5.5.



Rys. 5.5 Schemat blokowy kwantyzacji wektorowej współczynników transformaty

Blok danych tworzący wektor \mathbf{X} jest poddawany transformacji liniowej T . Wektor wynikowy \mathbf{Y} (współczynników transformaty) jest kwantowany wektorowo w bloku VQ . Skwantowane wektorowo współczynniki \mathbf{Y}' są poddawane transformacji odwrotnej T^{-1} dając w wyniku wektor odtworzony \mathbf{X}' .

Jak wspomniano wcześniej transformacja liniowa „upakuje” energię w obrębie części składowych wektora wynikowego. W pierwszym przybliżeniu pewna liczba współczynników transformaty może zostać pominięta, co powoduje ograniczenie rozmiarów wektora i tym samym upraszcza kwantyzację. W przypadku odrzucenia składowych wektora $\mathbf{X} = \{X_i; i = 1,2,3,\dots,K\}$ o znikomej zawartości energetycznej (selekcja) otrzymuje się wektor \mathbf{Z} o zmniejszonym rozmiarze, powiedzmy L . Kwantyzacji wektorowej poddawany jest wektor \mathbf{Z} , a w jej wyniku powstaje wektor \mathbf{Z}' . Wektor ten uzupełniony zerami do rozmiaru oryginalnego ($K > L$) jest następnie poddawany transformacji odwrotnej. Dodatkową zaletą omawianej metody są oszczędności w rozmiarze słownika, za który może posłużyć słownik oryginalny po odrzuceniu stosownych fragmentów każdego wektora kodowego. Schemat blokowy tego systemu przedstawia rys. 5.6.



Rys. 5.6 Schemat blokowy kwantyzacji wektorowej z redukcją rozmiarów wektora

Można łatwo wykazać, że:

$$\|\mathbf{X} - \mathbf{X}'\|^2 = \|\mathbf{Z} - \mathbf{Z}'\|^2 + \|\mathbf{Y} - \mathbf{S}^{-1} \mathbf{S} \mathbf{Y}'\|^2 \quad (5.18)$$

zaś średnia wartość błędu kwantyzacji wynosi:

$$D = E[\|\mathbf{Z} - \mathbf{Z}'\|^2] + \sum_{i=L+1}^K E[Y_i^2] \quad (5.19)$$

Wprowadzony błąd dodatkowy jest równoważny energii odrzuconych próbek widma.

Jak już wspomniano przy okazji omawiania standardowej kwantyzacji wektorowej FSVQ, w przypadku gdy w dziedzinie transformaty wektor wynikowy ma rozmiar zbyt duży do przeprowadzenia bezpośredniej kwantyzacji, z uwagi na absurdalny rozmiar słownika, składowe tego wektora dzieli się na podwektory przeznaczone do kwantyzacji z użyciem indywidualnych słowników. Globalne słowo kodowe stanowi efekt złożenia cząstkowych słów kodowych - stąd nazwa metody: kwantyzacja wektorowa z kodem składanym (PCVQ - Product Code Vector Quantization). Z kolei globalny słownik wynikowy stanowi iloczyn kartezyjski słowników cząstkowych.

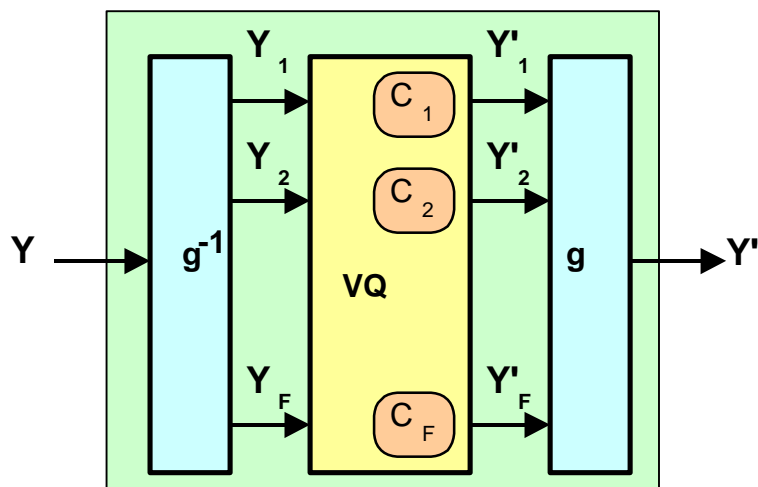
Niech dana będzie funkcja $g(\mathbf{Y})$ służąca do powrotnego złożenia wektora \mathbf{Y} z wektorów cząstkowych: $\mathbf{Y} = g(\mathbf{Y}_1, \mathbf{Y}_2, \mathbf{Y}_3, \dots, \mathbf{Y}_F)$. Dla każdego podwektora \mathbf{Y}_i , dany jest N_i -wymiarowy słownik C_i . Kwantyzator generuje zbiór indeksów $\{i_i\}$, służących do późniejszego odtworzenia wektorów wynikowych $\{\mathbf{Y}'_i\}$, gdzie $\mathbf{Y}'_i \in C_i$, dla $i = 1, 2, 3, \dots, F$. W efekcie globalna postać wektora odtworzonego \mathbf{Y}' zostaje utworzona na podstawie globalnego słownika wynikowego. Postać jego jest równoważna iloczynowi kartezyjskiemu słowników składowych:

$$\mathbf{C} = \mathbf{C}_1 \times \mathbf{C}_2 \times \mathbf{C}_3 \times \dots \times \mathbf{C}_F \quad (5.20)$$

Odtworzony wektor \mathbf{Y}' powstanie po ponownym złożeniu podwektorów \mathbf{Y}_i . Zbiór wszystkich wektorów kodowych dostępnych dla odtworzenia wektora \mathbf{Y}' ma licznosc N równoważną efektywnemu rozmiarowi słownika globalnego \mathbf{C} :

$$N = \prod_{i=1}^F N_i \quad (5.21)$$

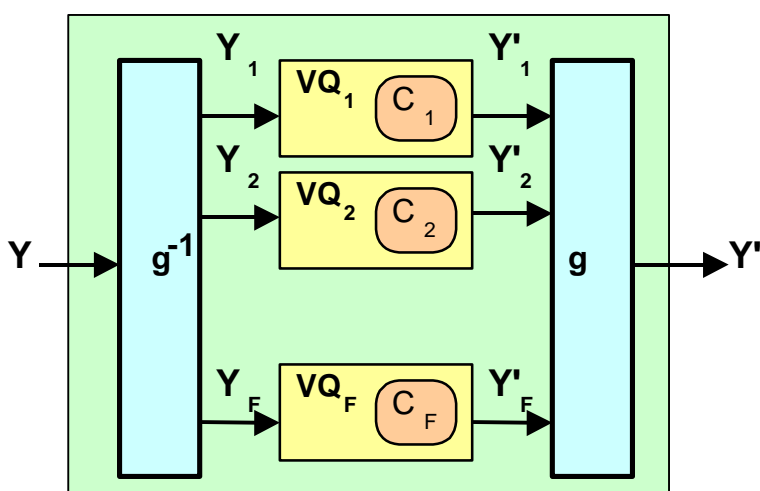
Słownik składany nie spełnia warunku optymalności, gdyż w końcowym etapie tworzony jest na podstawie zależności zdefiniowanej *explicitie*. Ogólna struktura systemu kwantyzacji wektorowej z kodem składanym jest przedstawiona na rysunku 5.7 (*symbolami C_i oznaczone są słowniki cząstkowe*).



Rys. 5.7 Ogólna struktura systemu kwantyzacji wektorowej z kodem składanym

Jak wynika ze schematu, składowe wektory odtworzone Y_i , nie muszą być wybierane niezależnie (indywidualnie) lecz mogą być związane z doбором sąsiadów w taki sposób, aby optymalizować minimum błędu globalnego odtworzenia wektora Y' .

Jak zwykle praktyka narzuca konieczność stosowania struktur nieco bardziej uproszczonych jak np. struktura z niezależną kwantyzacją wektorową składowych w oparciu o indywidualne słowniki (Rys.5.8).



Rys. 5.8 Struktura kwantyzatora wektorowego z kodem składanym i niezależną kwantyzacją

Teoretycznie struktura ta oddala się jeszcze bardziej od konfiguracji optymalnej, ale charakteryzuje się znacznie mniejszą złożonością realizacyjną. Szczególny przypadek tej struktury, wydaje się najprostszy i daje podział wektora globalnego Y na podwektory metodą sekwencyjną (w sposób geometryczny), co można zapisać jako: $Y = (Y_1 Y_2 Y_3 \dots Y_F)$.

Dla średniokwadratowej miary błędu prawdziwa jest zależność:

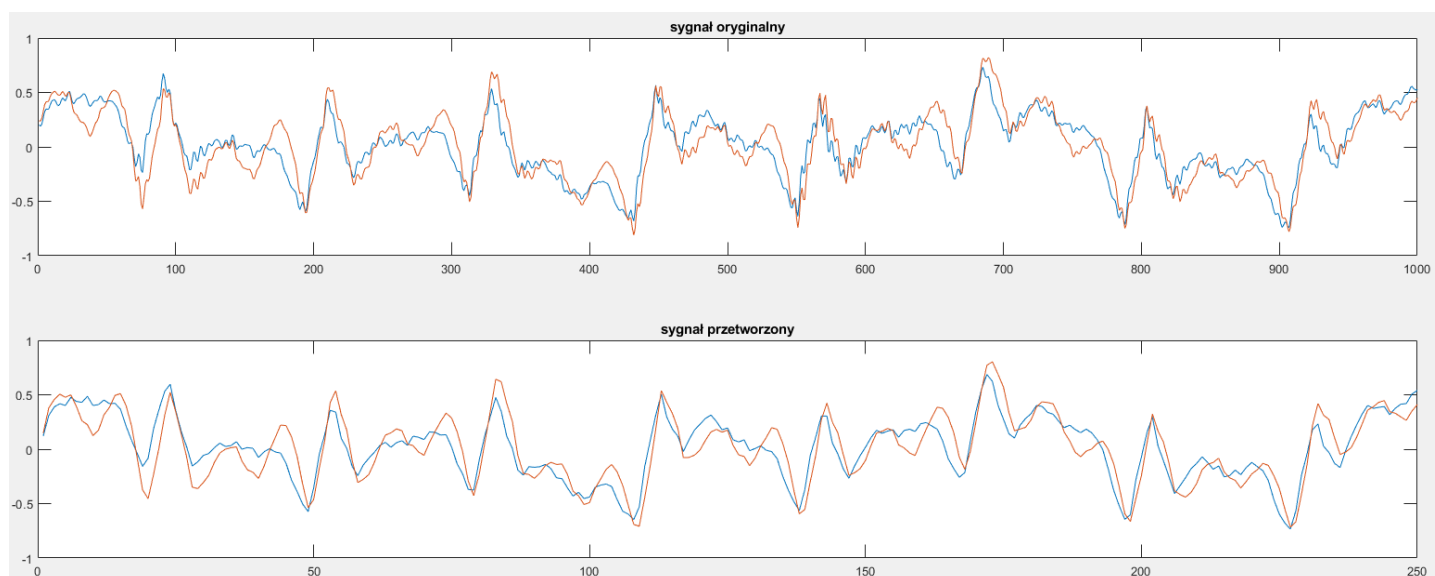
$$\|Y - Y'\|^2 = \|Y_1 - Y'_1\|^2 + \|Y_2 - Y'_2\|^2 + \|Y_3 - Y'_3\|^2 + \dots + \|Y_F - Y'_F\|^2 \quad (5.22)$$

6. Ćwiczenia do modułu (rozwiązane problemy praktyczne - zadania, projekty)

Celem ćwiczeń jest zapoznanie studenta z podstawowymi metodami cyfrowego przetwarzania sygnałów. Program ćwiczeń obejmuje: elementy analizy częstotliwościowej (algorytm dyskretnej transformaty Fouriera), podstawy analizy czasowo-częstotliwościowej (algorytm krótkoczasowej analizy Fouriera), analizę falkową sygnałów, badanie filtrów cyfrowych o skończonej i nieskończonej odpowiedzi impulsowej.

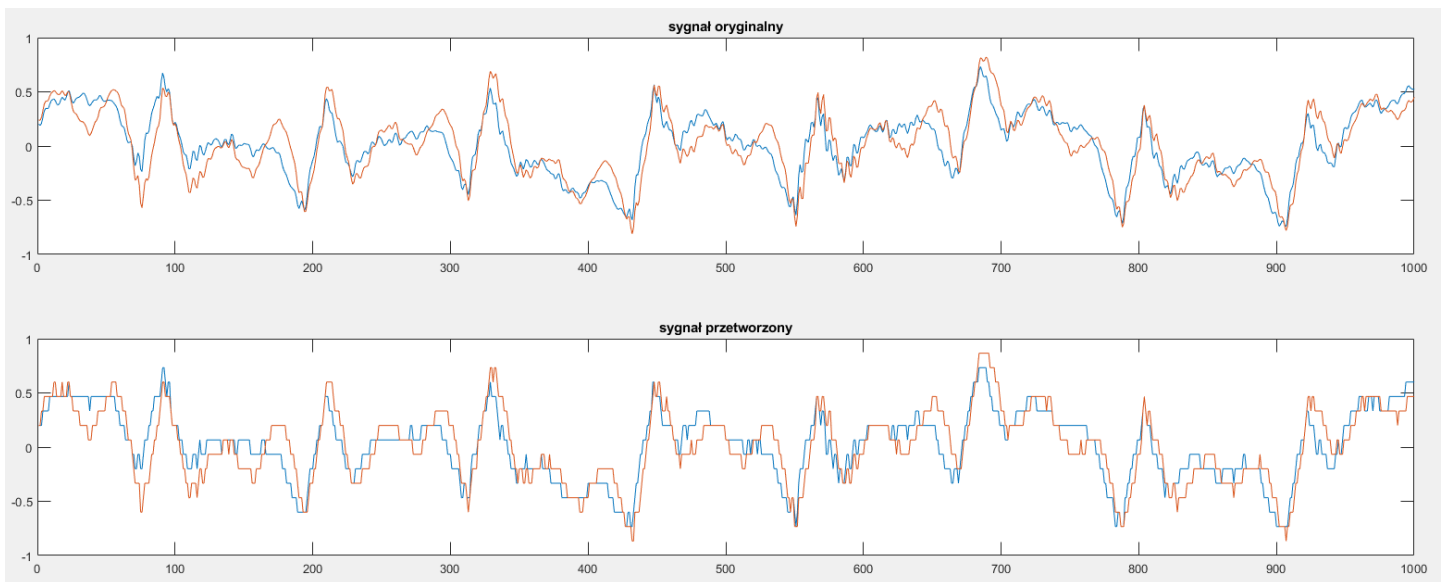
6.1. Próbkowanie i kwantyzacja sygnału

Zamiana sygnału analogowego na cyfrowy jest związana ze stratą części informacji. Wiąże się to z koniecznością przeprowadzenia próbkowania i kwantyzacji sygnału. Obie te operacje muszą być tak dobrane, żeby zachowane były potrzebne informacje w sygnale cyfrowym. Przyjmuje się zakres częstotliwości sygnału cyfrowego od zera do połowy częstotliwości próbkowania (częstotliwości Nyquista). Kwantyzacja z kolei warunkuje dokładność zapisu amplitudy próbek. Na rys. 6.1 zilustrowany jest wpływ częstotliwości próbkowania na jakość sygnału. Sygnałem oryginalnym jest sygnał dźwiękowy próbkowany z częstotliwością 44100 Hz i skwantowany z rozdzielczością 16 bitów. Na dole rys. 6.1 przedstawiony jest ten sam sygnał, ale próbkowany czterokrotnie wolniej (z częstotliwością 11025 Hz). W tym samym oknie czasowym mieści się czterokrotnie mniej próbek (1000 próbek w sygnale oryginalnym i 250 próbek w sygnale przetworzonym). Widać też wyraźnie, że wysokie częstotliwości zostały utracone (sygnał jest bardziej „gładki”).



Rys. 6.1. Sygnał dźwiękowy oryginalny i próbkowany czterokrotnie wolniej

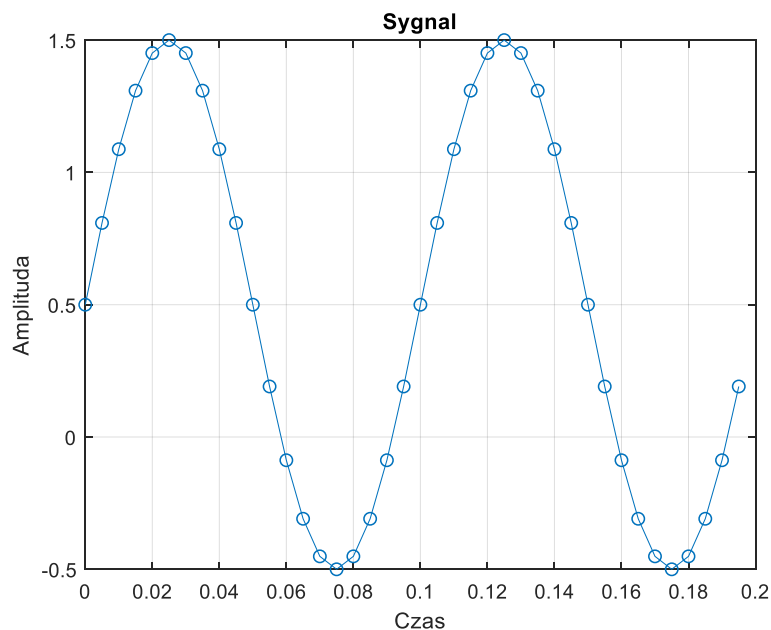
Z kolei na rys. 6.2 przedstawiono te same sygnały, ale kwantowane z różną rozdzielczością. Sygnał oryginalny jest zapisany z wykorzystaniem 16 bitów, a sygnał przetworzony 4 bitów. Oczywiście w przypadku dźwięku 4 bity nie są wystarczające do zachowania dobrej jakości dźwięku. Chodzi tu o pokazanie zniekształceń amplitudy. Przebieg po skwantowaniu z rozdzielczością 4 bitów ma wyraźnie schodkowy charakter. Częstotliwość próbkowania w tym przypadku pozostała niezmienną i wynosiła 44100 Hz.



Rys. 6.2. Sygnał dźwiękowy kwantowany z wykorzystaniem 16 bitów (góra) i 4 bitów (dół)

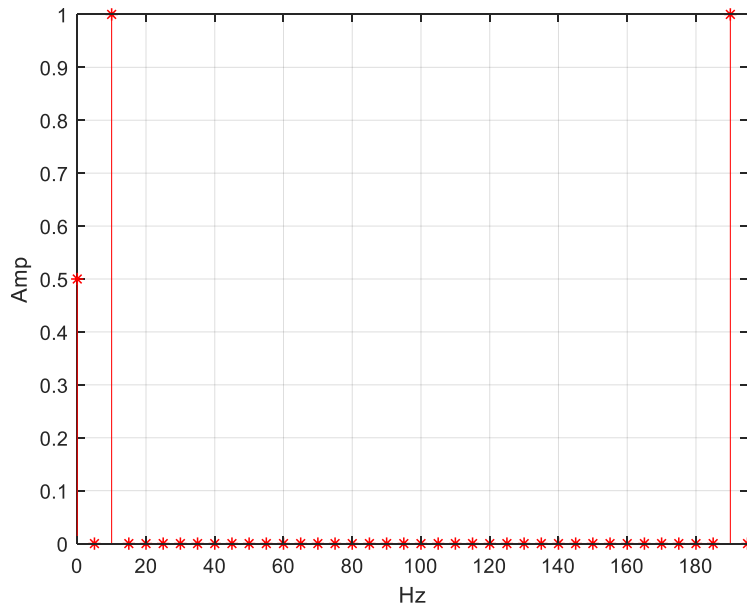
6.2. Analiza częstotliwościowa – szybka transformata Fouriera

Szybka transformata Fouriera umożliwia policzenie charakterystyk widmowych sygnału określonego w czasie. Zwykle bardziej interesuje nas amplitudowa charakterystyka widmowa. W pierwszym ćwiczeniu zostanie policzona charakterystyka amplitudowa sygnału sinusoidalnego. Najpierw należy wygenerować sygnał sinusoidalny jak na rys. 6.3 (amplituda 1, faza 0, częstotliwość sygnału 10Hz, czas trwania 0.2 s, częstotliwość próbkowania 200 Hz). Odpowiedni skrypt Matlaba jest zaprezentowany na Listingu 5.1.



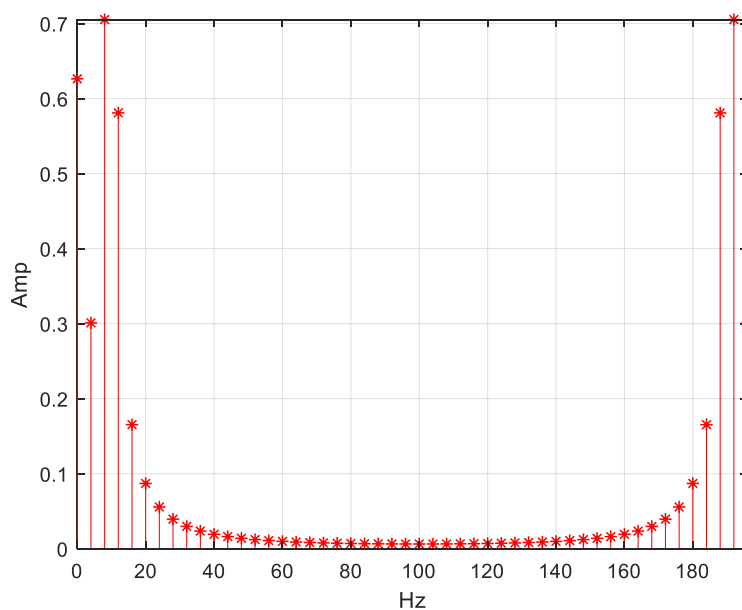
Rys. 6.3. Sygnał sinusoidalny

Następnie, korzystając z szybkiej transformaty Fouriera, policzymy widmo amplitudowe tego sygnału stosując okno prostokątne (rys. 6.4).



Rys. 6.4. Widmo amplitudowe sygnału sinusoidalnego – analiza synchroniczna

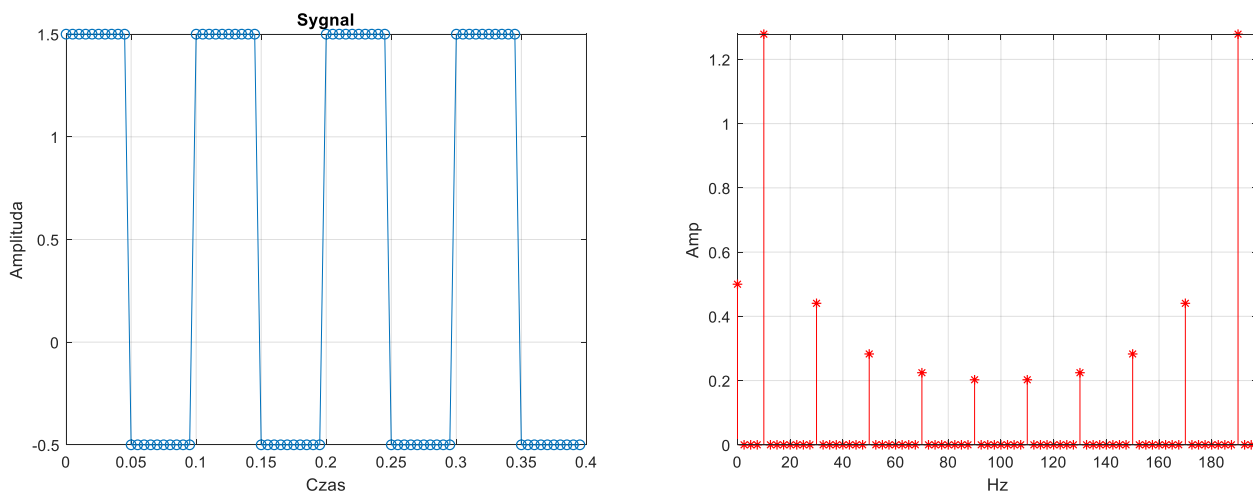
Prążek w częstotliwości 0 Hz odpowiada wartości składowej stałej. Prążek w 10 Hz jest składową sinusoidalną o częstotliwości 10 Hz i amplitudzie 1 (wysokość prążka). Częstotliwość próbkowania jest równa 200 Hz. Zatem zakres analizy jest równy połowie częstotliwości próbkowania i wynosi 100 Hz. Prążek w 190 Hz jest odbiciem lustrzanym widma względem częstotliwości Nyquista (100 Hz w tym przypadku) i nie niesie żadnej dodatkowej informacji. W tym przypadku mamy do czynienia z analizą synchroniczną, dla której parametry są tak dobrane, że analiza obejmuje całkowitą liczbę okresów sygnału badanego. Jest to szczególny przypadek, który trudno jest uzyskać w warunkach rzeczywistych. Częstszym przypadkiem jest tzw. analiza asynchroniczna (rys. 6.5).



Rys. 6.5. Widmo amplitudowe sygnału sinusoidalnego – analiza asynchroniczna

Wyraźnie widać, że w widmie występuje dużo więcej prążków niż wynikałoby to z teorii. Jest to wynikiem tzw. przecieku widma. W tym przypadku rozdzielczość analizy wynosi 4 Hz, zatem punkty, w których obliczane są prążki widma to 0, 4, 8, 12, 16 ... Hz. Nie można wystawić prążka dla 10 Hz. Prążki „rozpływają się” na sąsiednie. Amplituda najwyższego prążka jest równa 0.7 (a nie 1, jak było dla analizy synchronicznej). Widmo jest trudniej zinterpretować. Taka analiza jest najczęściej spotykana w warunkach rzeczywistych.

W drugim ćwiczeniu policzymy widmo amplitudowe sygnału prostokątnego. Dla analizy synchronicznej otrzymujemy przebiegi jak na rys. 6.6. Wyraźnie widać kolejne nieparzyste harmoniczne (10, 30, 50, 70, 90 Hz) wchodzące w skład sygnału prostokątnego.



Rys. 6.6. Sygnał prostokątny i jego widmo amplitudowe – analiza synchroniczna

Proszę sprawdzić jak wygląda widmo w przypadku analizy asynchronicznej. Jeśli chcemy zwiększyć zakres analizy widmowej wystarczy zwiększyć częstotliwość próbkowania, jeśli chcemy zwiększyć rozdzielczość analizy należy wydłużyć czas analizy. Proszę sprawdzić jak to działa w praktyce wykorzystując załączony skrypt Matlaba.

Skrypt Matlaba obliczający widmo sygnałów sinusoidalnego i prostokątnego jest zamieszczony na Listingu 6.1.

```
clear;
clc;
close all;

%częstotliwość próbkowania
Fs=200;

%parametry sygnału
fsyg=10;
ok=1/fsyg;
A=1;
faza=0;
sks=0.5;

T=1/Fs;
%czas analizy sygnału
t=0:T:(4*ok)-T;
```



```

y=A*sin(2*pi*fsyg*t+faza)+sks;
y=A*square(2*pi*fsyg*t+faza)+sks;

figure
plot(t,y,'-o');
grid on;

xlabel('Czas')
ylabel('Amplituda')
title('Sygnał')

%liczba punktów sygnału i widma
N=length(y);

%obliczenie widma sygnału
widmo=abs(fft(y));
widmo(1)=widmo(1)/N;
widmo(2:end)=widmo(2:end)./(N/2);

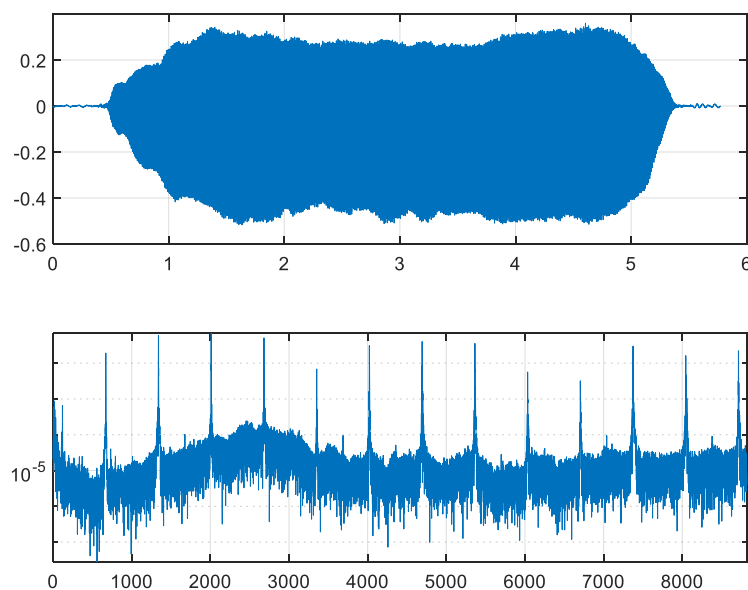
dFs=Fs/N;
f=0:dFs:Fs-dFs;

figure
stem(f,widmo,'-*r')
axis tight
xlabel('Hz');
ylabel('Amplituda')
title('Widmo amplitudowe')

```

Listing 6.1. Skrypt liczący widmo sygnałów sinusoidalnego i prostokątnego

W trzecim ćwiczeniu przeprowadzimy analizę widmową krótkiego sygnału dźwiękowego. Jest to dźwięk E1 grany na piszczałce (rys. 6.7). Widać poszczególne harmoniczne (kolejne wielokrotności) częstotliwości podstawowej.



Rys. 6.7. Sygnał dźwiękowy i jego widmo amplitudowe

Dodatkowe ćwiczenia:

Zmienić częstotliwość próbkowania sygnału i liczbę próbek w oknie czasowym. Zaobserwować wpływ tych parametrów na rozdzielczość analizy widmowej. Podać zależności określające zakres i rozdzielczość analizy widmowej.

Zastosować różne okna czasowe i zaobserwować ich wpływ na kształt widma. Podać jakie okna i w jakich przypadkach dają najlepsze efekty analizy.

Skrypt Matlaba realizujący analizę widmową sygnału dźwiękowego jest przedstawiony na Listingu 6.2.

```
clc
clear
close all

% Wczytanie pliku dźwiękowego
plik='.\sounds\E1.wav';
[wav, fp] = audioread(plik);

lprob=length(wav);
tk=lprob/fp;
t=linspace(0,tk,lprob);

figure(1)
subplot(211)
plot(t,wav);
grid on

% Obliczenie widma sygnału dźwiękowego
yw=abs(fft(wav));
yw(1)=yw(1)/lprob;
yw(2:end)=yw(2:end)./(lprob/2);
f=linspace(0,fp,lprob);

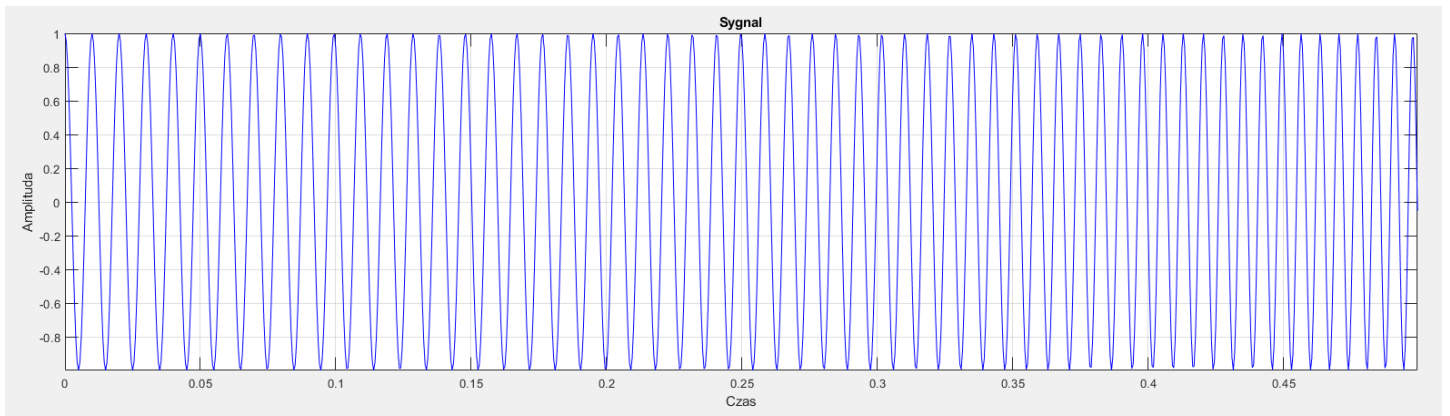
subplot(212)
semilogy(f(1:ceil(lprob/2.5)),yw(1:ceil(lprob/2.5)));
grid on
axis tight
```

Listing 6.2. Skrypt liczący widmo sygnału dźwiękowego

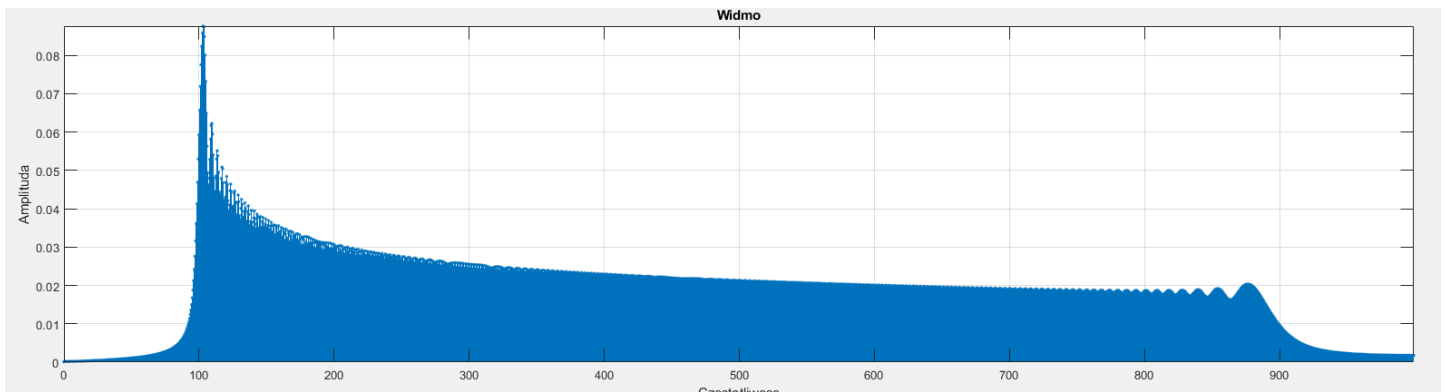
6.3. Analiza czasowo-częstotliwościowa – krótkoczasowa transformata Fouriera

Licząc widmo z wykorzystaniem transformaty Fouriera nie mamy żadnej informacji o zmianach częstotliwości sygnału badanego w czasie. W pewnych przypadkach taka informacja jest bardzo pomocna. Można ją uzyskać stosując krótkoczasową transformatę Fouriera.

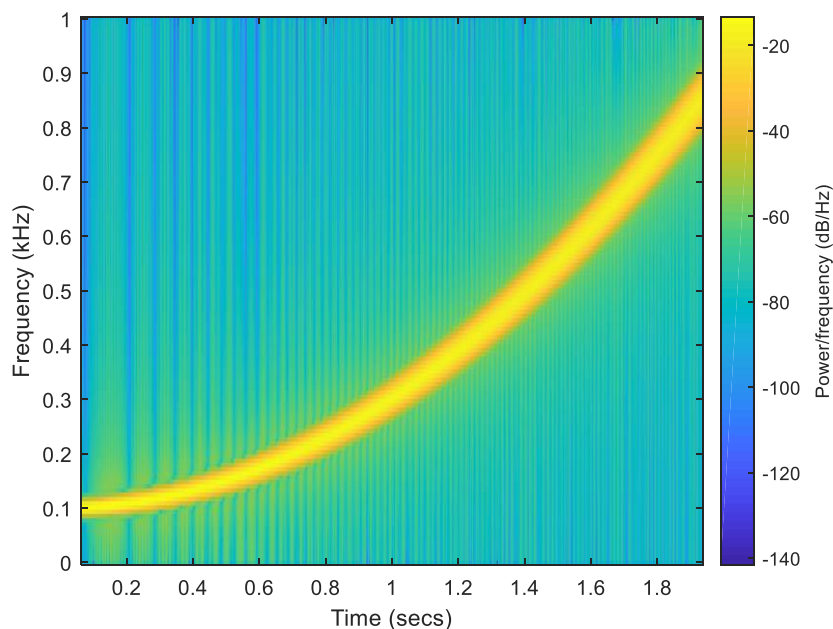
Założmy, że mamy sygnał y , którego częstotliwość zmienia się w funkcji kwadratowej w czasie od 100 Hz do 880 Hz. Fragment tego sygnału przedstawiony jest na rys. 6.8. Jeśli policzymy widmo amplitudowe tego sygnału, to otrzymamy wynik zaprezentowany na rys. 6.9.



Rys. 6.8. Sygnał y o zmiennej częstotliwości



Rys. 6.9. Widmo amplitudowe sygnału y o zmiennej częstotliwości



Rys. 6.10. Spektrogram sygnału o zmiennej częstotliwości

W widmie można znaleźć prążki reprezentujące wszystkie częstotliwości z przedziału 100-880 Hz, ale nie wiemy, w której chwili czasowej one występowały. Dużo więcej informacji można uzyskać stosując krótkoczasową transformatę Fouriera. W tym przypadku wynikiem analizy jest spektrogram zaprezentowany na rys. 6.10. Dominujące częstotliwości w sygnale (kolor żółty na rysunku) zaprezentowane są na płaszczyźnie czas-

częstotliwość. Łatwo można odczytać, że na początku sygnału (dla $t=0$) częstotliwość wynosiła 100Hz, dla $t=1$ sek. częstotliwość jest równa 300 Hz itd. Widać też, że częstotliwość zmienia się w funkcji kwadratowej z czasem.

Skrypt Matlab'a liczący widmo i spektrogram sygnału o zmiennej częstotliwości przedstawiony jest na Listingu 6.3.

```
clear
close all

% Częstotliwość próbkowania
Fs=2000;

tp=1/Fs;

% Czas analizy sygnału
t=0:tp:2; % częstotliwość próbkowania Fs=2000Hz; czas końcowy tk=2s

% Analizowany Sygnał
% t0 odpowiada f=100Hz; t=1s odpowiada f=300Hz
y=chirp(t,100,1,300,'quadratic');

M=length(y);

% Obliczenie widma
wid=abs(fft(y));
N=length(wid);
wid(1)=wid(1)/N;
wid(2:end)=wid(2:end)./(N/2);

df=Fs/N;
fprintf('Rozdzielczość analizy = %1.2f Hz\n',df);
f=0:df:(Fs-df);

figure(1)

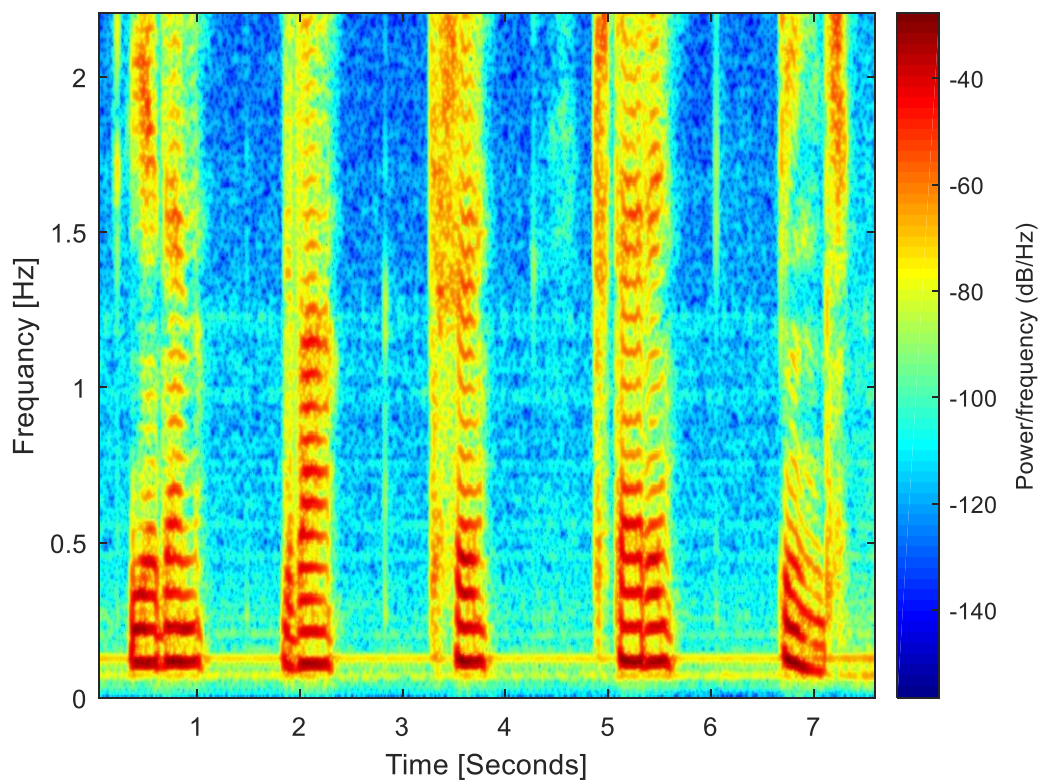
subplot(211)
%plot(t,y,'b');
plot(t(1:1000),y(1:1000),'b');
axis tight;
grid on
xlabel('Czas')
ylabel('Amplituda')
title('Sygnał')

subplot(212)
stem(f(1:fix(N/2)),wid(1:fix(N/2)),'.');
axis tight;
grid on
xlabel('Czestotliwosc')
ylabel('Amplituda')
title('Widmo')

figure(2)
% Obliczenie spektrogramu
spectrogram(y,256,250,256,Fs,'yaxis');
```

Listing 6.3. Skrypt Matlab'a liczący widmo i spektrogram sygnału o zmiennej częstotliwości

Na rys. 6.11 przedstawiony jest wynik zastosowania krótkoczasowej transformacji Fouriera do sygnału mowy (kolejno wypowiedzane słowa „jeden, dwa, trzy, cztery, pięć”). Widać złożoność takiego sygnału. Można zauważyć podstawową częstotliwość wypowiedzanych słów oraz poszczególne harmoniczne.



Rys. 6.11. Spektrogram głosu

Skrypt Matlab'a liczący spektrogram sygnału mowy przedstawiony jest na Listingu 6.4.

```
clear
close all

[y2,Fs]=audioread('..\sounds\123.wav');

figure(1)
spectrogram(y2,kaiser(2048,8),2000,4096,Fs,'yaxis');
colormap jet
xlabel('Time [Seconds]'); ylabel('Frequency [Hz]');

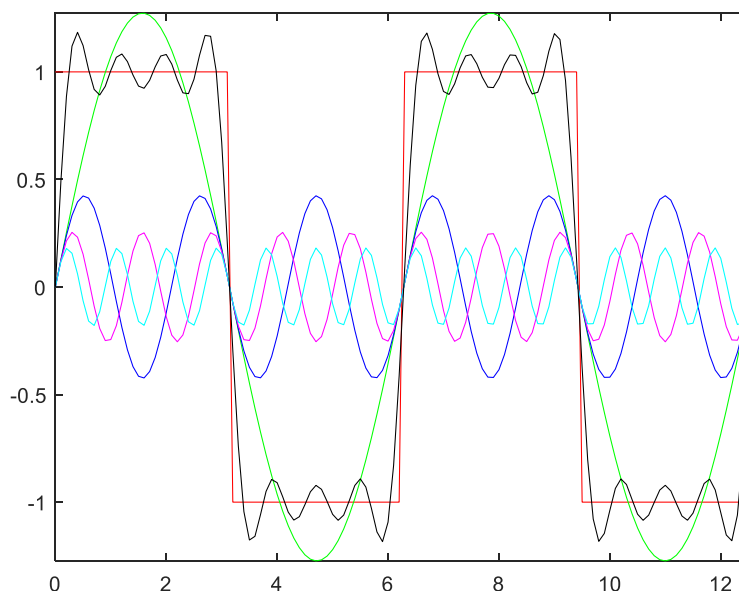
pause (1)
ylim([0 Fs/(5*2*1000)])
```

Listing 6.4. Skrypt Matlab'a liczący spektrogram sygnału mowy

Korzystając z załączonych skryptów (Listing 5.3 i 5.4) proszę policzyć spektrogramy innych sygnałów.

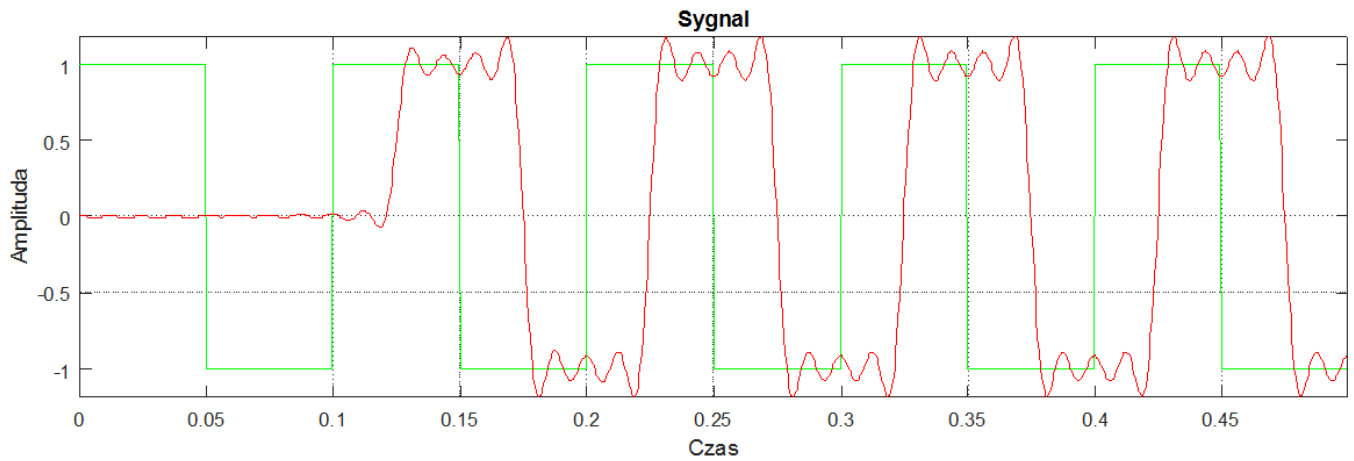
6.4. Badanie filtrów cyfrowych o skończonej i nieskończonej odpowiedzi impulsowej

Sygnal prostokątny może być rozłożony w szereg Fouriera. Zawiera on sinusoidę o podstawowej częstotliwości (takiej samej jak częstotliwość sygnału prostokątnego) oraz sinusoidy o nieparzystej wielokrotności tej częstotliwości (harmoniczne) o odpowiednich amplitudach. Przybliżenie sygnału prostokątnego zrekonstruowanego z czterech sinusoid jest przedstawione na rys. 6.12 (kolor czarny).

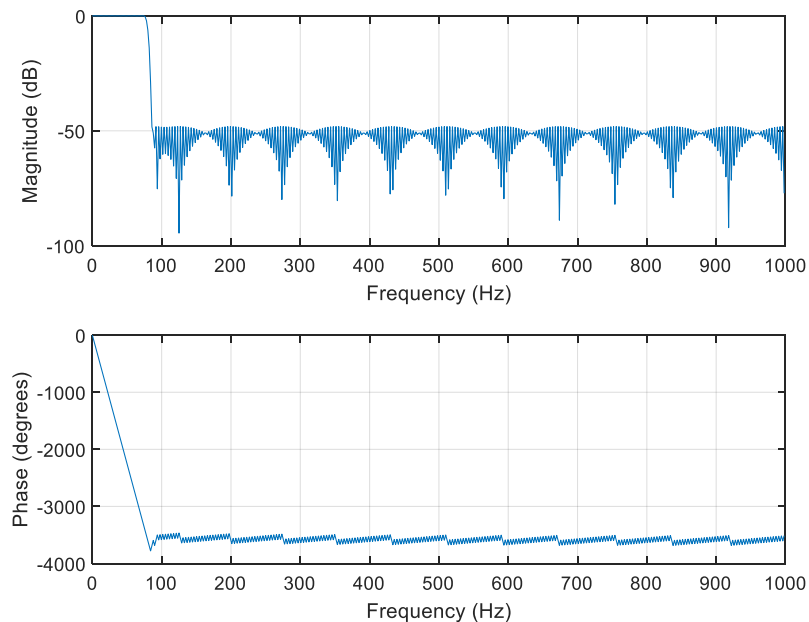


Rys. 6.12. Sygnal prostokątny zrekonstruowany z czterech sygnałów sinusoidalnych

W pierwszym ćwiczeniu spróbujemy przefiltrować sygnał prostokątny w taki sposób, aby pozostawić tylko cztery pierwsze harmoniczne. W prezentowanym przykładzie sygnał prostokątny ma częstotliwość 10 Hz. Aby pozostawić cztery pierwsze (nieparzyste) harmoniczne należy skonstruować filtr dolnoprzepustowy o częstotliwości odcięcia 80 Hz (czyli filtr przepuści sinusoidy o częstotliwości 10 Hz, 30 Hz, 50 Hz, 70 Hz, ale zablokuje 90 Hz i wyższe harmoniczne). Na początku zaprojektujemy filtr o skończonej odpowiedzi impulsowej (funkcja *firpm()* w Matlabie). Charakterystyka amplitudowa i fazowa tego filtru jest przedstawiona na rys. 6.14. Należy zwrócić uwagę, że charakterystyka fazowa jest liniowa w paśmie przepustowym. Filtr nie zniekształca więc sygnału. Sygnal oryginalny oraz przefiltrowany są zaprezentowane na rys. 6.13. Sygnal przefiltrowany jest przesunięty w stosunku do oryginalnego, ale nie zniekształcony. Filtry o skończonej odpowiedzi impulsowej mają duży rząd (w tym przypadku 500). Powoduje to, że aż 500 próbek sygnału musi przejść przez filtr, żeby on w pełni zadziałał. Stąd czas odpowiedzi takich filtrów jest stosunkowo długi.



Rys. 6.13. Sygnał oryginalny (zielony) oraz przefiltrowany (czerwony) z użyciem filtru o skończonej odpowiedzi impulsowej



Rys. 6.14. Charakterystyka amplitudowa i fazowa filtru o skończonej odpowiedzi impulsowej

Skrypt Matlab'a realizujący filtr SOI jest przedstawiony na Listingu 6.5.

```
clear;
clc;
close all;

% Częstotliwość próbkowania
Fs=2000;

%parametry sygnału
fsyg=10;
ok=1/fsyg;
A=1;
faza=0;
```

```

sks=0;

T=1/Fs;
% Czas analizy
t=0:T:(5*ok)-T);

y=A*square(2*pi*fsyg*t+faza)+sks;
%y=A*sawtooth(2*pi*fsyg*t+faza, 0)+sks;

% Projekt filtru SOI - funkcja firpm()
a=1;
N=500;
Ho=[1 1 0 0];
fo=[0 75 85 (Fs/2)]/(Fs/2);
b=firpm(N,fo,Ho);

figure(1)
freqz(b,a,512,Fs);

yf = filter(b,a,y);

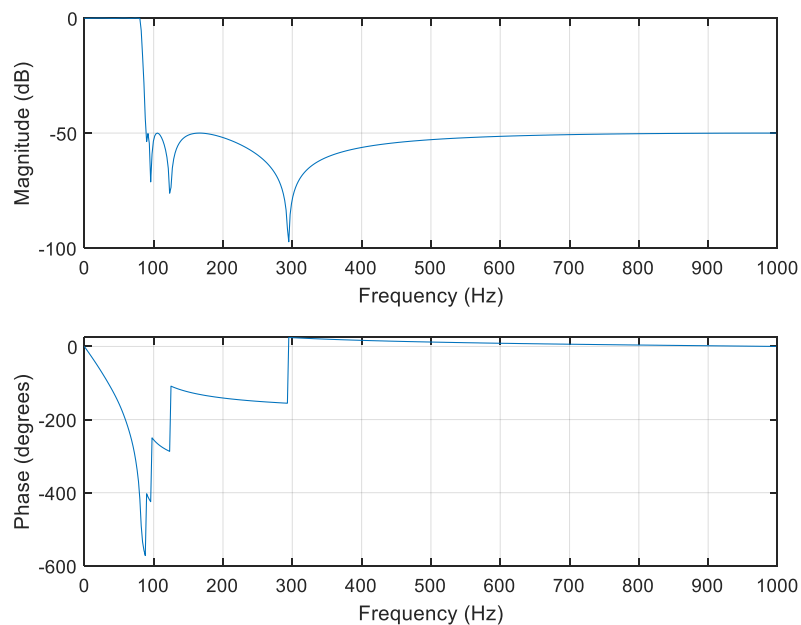
figure(2)
subplot(211)
plot(t,y,'-g');
hold on
plot(t,yf,'-r');

axis tight
grid on;
xlabel('Czas')
ylabel('Amplituda')
title('Sygnal')

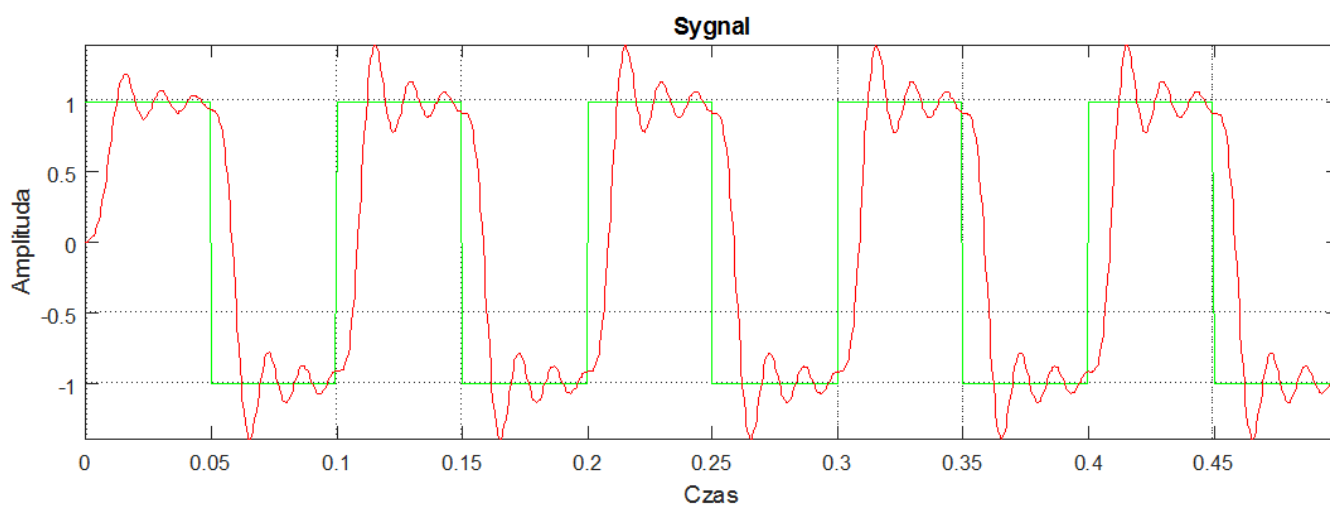
```

Listing 6.5. Skrypt Matlaba realizujący filtr SOI

W drugim ćwiczeniu zrobimy to samo, ale z użyciem filtru o nieskończonej odpowiedzi impulsowej. Do zaprojektowania filtru użyjemy funkcji *ellip()*. Charakterystyka amplitudowa i fazowa tego filtru jest przedstawiona na rys. 6.16. Jeśli chodzi o charakterystykę amplitudową, to jest ona bardzo podobna do charakterystyki filtru o skończonej odpowiedzi impulsowej (ćwiczenie 1). Jednak charakterystyka fazowa jest nieliniowa w paśmie przepustowym. Chociaż na rysunku ta nieliniowość może wydawać się niewielka, ma ona znaczny wpływ na sygnał filtrowany. Sygnał oryginalny oraz przefiltrowany są zaprezentowane na rys. 6.16. Widać zniekształcenie sygnału przefiltrowanego. Jest ono spowodowane nieliniowym przesunięciem poszczególnych harmonicznym sygnału, które są później dodawane do siebie tworząc sygnał przefiltrowany. Filtry o nieskończonej odpowiedzi impulsowej mają znacznie mniejsze rzędy niż filtry o skończonej odpowiedzi impulsowej (w tym przypadku 8). Stąd też czas odpowiedzi takiego filtru jest bardzo krótki.



Rys. 6.15. Charakterystyka amplitudowa i fazowa filtra o skończonej odpowiedzi impulsowej



Rys. 6.16. Sygnał oryginalny (zielony) oraz przefiltrowany (czerwony) z użyciem filtra o nieskończonej odpowiedzi impulsowej

Skrypt Matlab'a realizujący filtr NOI jest przedstawiony na Listingu 6.6.

```
clear;
clc;
close all;

% Częstotliwość próbkowania
Fs=2000;

%parametry sygnału
fsyg=10;
ok=1/fsyg;
A=1;
faza=0;
```

```

sks=0;

T=1/Fs;
% Czas analizy
t=0:T:(5*ok)-T);

y=A*square(2*pi*fsyg*t+faza)+sks;
%y=A*sawtooth(2*pi*fsyg*t+faza, 0)+sks;

% Projekt filtru NOI - funkcja ellip()
N=8;
Rp=0.1;
Rs=50;
wn=[80]/(Fs/2);
[b,a]=ellip(N,Rp,Rs,wn);

figure(1)
freqz(b,a,512,Fs);

yf = filter(b,a,y);

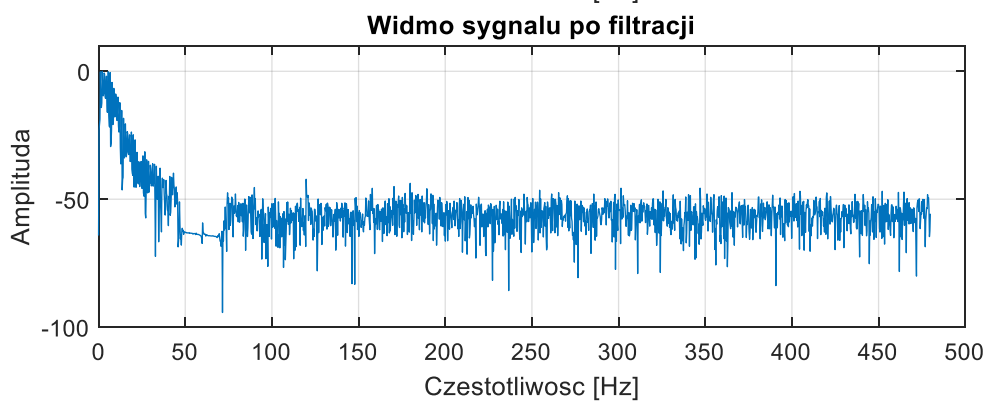
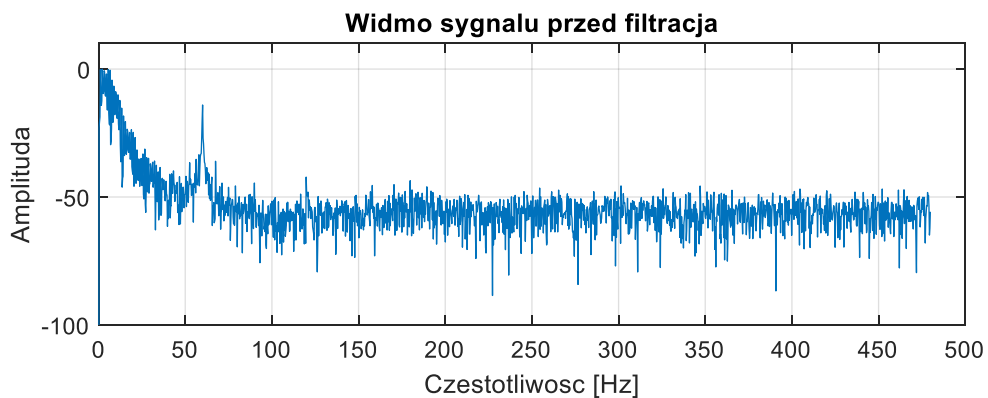
figure(2)
subplot(211)
plot(t,y,'-g');
hold on
plot(t,yf,'-r');

axis tight
grid on;
xlabel('Czas')
ylabel('Amplituda')
title('Sygnal')

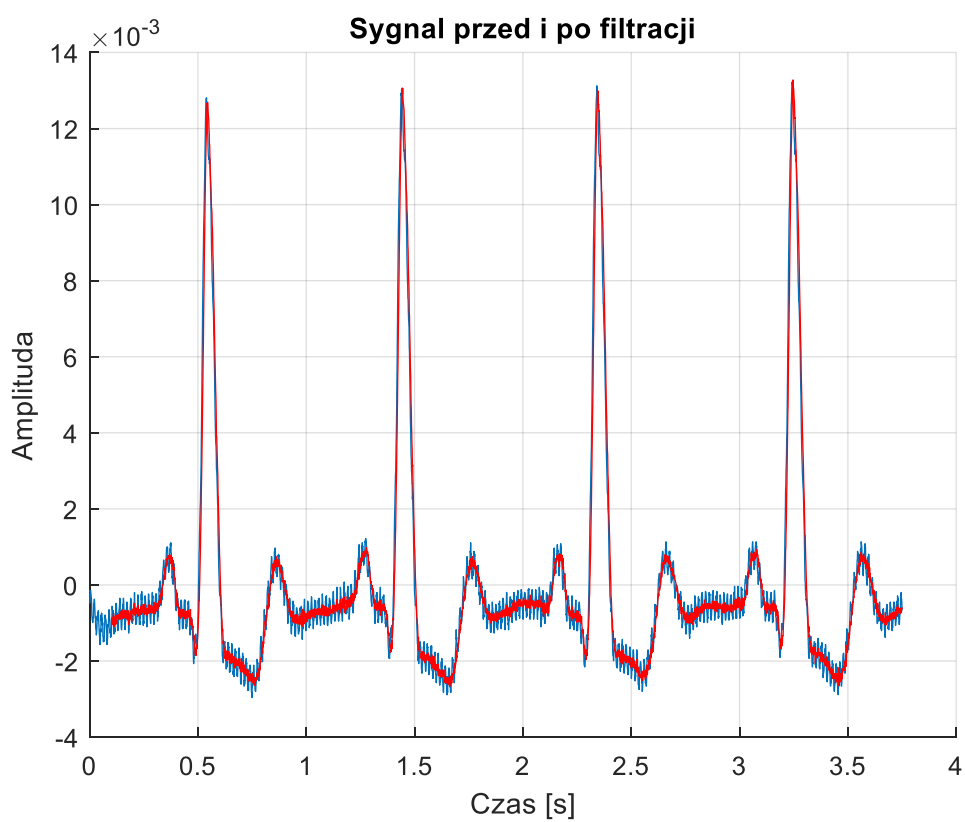
```

Listing 6.6. Skrypt Matlab'a realizujący filtr NOI

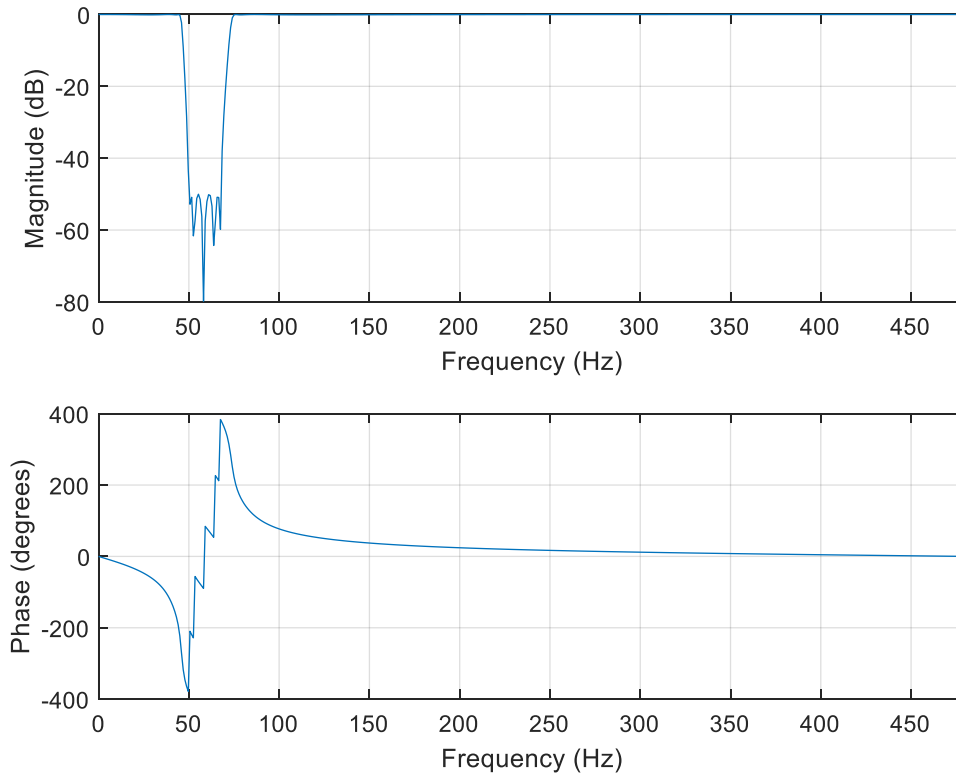
W ćwiczeniu 3 zajmiemy się bardziej praktycznym przypadkiem. Mamy sygnał EKG, na który nakładają się zakłócenia od sieci (60 Hz) (rys. 6.18 – kolor niebieski). Zaprojektujemy filtr cyfrowy, który usunąłby te zakłócenia. Ponieważ zależy nam na szybkim zadziałaniu filtru, zastosujemy filtr eliptyczny (NOI). Wprowadzone przez ten filtr zniekształcenia fazowe w tym przypadku nie powinny mieć większego znaczenia. Najpierw wyznaczmy widmo sygnału EKG (rys. 6.17 – góra). Widać zakłócenia od sieci energetycznej – prążek 60 Hz. Zaprojektujemy filtr pasmowo-przepustowy tak by zablokować częstotliwości wokół 60 Hz (pasma przepustowe 0-55 Hz, 65-fs/2 Hz). Charakterystyki amplitudowa i fazowa tego filtru są przedstawione na rys. 6.19. Następnie przefiltrujemy sygnał. Rezultat działania takiego filtru jest przedstawiony na rys. 6.18 – kolor czerwony. Widać wyraźnie, że zakłócenia od sieci zostały odfiltrowane. Widmo sygnału po filtracji jest przedstawione na rys. 6.17 – dół. Tu też widać, że częstotliwości wokół 60 Hz zostały usunięte z sygnału.



Rys. 6.17. Widmo sygnału EKG przed i po filtracji



Rys. 6.18. Sygnal EKG przed i po filtracji



Rys. 6.19. Charakterystyka amplitudowa i fazowa zaprojektowanego filtra eliptycznego

Skrypt Matlab'a użyty do filtracji sygnału EKG jest przedstawiony na Listingu 6.7.

```
clear;
close all;

load ekg.mat
ekg=ekg-mean(ekg);

fp=960;
f=linspace(0,fp,3600);
t=linspace(0,(3600/fp),3600);

% Projekt filtru ellip
Rp=0.1;
Rs=50;
wp=[45 75]/(fp/2);
ws=[50 65]/(fp/2);
[N,wn]=ellipord(wp,ws,Rp,Rs);
[b,a]=ellip(N,Rp,Rs,wn,'stop');

figure(1);
freqz(b,a,512,fp);

% Filtracja sygnału
sf=filter(b,a,ekg);

figure(2);
hold on
plot(t,ekg);
plot(t(100:3600),sf(100:3600),'r');
grid;
hold off
```

```

title('Sygnał przed i po filtracji');
xlabel('Czas [s]');ylabel('Amplituda');

w=fft(ekg);
wlin=abs(w);
wlog=20*log10(wlin/max(wlin));
wf=fft(sf);
wflin=abs(wf);
wflog=20*log10(wflin/max(wflin));

figure(3);
subplot(211);
plot(f(1:1800),wlog(1:1800));
ylim([-100 10]);
grid;
title('Widmo sygnału przed filtracją');
xlabel('Częstotliwość [Hz]');ylabel('Amplituda');
subplot(212);
plot(f(1:1800),wflog(1:1800));
ylim([-100 10]);
grid;
title('Widmo sygnału po filtracji');
xlabel('Częstotliwość [Hz]');ylabel('Amplituda');

```

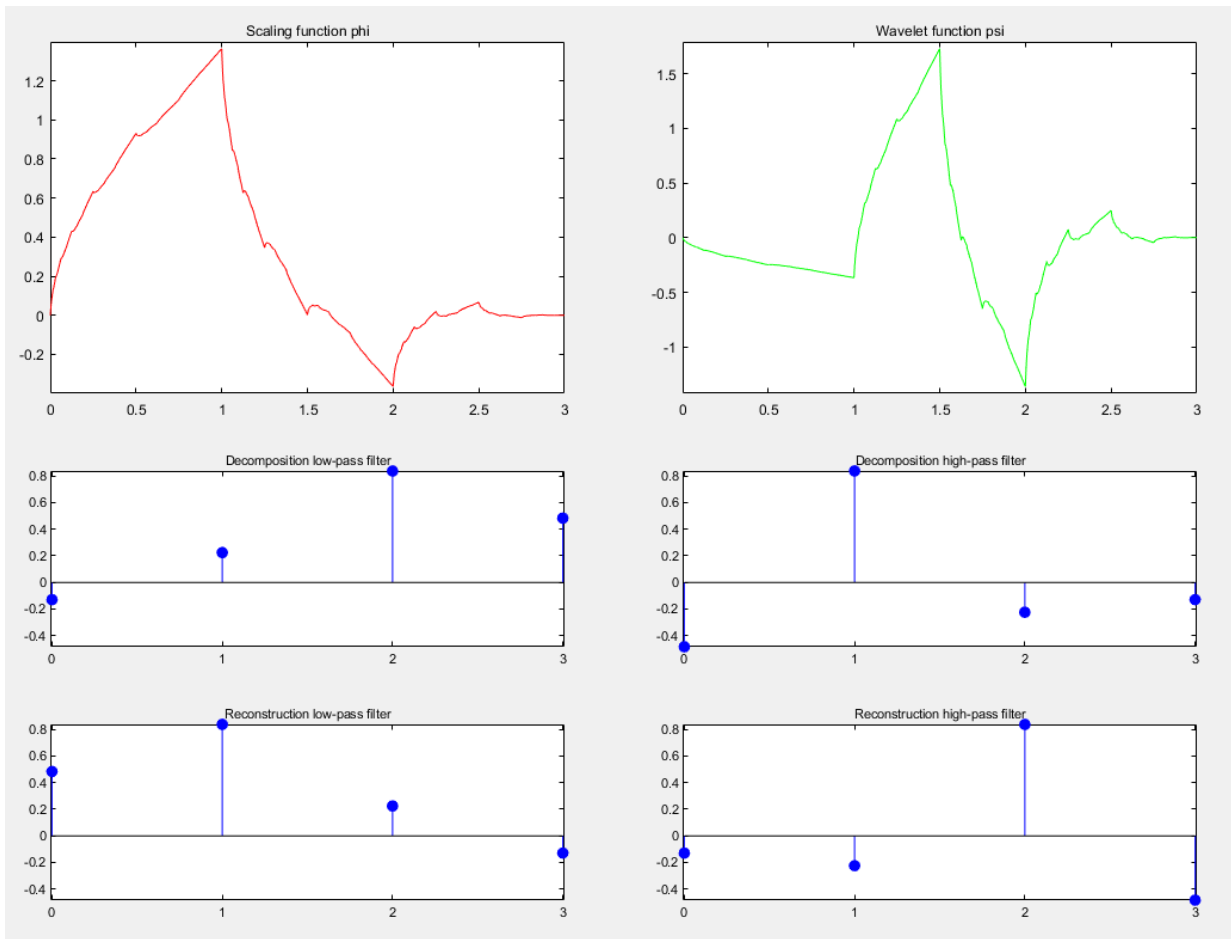
Listing 6.7. Skrypt Matlaba użyty do filtracji sygnału EKG

Jako dodatkowe ćwiczenie proszę zaprojektować analogiczny filtr, ale o skończonej odpowiedzi impulsowej i porównać otrzymane rezultaty.

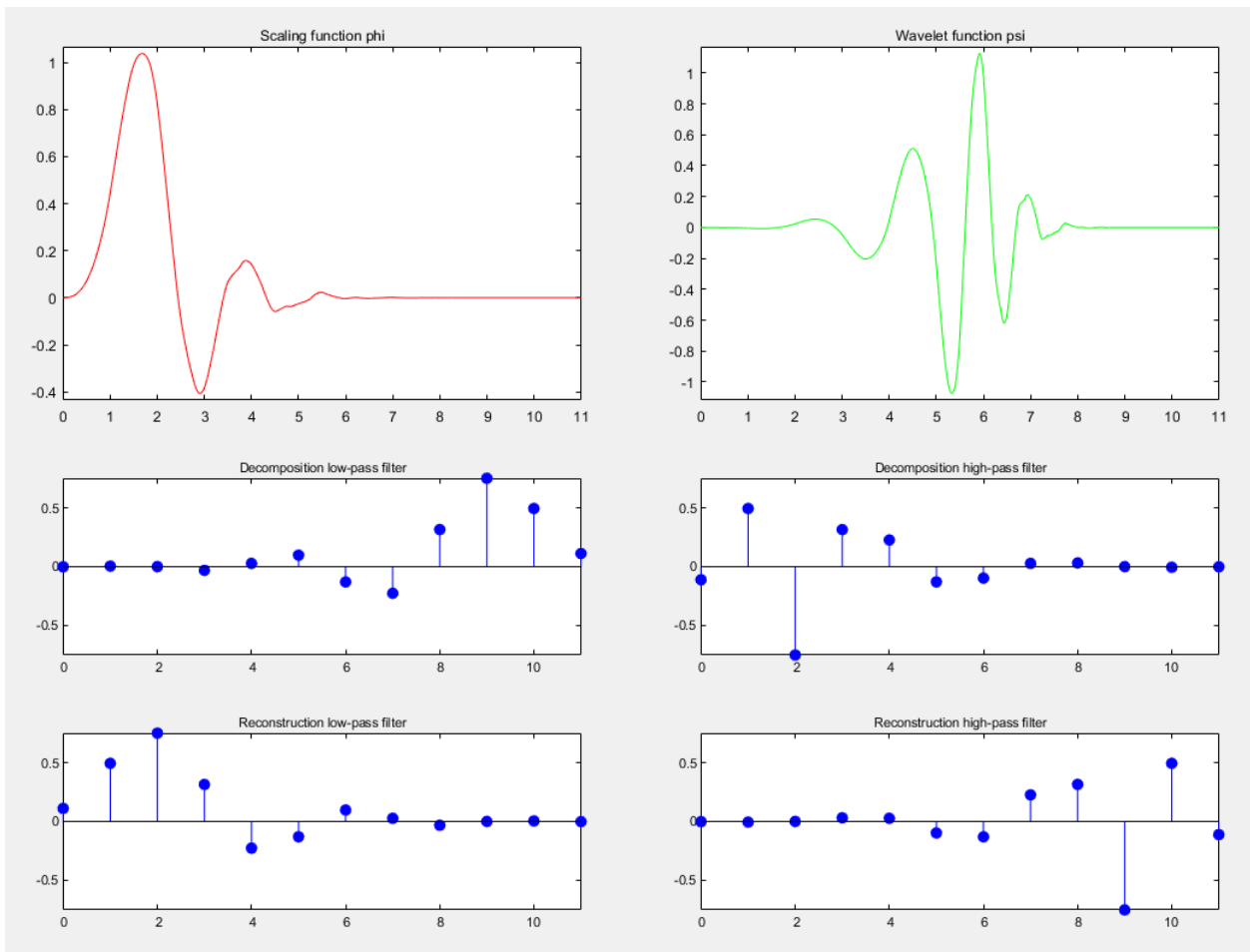
6.5. Transformata falkowa

W tym ćwiczeniu skorzystamy z narzędzia Matlaba o nazwie „waveletAnalyzer”. Umożliwia ono przeprowadzenie bardzo wielu operacji z użyciem falek. My skoncentrujemy się na obejrzeniu niektórych funkcji falkowych oraz przeprowadzeniu dekompozycji falkowej sygnałów jednowymiarowych.

Po wybraniu opcji „WaweletDisplay” mamy możliwość obejrzenia i porównania wielu funkcji falkowych. Na rys. 6.20. i 6.21. zostały zobrazowane funkcje falkowe Daubechies db2 i db6. Można zaobserwować kształt funkcji macierzystej i falkowej. Im rząd jest wyższy tym funkcja jest bardziej gładka. Przy dekompozycji falkowej należy dobrać kształt funkcji falkowej po kontem swoich potrzeb, albo do konkretnego sygnału. W dole rysunków przedstawione są współczynniki filtrów używanych do dekompozycji i rekonstrukcji falkowej. Widać, że liczba współczynników filtrów zależy od rzędu falki.



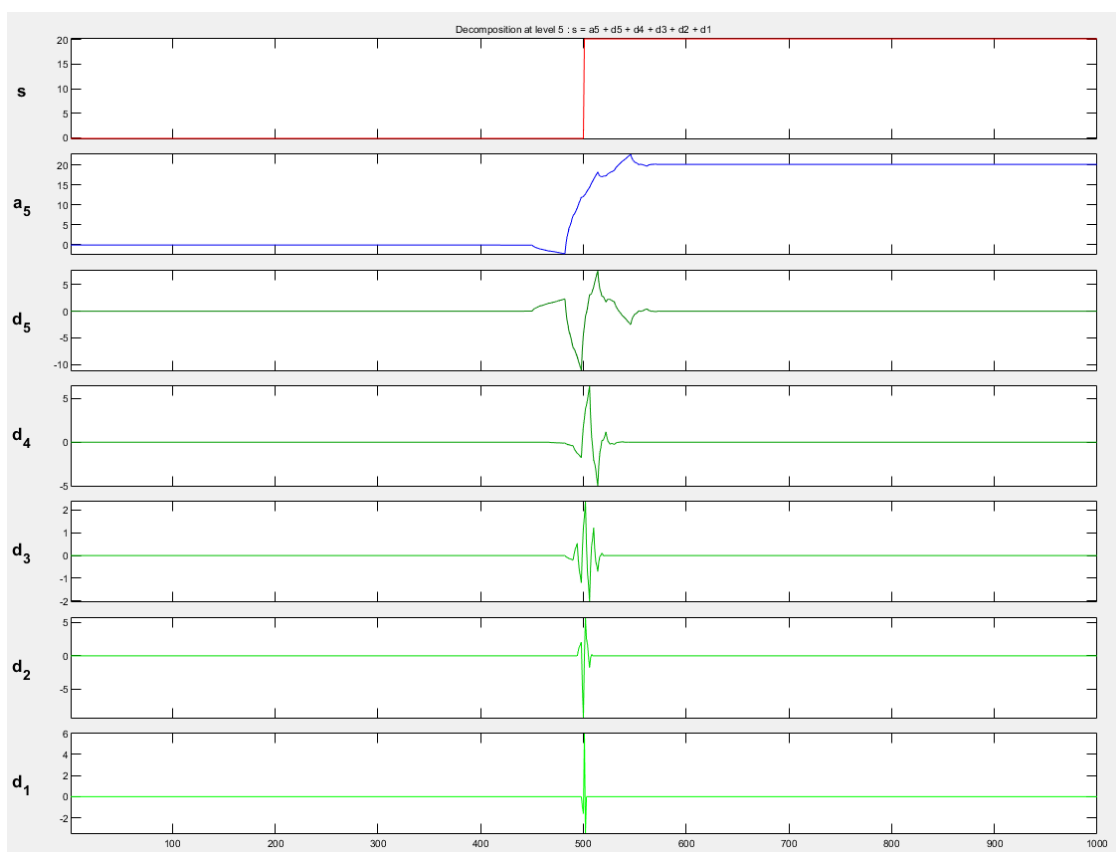
Rys. 6.20. Funkcje falkowe db2 oraz odpowiadające im filtry używane w procesie dekompozycji i rekonstrukcji



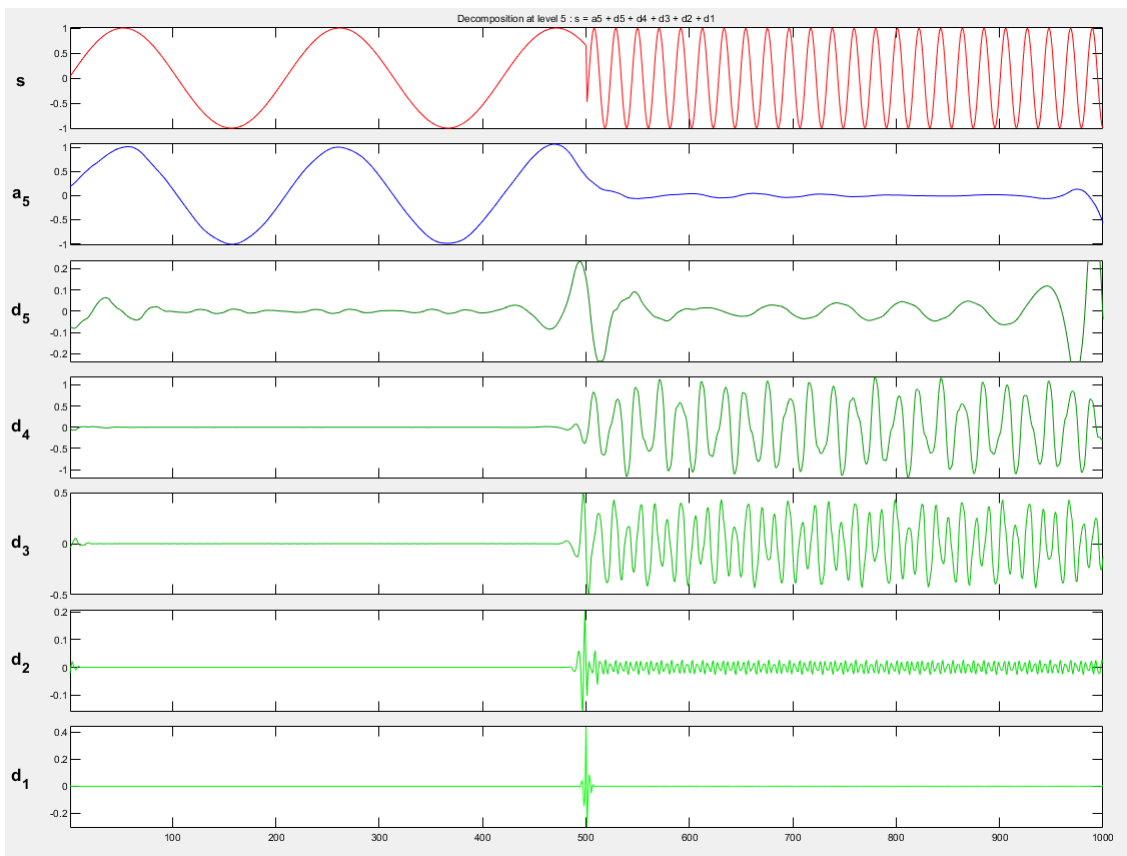
Rys. 6.21. Funkcje falkowe db6 oraz odpowiadające im filtry używane w procesie dekompozycji i rekonstrukcji

Kolejne przykłady przedstawiają dekompozycję falkową sygnałów jednowymiarowych. W pierwszym przykładzie analizowany jest sygnał skoku jednostkowego (rys. 6.21). Dekompozycja została wykonana za użyciem narzędzia „Wawelet 1-D”. Zastosowaną falką była db2, rozkład przeprowadzono na pięciu poziomach. Mamy więc sygnał aproksymacji a_5 (najniższa częstotliwość) oraz pięć sygnałów detali d_5 do d_1 . Detal d_1 reprezentuje składową o najwyższej częstotliwości, a d_5 o najniższej. W tym przypadku łatwo zaobserwować, że składowe dekompozycji przypominają kształtem zastosowaną falkę.

W drugim przykładzie sygnałem, który został poddany dekompozycji falkowej, jest sygnał zaprezentowany na rys. 6.23. Sygnał ten składa się z połączonych dwóch sygnałów sinusoidalnych. W miejscu połączenia występuje nieciągłość. Drugi sygnał sinusoidalny ma częstotliwość większą niż pierwszy. Zastosowano do dekompozycji falkę db2, a rozkład przeprowadzono na pięciu poziomach. W tym przypadku aproksymacja a_5 reprezentuje w zasadzie pierwszy sygnał sinusoidalny o niższej częstotliwości. Natomiast drugi sygnał sinusoidalny został głównie zdekomponowany na detale d_4 i d_3 . W miejscu nieciągłości występują wszystkie częstotliwości.



Rys. 6.22. Dekompozycja falkowa sygnału skoku jednostkowego



Rys. 6.23. Dekompozycja falkowa dwóch połączonych sinusoid z nieciągłością

7. Pytania kontrolne

Proszę zaznaczyć poprawne odpowiedzi. Może być jedna bądź dwie poprawne odpowiedzi w każdym pytaniu.

1) Filtr cyfrowy:

- a) Jest układem odpowiednio połączonych elementów RLC
- b) *Może być programem komputerowym, programowalnym procesorem lub dedykowanym układem scalonym
- c) Ma zawsze wbudowany przetwornik a/c
- d) *Do poprawnego działania wymaga jedynie podstawowych operacji arytmetycznych (+, -, *, /)

2) Filtracja sygnału to:

- a) Policzenie transformacji Fouriera sygnału
- b) *Proces przetwarzania dokonany na sygnale w dziedzinie czasu, powodujący zmiany w widmie sygnału oryginalnego
- c) Usunięcie średniej z sygnału
- d) *Usunięcie, redukcja pewnych niepożądanych składowych sygnału filtrowanego

3) Filtr o nieskończonej odpowiedzi impulsowej (NOI):

- a) *Ma nieliniową charakterystykę fazową
- b) Zazwyczaj wymaga dużej liczby współczynników do poprawnego działania
- c) *Działa w oparciu o bieżącą i poprzednie wartości wejściowe oraz poprzednie wartości wyjściowe
- d) Nie powoduje zniekształceń sygnału

4) Filtr o skończonej odpowiedzi impulsowej (SOI):

- a) *Ma odpowiedź impulsową, która zbiega do zera w skończonym czasie
- b) Działa w oparciu o bieżącą i poprzednie wartości sygnału brane z wyjścia filtru
- c) Przestaje działać po określonym czasie
- d) Korzysta tylko z operacji dodawania do obliczenia wartości próbek wyjściowych

5) Zaletą filtrów NOI w porównaniu z filtrami SOI jest to, że:

- a) Nie zniekształcają sygnału
- b) *Szybciej zaczynają działać
- c) *Można uzyskać bardziej strome pasma przejściowe
- d) Są łatwiejsze do projektowania i zastosowania

6) Odszumianie sygnałów można zrealizować poprzez:

- a) Usunięcie trendu
- b) *Filtrację cyfrową
- c) Usunięcie dryftu
- d) Zastosowanie transformacji Gabora

7) Transformata Fouriera umożliwia:

- a) Rozkład sygnału w szereg Taylora
- b) *Policzenie widma sygnału
- c) Wyznaczenie spektrogramu sygnału
- d) Jest podstawową operacją wykonywaną przez filtry cyfrowe

8) Jeżeli rozłożymy sygnał prostokątny na harmoniczne to:

- a) Otrzymamy tylko parzyste harmoniczne
- b) *Otrzymamy tylko nieparzyste harmoniczne
- c) Nie da się rozłożyć sygnału prostokątnego na harmoniczne
- d) Otrzymamy parzyste i nieparzyste harmoniczne

9) Okno czasowe w analizie Fouriera:

- a) Powoduje, że analiza jest zawsze synchroniczna
- b) *Może być stosowane w celu zmniejszenia skutków analizy asynchronicznej
- c) Wyznacza zakres analizowanych częstotliwości
- d) *Wyznacza początek i koniec analizowanego sygnału

10) Częstotliwość Nyquista to:

- a) Najwyższa częstotliwość w badanym sygnale
- b) Częstotliwość próbkowania pomnożona przez dwa
- c) *Określa zakres analizy widmowej
- d) *Połowa częstotliwości próbkowania

11) Analiza synchroniczna

- a) *Ułatwia interpretacją widma
- b) Zachodzi najczęściej podczas rzeczywistych pomiarów
- c) Zachodzi, kiedy częstotliwość próbkowania jest odpowiednio duża
- d) *Zachodzi, kiedy analiza Fouriera obejmuje całkowitą liczbę okresów sygnału badanego

12) Okres próbkowania to:

- a) Czas rejestracji sygnału
- b) *Różnica czasowa pomiędzy zarejestrowaniem dwóch kolejnych próbek sygnału
- c) Wielokrotność okresu sygnału badanego
- d) Czas, po którym sygnał badany zmienia znak (z dodatniego na ujemny bądź odwrotnie)

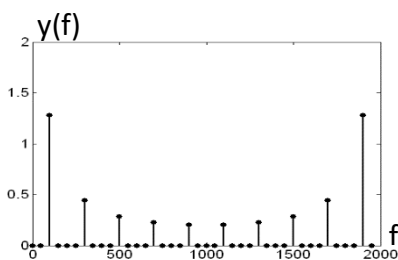
13) Częstotliwość próbkowania (przy ustalonym czasie próbkowania sygnału) wpływa na:

- a) *Zakres analizy widmowej
- b) Rozdzielczość analizy widmowej
- c) *Liczbę próbek sygnału wziętą do analizy
- d) Częstotliwość badanego sygnału

14) Zjawisko 'aliasingu' powstaje gdy:

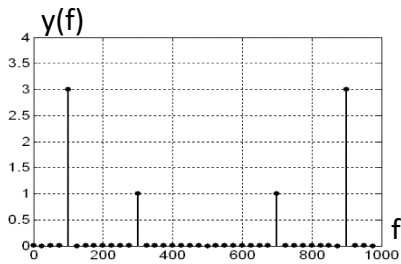
- a) Czas próbkowania sygnału jest zbyt długi,
- b) *Nie jest spełniony warunek podany w twierdzeniu o próbkowaniu (Shannona),
- c) Widma dwóch badanych sygnałów posiadają część wspólną,
- d) Częstotliwość próbkowania jest co najmniej dwukrotnie większa od częstotliwości najwyższej harmonicznej widma badanego sygnału.

15) Na poniższym rysunku przedstawiono moduł widma sygnału:



- a) Sinusoidalnego
- b) *Prostokątnego
- c) $y(t) = A_1 \sin(2\pi f_1 t) + A_2 \sin(2\pi f_2 t)$,
- d) Szumu

16) Na podstawie poniższego rysunku przedstawiającego moduł widma sygnału ($n = 40$) można powiedzieć że:



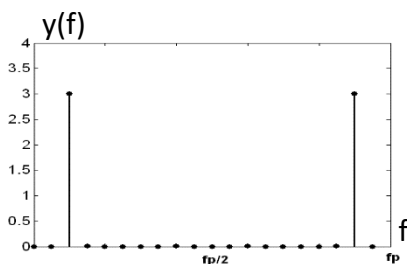
- a) *Występuje synchroniczna analiza widmowa
- b) Występuje asynchroniczna analiza widmowa
- c) *Przedstawiony sygnał można zapisać w postaci:

$$y(t) = A_1 \sin(2\pi f_1 t) + A_2 \sin(2\pi f_2 t)$$
- d) Rozdzielczość analizy widmowej $r = 0.40$

17) Zwiększanie liczby próbek n badanego sygnału przy stałej częstotliwości próbkowania powoduje:

- a) *Wydłużenie czasu analizy sygnału
- b) Poszerzenie zakresu analizy widmowej
- c) *Zwiększenie rozdzielczości r analizy widmowej
- d) Zmniejszenie rozdzielczości r analizy widmowej

18) Na poniższym rysunku przedstawiono moduł widma sygnału (spełniającego warunek podany w twierdzeniu o próbkowaniu):



- a) * $y(t) = A \sin(2\pi f t + \varphi)$
- b) Prostokątnego
- c) $y(t) = A_1 \sin(2\pi f_1 t) + A_2 \sin(2\pi f_2 t)$, gdzie $f_1 \neq f_2$
- d) Szumu

19) Krótkoczasowa transformata Fouriera:

- a) *Umożliwia wydobycie z sygnału informacji o tym, jak zmienia się jego widmo w czasie
- b) Pozwala uzyskać dokładniejsze wyniki niż zwykła transformata Fouriera
- c) Nadaje się tylko do analizy krótkich sygnałów
- d) *Podczas tej analizy sygnał zostaje podzielony na segmenty, a każdy segment podlega analizie widmowej niezależnie

20) O spektrogramie można powiedzieć że:

- a) Jest to wynik transformacji Fouriera
- b) *Argumentami w spektrogramie są częstotliwość i czas
- c) *Jest to wykres widma amplitudowego sygnału dla każdej chwili t , dla której sygnał jest określony
- d) Jest to wynik transformacji Gabora

21) Transformata Gabora:

- a) *Umożliwia osiągnięcie optymalnej rozdzielczości analizy w czasie i częstotliwości
- b) Umożliwia analizę bardzo krótkich sygnałów
- c) *Jest przypadkiem szczególnym STFT, w którym okno czasowe ma kształt gaussowski
- d) Niweluje przeciek widma

22) O rozdzielczości analizy w STFT można powiedzieć że:

- a) Największy wpływ ma na nią częstotliwość próbkowania sygnału
- b) *Rozdzielczość w czasie i częstotliwości jest ze sobą powiązana, zwiększając rozdzielczość w dziedzinie czasu jednocześnie zmniejszamy rozdzielczość w dziedzinie częstotliwości
- c) Rozdzielczość w czasie i częstotliwości jest ze sobą powiązana, zwiększając rozdzielczość w dziedzinie czasu jednocześnie zwiększamy rozdzielczość w dziedzinie częstotliwości
- d) *Jest zdeterminowana przez szerokość okna analizy

23) W przypadku przeprowadzenia analizy falkowej sygnału

- a) Uzyskujemy dokładniejsze informacje o składowych widma sygnału niż w analizie Fouriera
- b) *Do dekompozycji falkowej możemy wykorzystać wiele różnych funkcji falkowych
- c) *Uzyskujemy wynik dekompozycji przedstawiony na płaszczyźnie czas-skala
- d) Mamy bezpośrednią możliwość odczytania jakie częstotliwości występują w określonym czasie trwania sygnału

24) Sygnały detalu w analizie falkowej

- a) *Zawierają komponenty sygnału o wyższych częstotliwościach
- b) Powstają w wyniku filtracji sygnału filtrem dolnoprzepustowym
- c) *Może być wiele sygnałów detalu, ich liczba zależy od liczby poziomów dekompozycji
- d) Zawierają komponenty sygnału o niższych częstotliwościach

25) Aproksymacja w analizie falkowej

- a) Zawiera komponenty sygnału o najwyższych częstotliwościach
- b) *Zawiera komponenty sygnału o najniższych częstotliwościach
- c) Powstaje przez usunięcie średniej z bloków sygnału analizowanego
- d) *Jest tylko jeden taki komponent w dekompozycji falkowej

26) Kiedy warto stosować alokację bitów

- a) W odniesieniu do kolejnych próbek sygnału
- b) *W odniesieniu do próbek transformaty sygnału
- c) *W przypadku kwantyzacji wektorowej
- d) W przypadku kwantyzacji skalarnej próbek sygnału

27) Co oznacza termin kodowanie transformacyjne

- a) Nierównomierną kwantyzację sygnału
- b) *Kwantyzację próbek transformaty
- c) Policzenie odwrotnej transformaty Fouriera
- d) *Rodzaj kodowania blokowego

8. Literatura

- [1] Akansu A.N., Haddad R.A.: „Multiresolution Signal Decomposition Transforms, Subbands and Wavelets”, New York, Marcel Dekker, 1995.
- [2] Białasiewicz J.T.: „*Falki i aproksymacje*”, Warszawa WNT, Warszawa, 2000.
- [3] Brigham E.O.: *The Fast Fourier Transform and its Applications*, Prentice-Hall, 1988.
- [4] Burk H. B.: *The World According to Wavelets*, A.K. Peters, 1996
- [5] Chui C.K.: „*An introduction to wavelets*”, Academic Press, San Diego, California, 1992.
- [6] Cohen L., Loughlin P.: „*Recent Developments in Time-Frequency Analysis*”, Kluwer, New York, 1998.
- [7] Daubechies I: „*Ten Lectures on Wavelets*”, SIAM Press, 1992.
- [8] Drozdek A.: „*Wprowadzenie do kompresji danych*”, WNT, Warszawa, 1999.
- [9] Gersho A., Gray R.: „*Vector Quantization and Signal Compression*”, Kluwer Academic Publishers, 1992.
- [10] Goswami J.C., Chan A.K.: „*Fundamentals of Wavelets-theory algorithms and applications*”, John Wiley & Sons Inc., New York, 1999.
- [11] Projekt badawczy finansowany przez Komitet Badań Naukowych: „*Aplikacje nowoczesnych metod analizy sygnałów w wirtualnych przyrządach pomiarowych*”, Nr rej. 8T10C00921, 2001-2003, Kier.proj.: R.J. Rak.
- [12] Lynch T.J.: „*Data Compression: Techniques and Applications*”, Lifetime Learning Publications, A Division of Wadsworth Inc., 1985.
- [13] Mallat S.: „*Wavelet Tour of Signal Processing*”, Academic Press, 1998.
- [14] „*Matlab User Manual*”, Mathworks Inc., 1999.
- [15] Mertins A.: „*Signal Analysis: Wavelets, Filter Banks, Time-Frequency Transasforms and Applications*”, Wiley, Chichester, 1999.
- [16] Rak R.: „*Wirtualny przyrząd pomiarowy*” – realne narzędzie współczesnej metrologii, OWPW, Warszawa 2003.
- [17] Sayood K.: „*Introduction to Data Compression*”, Morgan Kaufman Publishers Inc, San Francisco, 1996.
- [18] Stabrowski M.M.: „*Miernictwo elektryczne – cyfrowa technika pomiarowa*”, Oficyna Wydawnicza Politechniki Warszawskiej, Warszawa 1999.
- [19] Stabrowski M.M.: „*Cyfrowe przyrządy pomiarowe*”. PWN, Warszawa 2002.
- [20] Wojtaszczyk P.: „*Teoria falek*”, PWN, Warszawa, 2000.
- [21] Zelniker G., Taylor F.J.: „*Advanced Digital Signal Processing: Theory and Applications*”, Marcel Dekker Inc., 1994
- [22] Zieliński T.: „*Od teorii do cyfrowego przetwarzania sygnałów*”, Wydział EAIIE Akademia Górniczo-Hutnicza, Kraków, 2002.