

Zastosowanie Transformaty Falkowej w nadzorowaniu

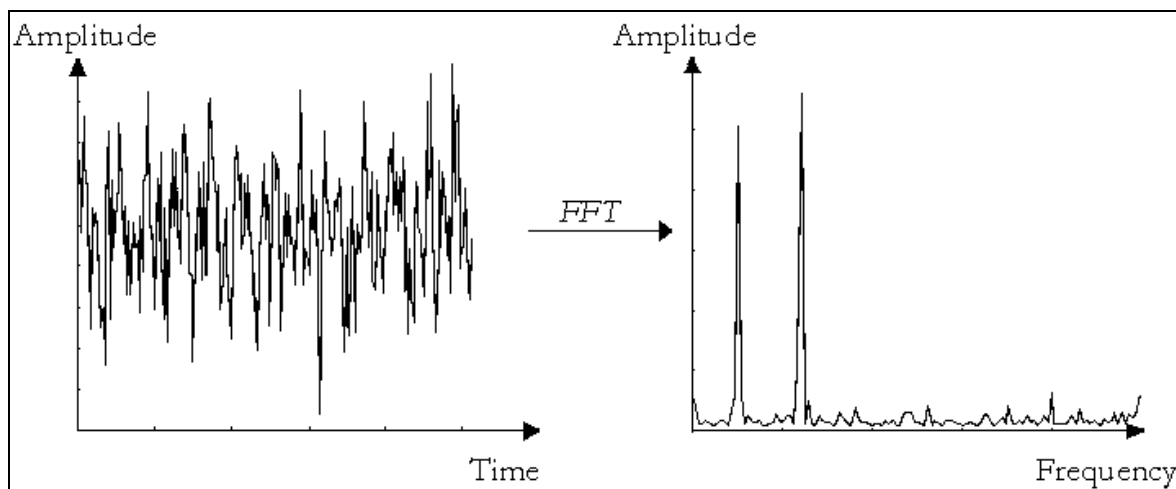
1. Wstęp

Dobrze znana i szeroko wykorzystywana w analizie sygnałów Transformata Fouriera (ang. *Fourier Transform*, FT) dokonuje dekompozycji sygnału na składowe sinusoidalne o różnych częstotliwościach:

$$F(\omega) = \int_{-\infty}^{\infty} f(t)e^{-j\omega t} dt \quad (1)$$

gdzie $F(\omega)$ jest widmem częstotliwościowym, a $f(t)$ badaną realizacją czasową.

Podczas transformacji następuje zmiana dziedziny czasu na dziedzinę częstotliwości (Rys. 1). W wyniku tej operacji informacja czasowa zostaje utracona. Jest to poważną wadą w przypadku analizy sygnałów niestacjonarnych, gdy informacja o chwili wystąpienia i czasie trwania danego zjawiska jest bardzo istotna. Z tego względu transformata Fouriera jest skuteczna jedynie dla sygnałów periodycznych, regularnych i stacjonarnych.



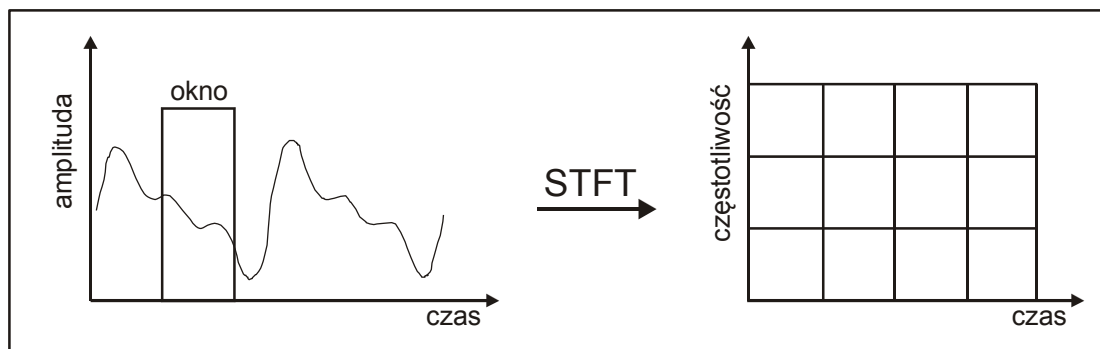
Rys. 1. Transformata Fouriera. Przejście z dziedziny czasu do dziedziny częstotliwości: sygnał w dziedzinie czasu (po lewej) i w dziedzinie częstotliwości (po prawej).

W pewnym stopniu remedium na wady transformaty Fouriera jest zaproponowana w roku 1946 przez Dennisa Gabora *krótko-okresowa transformata Fouriera* (ang. *Short-Time Fourier Transform*, STFT). Transformata ta, za pomocą przesuwanego wzdłuż sygnału okna czasowego $w(t)$, dzieli sygnał na małe kawałki, które można uznać za stacjonarne.

Następnie, kawałki te poddawane są FT (Rys. 2):

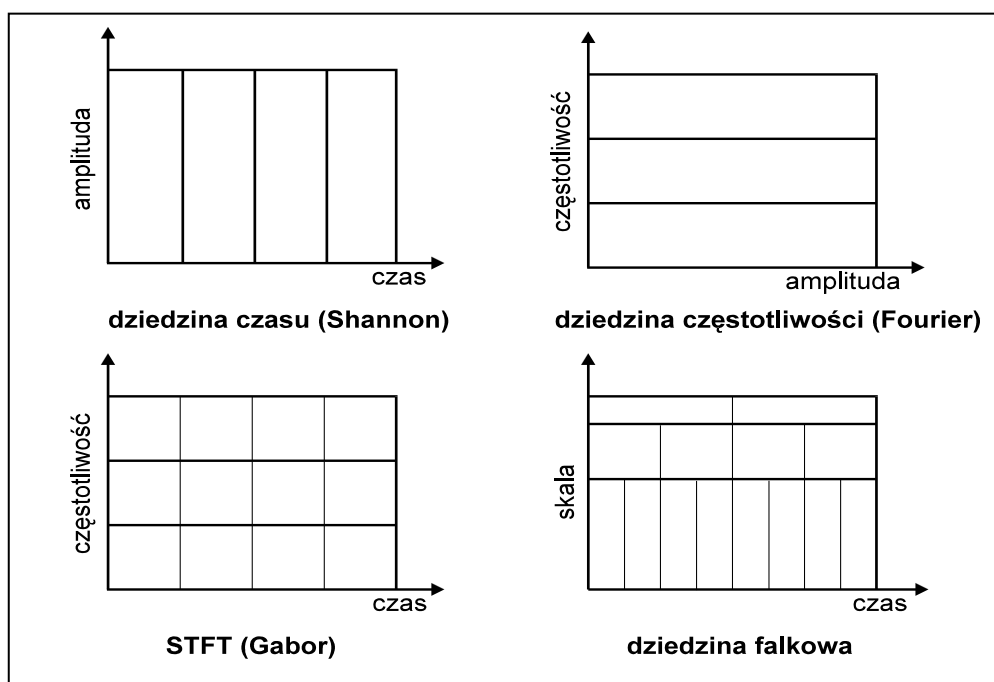
$$F(t, \omega) = \int_{-\infty}^{\infty} [f(t)w(t - \tau)]e^{-j\omega \tau} d\tau \quad (2)$$

W ten sposób odzyskuje się, w pewnym stopniu, traconą przez FT informację czasową.



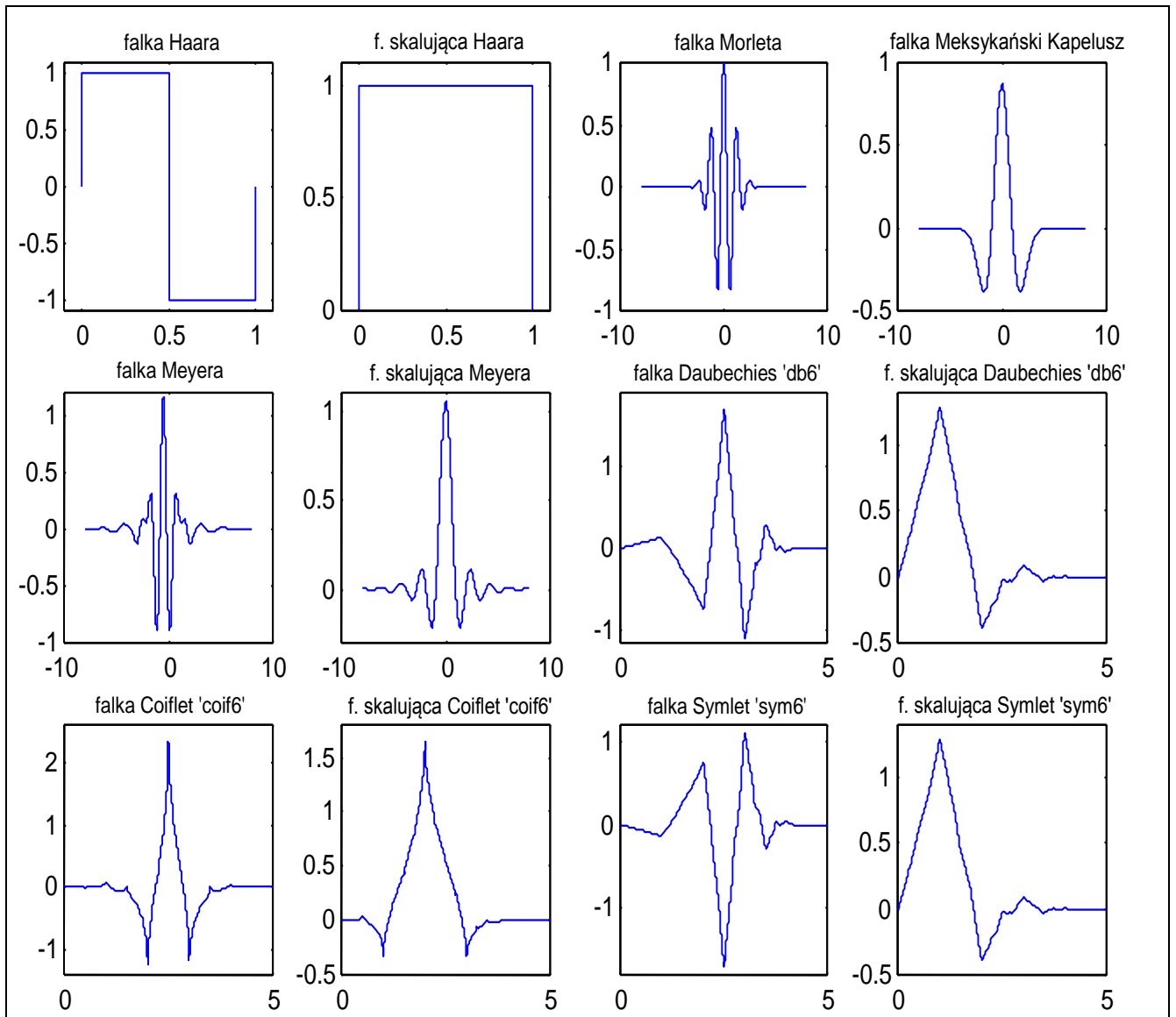
Rys. 2. Zasada działania STFT.

Główną wadą STFT jest fakt, że jakość informacji częstotliwościowej jest odwrotnie proporcjonalna do jakości informacji czasowej. Związane jest to z szerokością okna czasowego, która jest stała. Dla wąskiego okna występuje dobra rozdzielczość czasowa i słaba rozdzielczość częstotliwościowa, dla szerokiego okna sytuacja jest odwrotna. Transformata falkowa natomiast dokonuje dekompozycji sygnału na składowe falkowe o różnych częstotliwościach (skalach) i różnych pozycjach. Ponieważ falka zmienia swoją szerokość (skalę), nie występuje tu, jak w STFT, antagonizm między rozdzielczością informacji czasowej, a rozdzielczością informacji częstotliwościowej. Falkę można tu rozpatrywać w kategoriach okna o zmiennej szerokości. Dla porównania, na Rys. 3 przedstawiono rozdzielczość reprezentacji czasowej, częstotliwościowej, STFT oraz falkowej.



Rys. 3. Różnice w rozdzielczości w dziedzinie: czasowej, częstotliwościowej, transformaty STFT oraz falkowej.

Asymetria i nieregularne kształty niektórych falek naturalnie predysponują je do analizy sygnałów przejściowych, o gwałtownych zmianach i pikach. Sygnały takie, o wiele trudniej odwzorować za pomocą sinusoid. Ponieważ dla różnych falek uzyskuje się różne reprezentacje tego samego sygnału, kluczową sprawą jest wybór falki bazowej. Jest to bardzo ważny element całej analizy. Na Rys. 4 przedstawiono kilka najbardziej popularnych falek.



Rys. 4. Różne rodzaje falek, oraz odpowiadające im funkcje skalujące.

2. Ciągła transformata falkowa

Ciągła transformata falkowa (ang. *Continuous Wavelet Transform*, CWT) ma postać:

$$f_{\psi}(a,b) = \frac{1}{\sqrt{a}} \int_{-\infty}^{\infty} f(t) \psi\left(\frac{t-b}{a}\right) dt \quad (3)$$

gdzie: $\psi \in L^2(\mathbf{R})$, $f \in L^2(\mathbf{R})$, $b \in \mathbf{R}$, $a \in \mathbf{R}^+$.

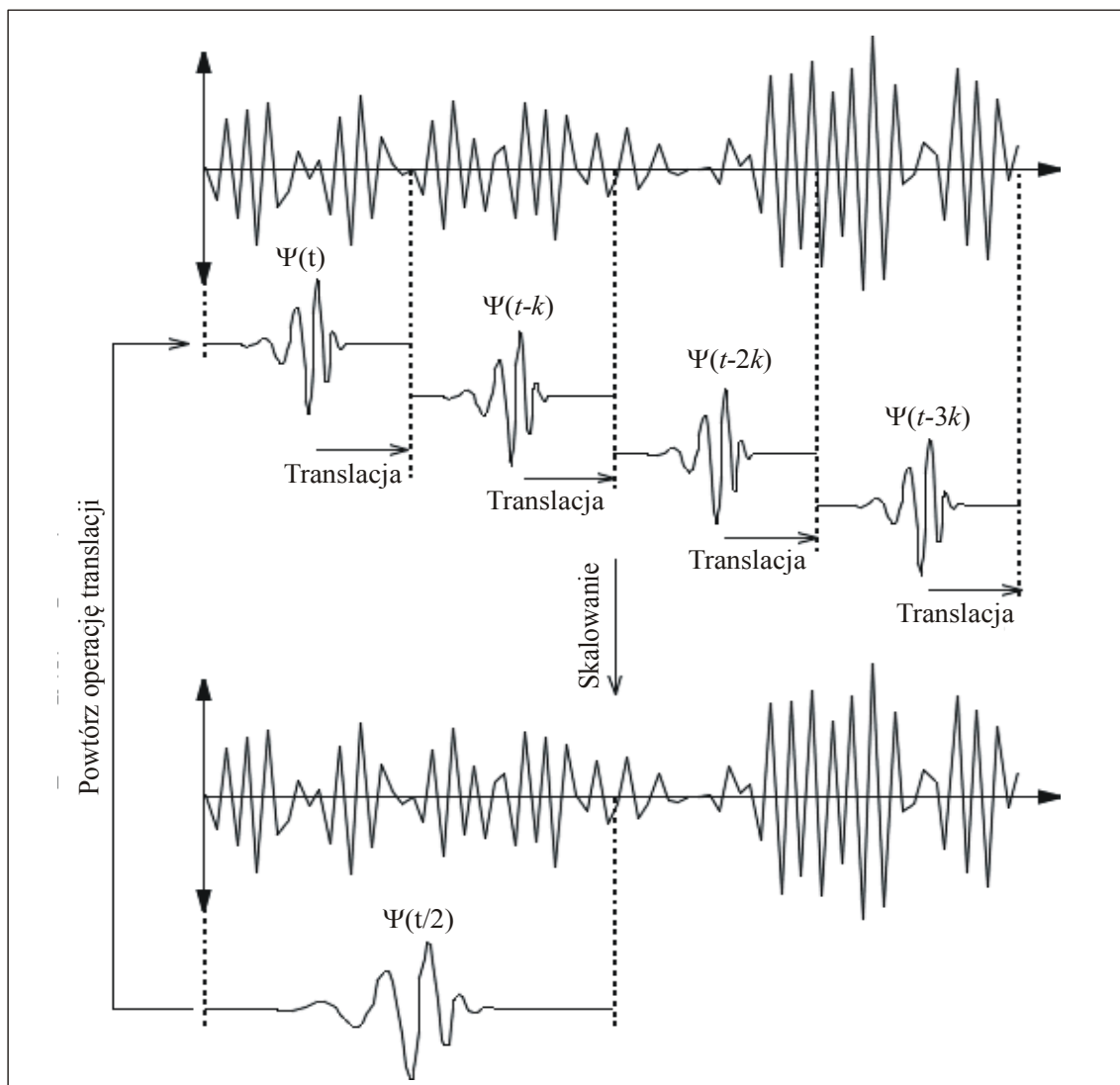
Jądrem transformaty falkowej jest falka podstawowa (bazowa) $\psi \in L^2(\mathbf{R})$ (ang. *mother wavelet*). Generalnie jest ona funkcją oscylującą, szybko zanikającą lub określoną na nośniku zwartym, o zerowej wartości średniej.

Rodzina falek jest tworzona z falki podstawowej poprzez jej przesunięcie, realizowane przez parametr translacji b oraz zmianę jej skali (częstotliwości) realizowanej przez parametr skalujący a :

$$\psi_{a,b}(t) = \frac{1}{\sqrt{a}} \psi\left(\frac{t-b}{a}\right) \quad (4)$$

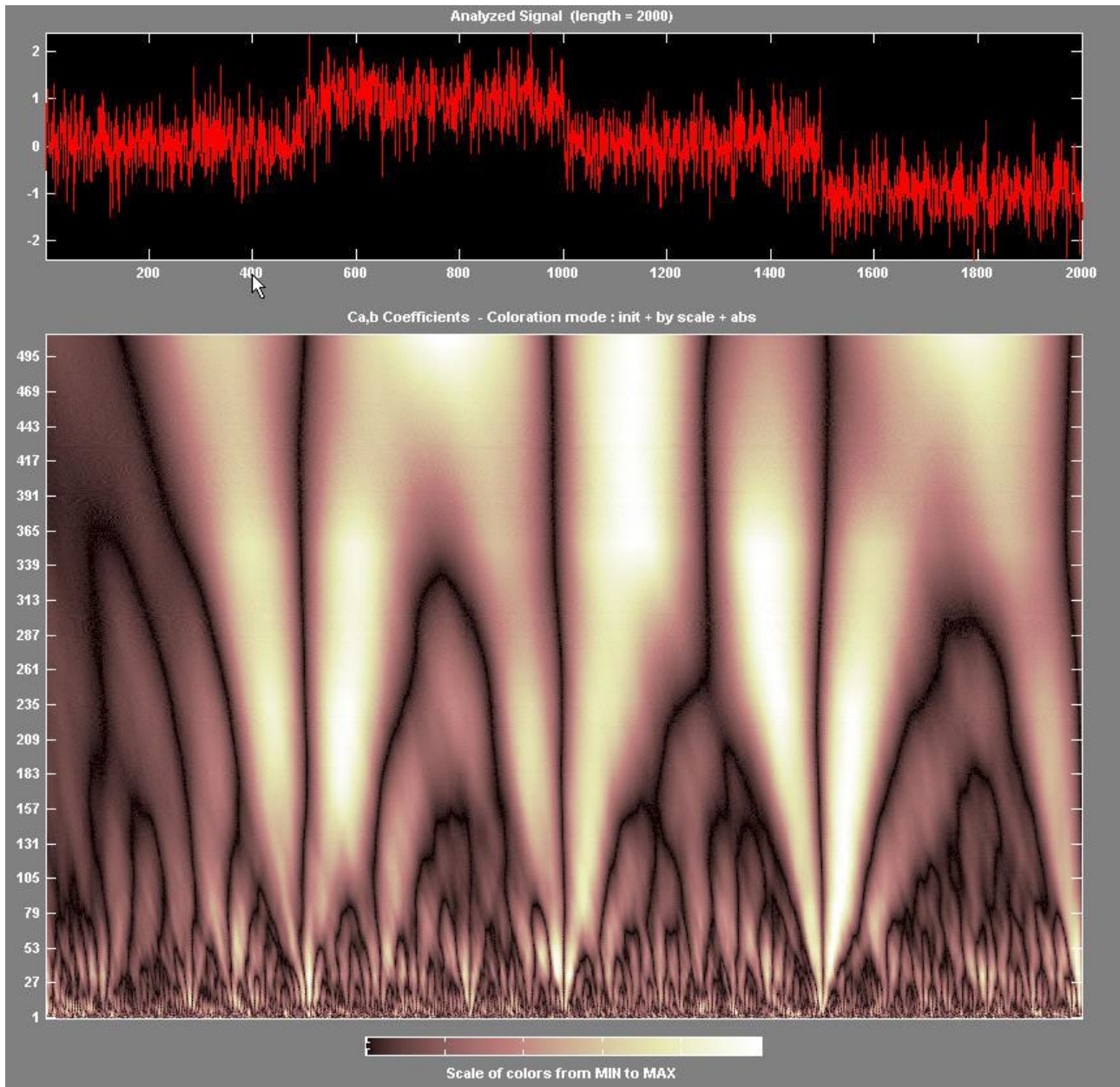
Czynnik normalizujący $a^{-1/2}$ sprawia, że wszystkie falki rodziny mają taką samą energię.

Wzór (3) produkuje współczynniki falkowe $f_{\psi}(a,b)$, które są funkcją skali a i przesunięcia b . Współczynnik falkowy jest miarą podobieństwa (korelacji) między sygnałem $f(t)$ a falką dla odpowiedniej skali i pozycji. Rys. 5 obrazuje procedurę obliczania tych współczynników. Falka jest przesuwana wzdłuż sygnału i dla każdej jej pozycji obliczany jest współczynnik falkowy. Po osiągnięciu końca sygnału, falka zostaje przeskalowana, powraca na początek sygnału i cała procedura się powtarza. Zestaw wszystkich współczynników $f_{\psi}(a,b)$ reprezentuje sygnał $f(t)$ w dziedzinie falkowej.



Rys. 5. Procedura obliczania współczynników falkowych dla CWT.

Istnieją różne metody graficznej reprezentacji ciągłej transformaty falkowej. Rys. 6 przedstawia jedną z nich: dwuwymiarową reprezentację współczynników falkowych w funkcji czasu i skali, gdzie kolor oznacza amplitudę współczynników. Analizowany jest tu sygnał skokowy, zakłócony szumem białym. W miejscach zmian wartości średniej (tj. w próbkach 500, 1000, 1500) wyraźnie widać pionowe czarne linie, które je identyfikują. Można zatem stosować tę transformatę jako detektor skokowych zmian w sygnale.



Rys. 6. Wizualizacja ciągłej transformaty falkowej z zastosowaniem falki db3.

3. Dyskretna transformata falkowa

Transformata falkowa jest przekształceniem dwuparametrowym, odwzorowującym jednowymiarowy sygnał $f(t)$ w dwuwymiarową tablicę współczynników $c_{j,k}$:

$$f(t) = \sum_{j,k} c_{j,k} 2^{j/2} \psi(2^j t - k) \quad (5)$$

gdzie zbiór *funkcji falkowych* $\psi_{j,k}(t)$ jest zazwyczaj bazą ortogonalną. Zbiór wszystkich współczynników $c_{j,k}$ jest nazywany **dyskretną transformatą falkową** sygnału $f(t)$, a wzór (5) jest transformatą odwrotną. Wprowadzając tzw. *funkcję skalującą* $\varphi_{j,k}$ sygnał $f(t)$ można przedstawić jako:

$$f(t) = \sum_{k=-\infty}^{\infty} a_k \varphi_k(t) + \sum_{j=0}^{\infty} \sum_{k=-\infty}^{\infty} d_{j,k} \psi_{j,k}(t) \quad (6)$$

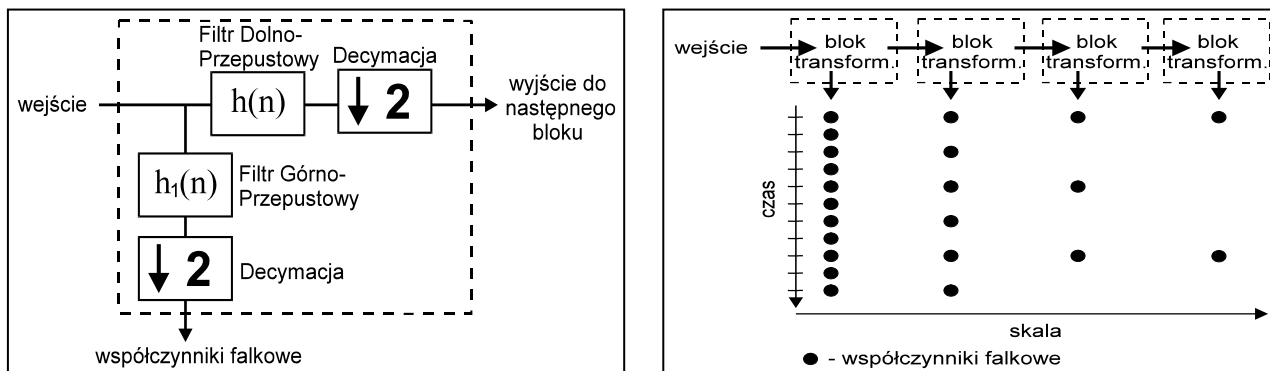
Pierwsza suma we wzorze (6) stanowi *aproksymację* sygnału, druga suma, wraz ze wzrostem j reprezentuje coraz bardziej szczegółowe *detale* sygnału.

Transformatę falkową można rozpatrywać w kategoriach filtrów cyfrowych. Jeżeli wyrazimy funkcję skalującą $\varphi(t)$ i falkę $\psi(t)$ jako sumę ważoną przesuniętych funkcji skalujących z wyższego poziomu $\varphi(2t)$:

$$\varphi(t) = \sum_n h(n)\sqrt{2} \varphi(2t - n) \quad (7)$$

$$\psi(t) = \sum_n h_1(n)\sqrt{2}\varphi(2t - n) \quad (8)$$

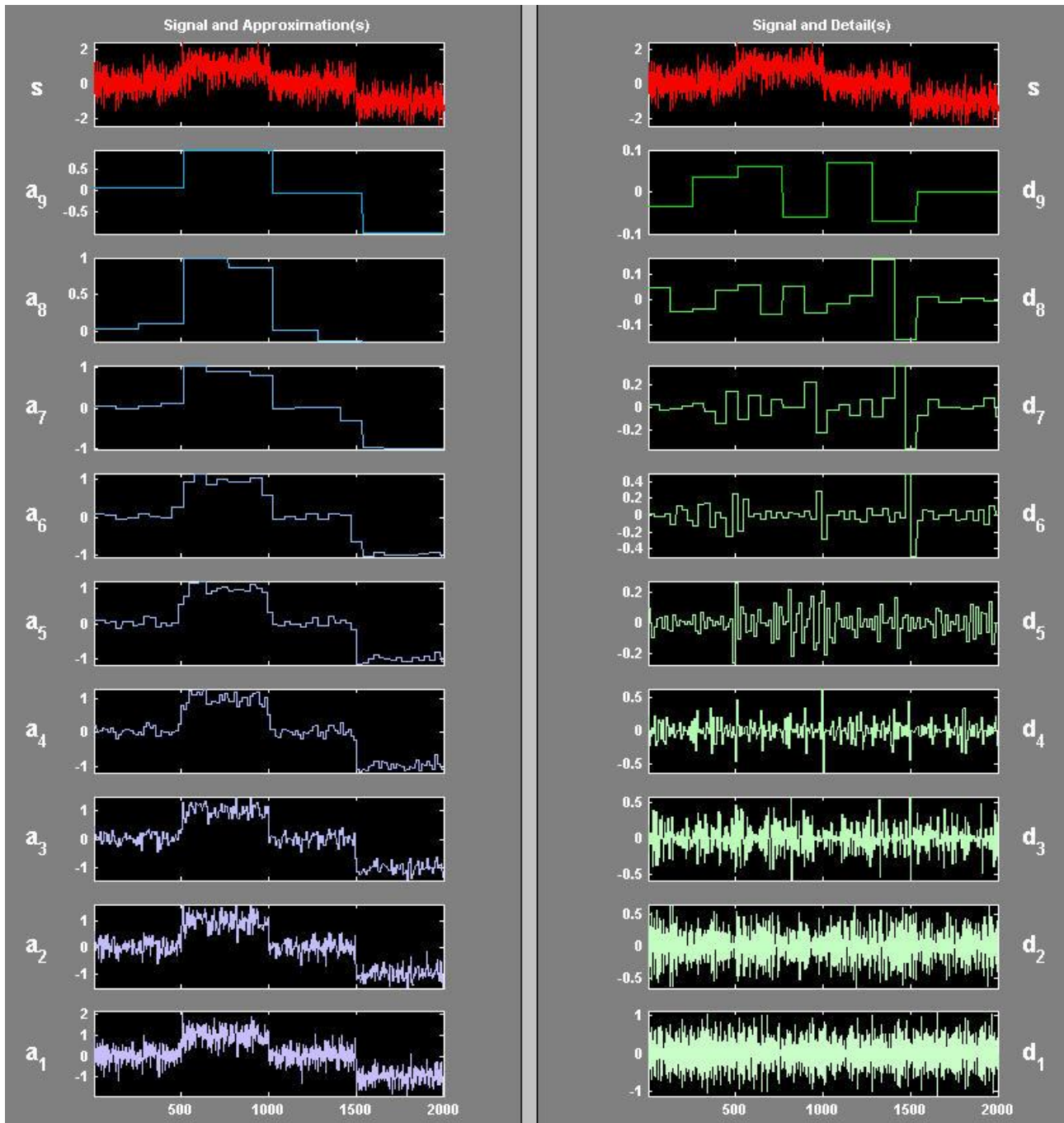
to $h(n)$ i $h_1(n)$ możemy traktować jako współczynniki cyfrowego filtra FIR.



Rys. 7. Pojedynczy blok transformacyjny (po lewej) oraz cały proces dekompozycji (po prawej).

Pojedynczy blok transformacyjny jest przedstawiony na Rys. 7. Filtr górno-przepustowy $h_1(n)$ reprezentuje falkę podstawową i na wyjściu produkuje współczynniki falkowe (detale sygnału). Filtr dolno-przepustowy $h(n)$ reprezentuje funkcję skalującą i na wyjściu daje aproksymację sygnału. Cały proces dekompozycji polega na przepływie sygnału przez identyczne bloki transformacyjne, tak, że wyjście z poprzedniego bloku jest wejściem do następnego bloku.

Podobnie jak w przypadku CWT, istnieje szereg możliwości graficznej reprezentacji DWT. Na Rys. 8 zostały pokazane przebiegi aproksymacji i detali dla tego samego sygnału, który był analizowanego za pomocą CWT na Rys. 6. Jak widać, sygnał aproksymacji na poziomie 9 (a_9) mógłby być z powodzeniem wykorzystany jako funkcja decyzyjna, dla celów rozpoznawania momentu zmiany wartości średniej zakłóconego sygnału.



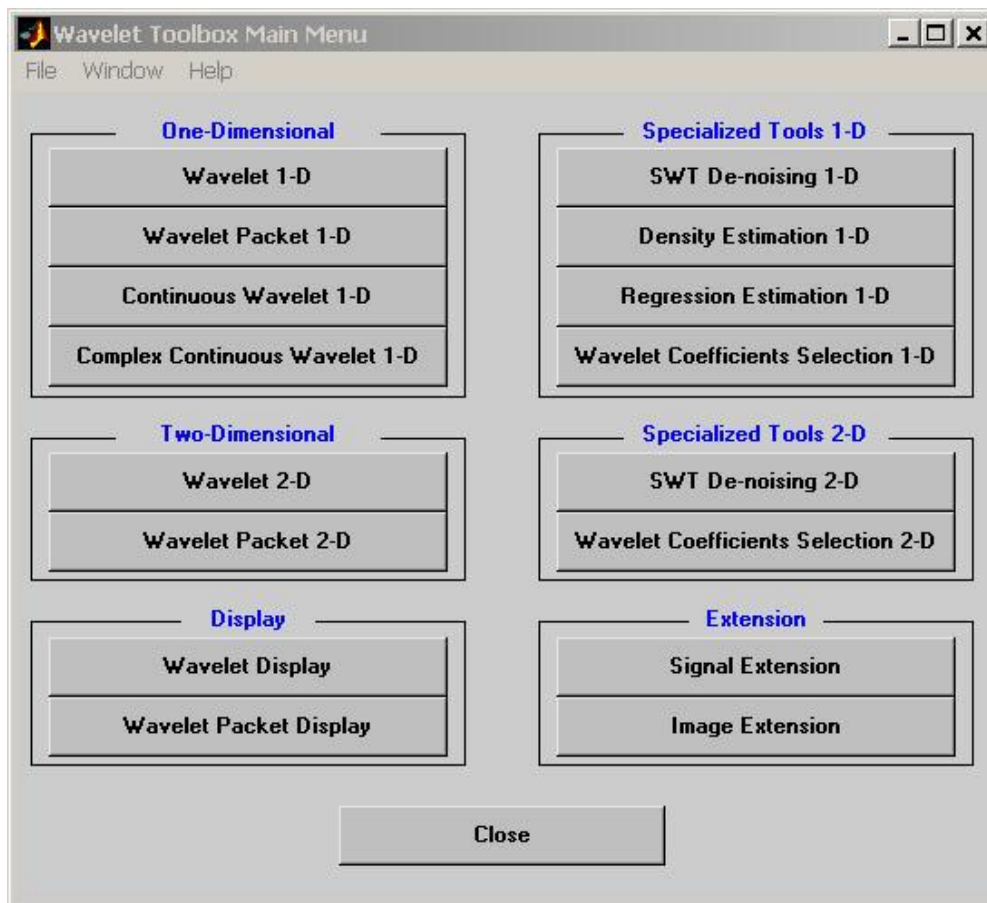
Rys. 8. Przebiegi aproksymacji i detali. Kolumna lewa zawiera analizowany sygnał (na górze) oraz kolejne jego aproksymacje (od a_1 do a_9). W kolumnie prawej analogicznie przedstawione są detale (od d_1 do d_9).

4. Matlab Wavelet Toolbox

Matlab posiada narzędzia do analizy falkowej zgromadzone w Wavelet Toolbox. Toolbox ten w postaci graficznego interfejsu uruchamia się wpisując w linii komend:

```
wavemenu
```

Pojawia się wtedy następujące okno (Rys. 9):



Rys. 9. Menu główne Wavelet Toolbox

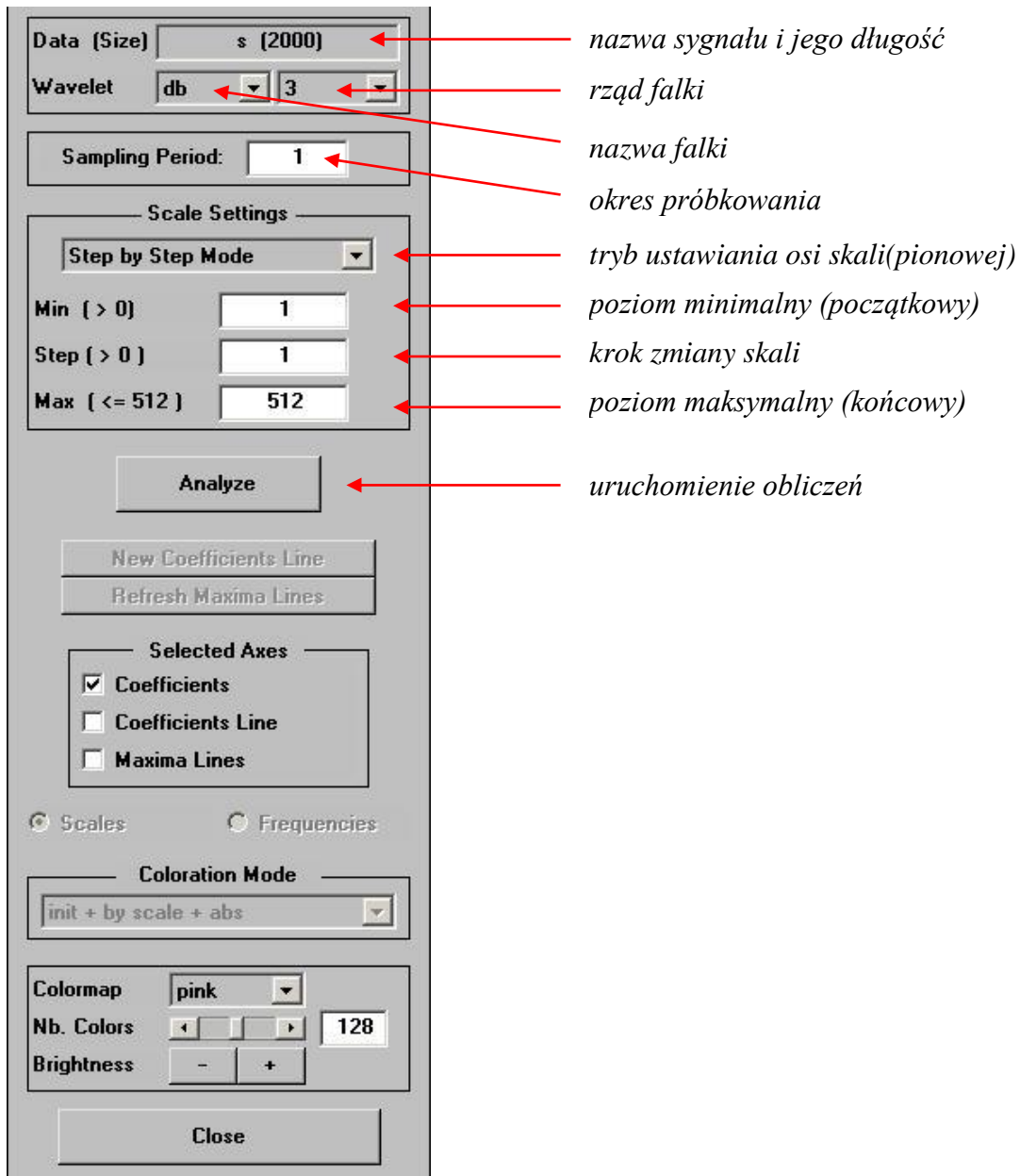
Transformata ciągła znajduje się pod przyciskiem *Continuous Wavelet 1-D*, a transformata dyskretna pod przyciskiem *Wavelet 1-D*.

Sygnal do analizy należy zapisać w formacie mat-pliku, za pomocą polecenia *save*:

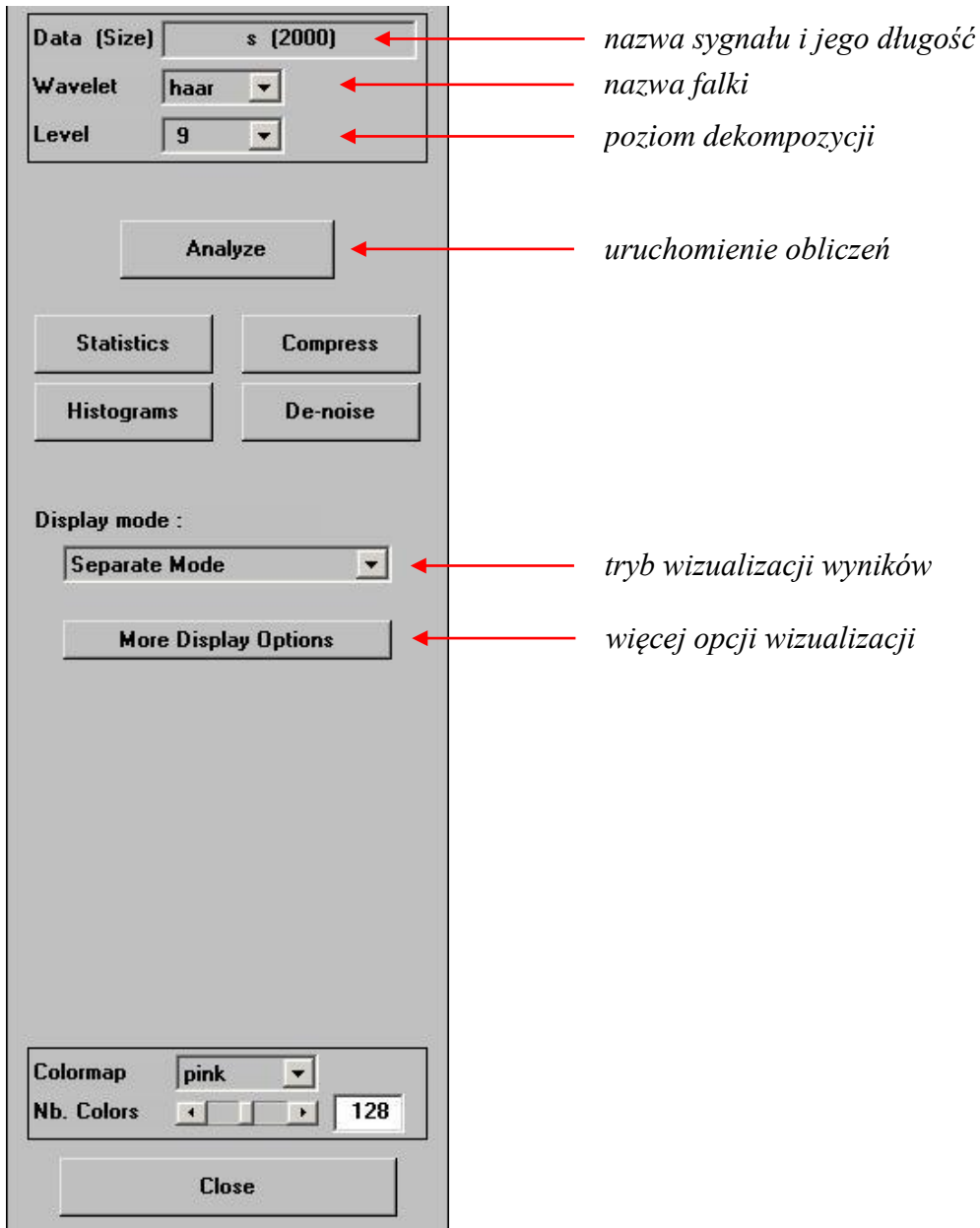
```
save nazwa_pliku nazwa_zmiennej
```

W wyniku powstanie sygnał *nazwa_pliku.mat* zawierający zmienną *nazwa_zmiennej*. Następnie w oknie *Continuous Wavelet 1-D* lub *Wavelet 1-D* należy kliknąć menu *File* → *Load Signal* i wybrać *nazwa_pliku.mat*.

Na Rys. 10 znajduje się opis parametrów ciągłej transformaty falkowej dla okna *Continuous Wavelet 1-D* (takie ustawienia zostały zastosowane na Rys. 6). Natomiast na Rys. 11 znajduje się opis parametrów dyskretny transformaty falkowej dla okna *Wavelet 1-D* (takie ustawienia zostały zastosowane na Rys. 8).



Rys. 10. Parametry transformacji okna *Continuous Wavelet 1-D*.



Rys. 11 Parametry transformacji okna Wavelet 1-D.

Zamiast korzystać z interfejsu graficznego, można napisać odpowiedni program w Matlabie. Poniżej znajduje się przykładowy kod realizujący obliczanie i wyświetlanie aproksymacji i detali dla dyskretnej transformaty falkowej (uzyskujemy efekt podobny do Rys. 8).

```
% Transformata Falkowa
% Obliczanie detali i aproksymacji sygnału

clear

% wygenerowanie sygnału testowego
N = 2000;
war = 0.3;
std = sqrt(war);
s = skoki(N,[0 1 0 -1]);
sygnał = s + std*randn(N,1);

% transformata
poziom=9; %maksymalny poziom dekompozycji
falka='haar'; %rodzaj falki
[C,L] = wavedec(sygnał,poziom,falka); %dyskretna transformata falkowa
a9=wrcoef('a',C,L,falka,9); %aproksymacja na poziomie 9
d9=wrcoef('d',C,L,falka,9); %detale na poziomie 9
a8=wrcoef('a',C,L,falka,8); %aproksymacja na poziomie 8
d8=wrcoef('d',C,L,falka,8); %detale na poziomie 8
%...
% analogicznie można obliczać detale i aproksymacje na kolejnych
poziomach (niższych)
%...
% wizualizacja
subplot(3,1,1),plot(sygnał),title('Sygnał'),axis([0 length(s) min(s)
max(s)])
subplot(3,2,3),plot(a9),ylabel('a9'),title('Aproksymacje')
subplot(3,2,4),plot(d9),ylabel('d9'),title('Detale')
subplot(3,2,5),plot(a8),ylabel('a8')
subplot(3,2,6),plot(d8),ylabel('d8')
%...
% analogicznie dla kolejnych poziomów dekompozycji
%...
```

5. Wykonanie ćwiczenia

Wygenerować sygnał ze skokową zmianą wartości średniej i sygnał ze skokową zmianą wariancji, zakłócony szumem białym.

- Dla różnych parametrów sygnałów testowych (wartości zakłócenia, poziomu skoku, wariancji, itp.) i różnych parametrów transformaty (rodzaju falki, rzędu falki, poziomu dekompozycji) wykonać kilka prób ciągłej transformaty falkowej i wyciągnąć wnioski w kwestii przydatności tej transformaty do detekcji nagłych zmian w sygnale.
- Zbadać przydatność aproksymacji w dyskretnej transformacie falkowej jako funkcji decyzyjnej, dla celów rozpoznawania momentu zmiany wartości średniej zakłóconego sygnału (rodzaj i rząd falki, poziom dekompozycji, itp. należy dobrać empirycznie). Wykonać badania statystyczne dla populacji 100-500 sygnałów testowych. Można się

oprzeć na przedstawionym powyżej kodzie Matlab i poniższej funkcji do generowania skoków

```
function [y,kdl] = skoki(N,amplitudy)

% generowanie sygnału złożonego ze skoków

% Wejścia:
% N - długość sygnału
% amplitudy - amplitudy skoków
% Np. amplitudy = [0 10 0 -15 -5]
% Wyjścia:
% y - sygnał skokowy
% kdl - długość skoku

k=length(amplitudy); %liczba skoków
kdl=round(N/k);
y=ones(kdl,k);
for i=1:k
    y(:,i)=y(:,i)*amplitudy(i);
end
y=y(:);

%wyrównanie sygnału do długości N
N2=length(y);
while(N~=N2)
    if N2 > N
        y(end)=[];
    else
        y(end+1)=y(end);
    end
    N2=length(y);
end
```