Sprawozdanie z laboratorium 1. z przedmiotu TRA prowadzonego w semestrze 24Z

Piotr Sienkiewicz 324 887

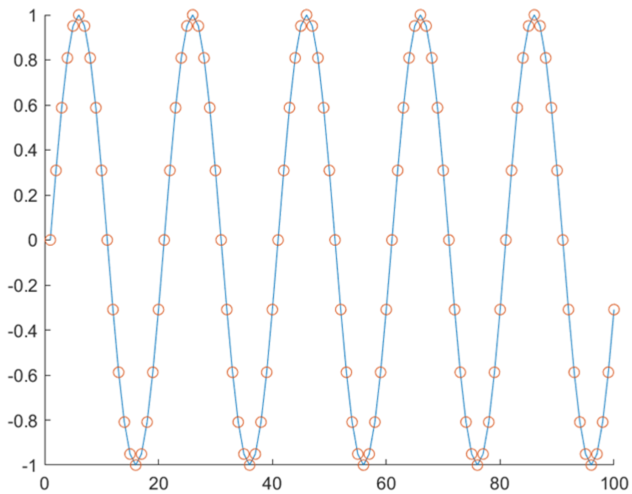
1 Zadanie 1

Napisano następujący skryp w MATLABie do wykonania zadania pierwszego:

Listing 1: Skrypt MATLAB do zadania 1.

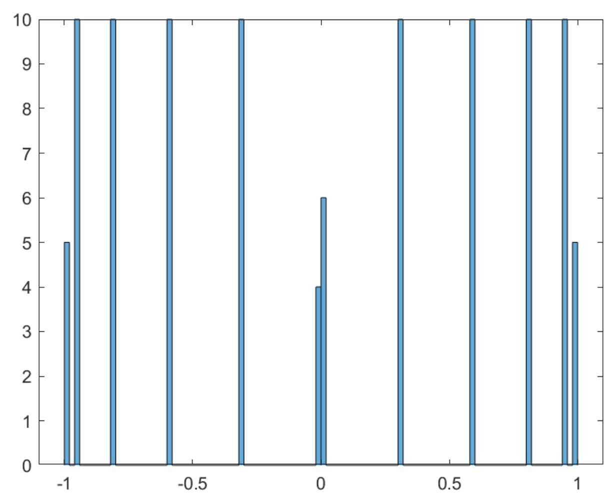
```
%% Zad1
clear variables
clc

%% parametry
A = 1;
phi = 0;
f = 0.1;
N = 100;
fs = 2;

q = [0.01 0.1 1];

%% rozwiazanie zadania
x = gensinsum(A, phi, f, N, fs);
y1 = sample(x, N);

figure
hold on
plot(x) % rysowanie wykresu sygnalu x
scatter(1:N,y1) % nanoszenie na wykres punktow probkowania
hold off

figure
histogram(x, 100);

for i=1:length(q)
    e = x - quant(x, q(i)); % liczenie bledu kwantyzacji
    figure
    histogram(e, 100);
    title(['q = ' num2str(q(i))])
end
```

W wyniku działania programu otrzymano wykres sygnału sinusoidalnego z zaznaczonymi próbkami:



Rysunek 1: Sygnał z naniesionymi próbkami

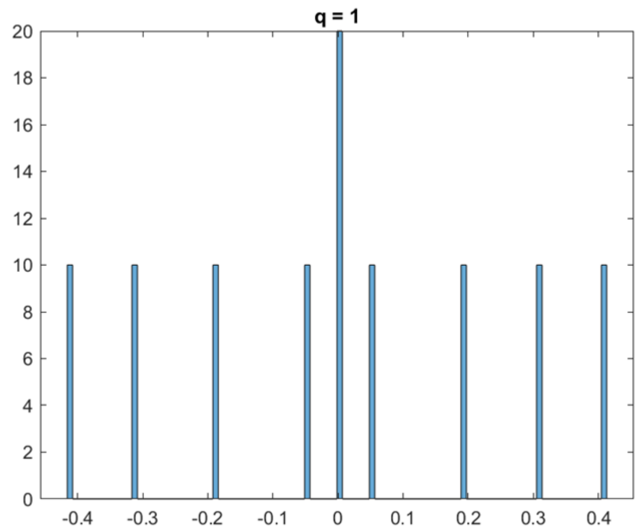
Histogram wartości sygnału sinusoidalnego:



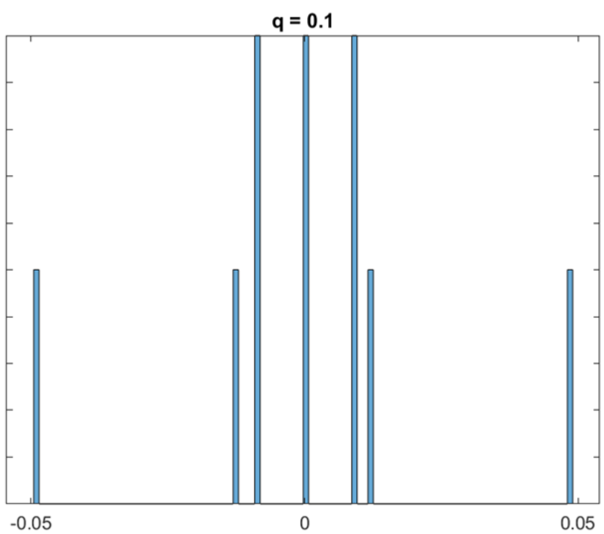
Rysunek 2: Histogram wartości sygnału x

Histogram ten nie przypomina typowego histogramu wartości sygnału sinusoidalnego ze względu na małą liczbę próbek N. Według polecenia histogram został narysowany dla 100 przedziałów wartości, co przy $N = 100$ sprawiło, że nie wszystkie wartości na histogramie zostały obsadzone ze względu na skorelowanie częstotliwości próbkowania i częstotliwości unormowanej. Z tego powodu niektóre wartości sygnału zdyskretyzowanego się powtarzały.

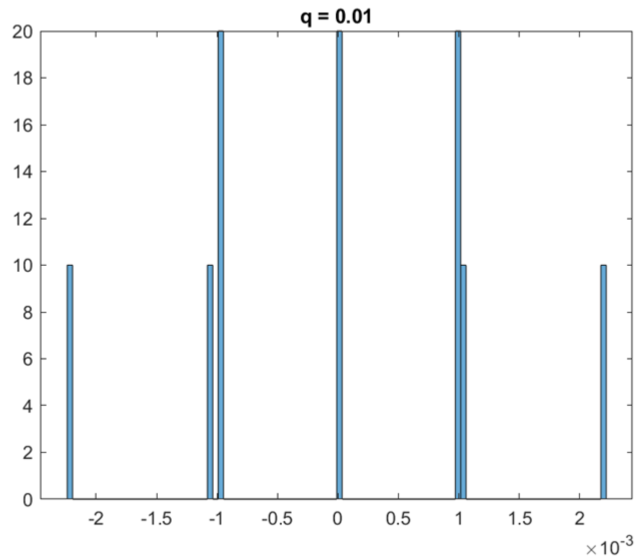
Dodatkowo otrzymano 3 histogramy błędu kwantyzacji:



Rysunek 3: Histogram błędu kwantowania dla $q = 1$



Rysunek 4: Histogram błędu kwantowania dla $q = 0,1$



Rysunek 5: Histogram błędu kwantowania dla $q = 0,01$

W przypadku błędu kwantyzacji także na histogramach zostały obsadzone tylko niektóre zakresy, co także wynika ze szczególnego przypadku w tym zadaniu, w którym częstotliwość próbkowania jest całkowitą wielokrotnością częstotliwości sygnału. Z tego powodu „miejsca" próbkowań sinusoidy powtarzają się okresowo.

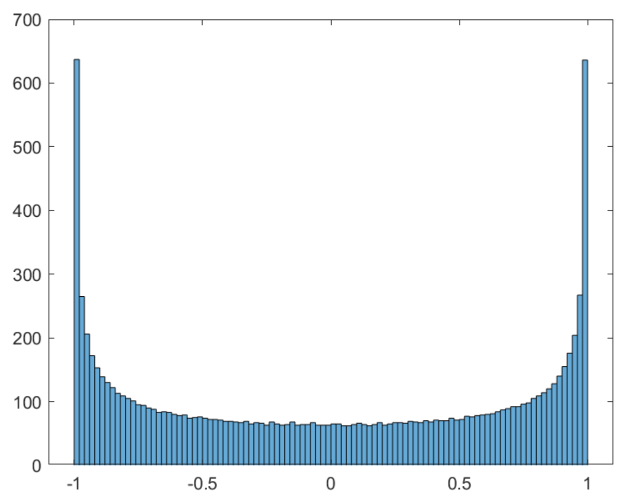
2 Zadanie 2

Listing 2: Skrypt MATLAB do zadania 2.

```
%% Zad2
clear variables
clc

%% parametry
A = 1;
phi = 0;
f = 0.13254234626165;
N = 10000;
fs = 2;

q = [0.0001 0.01 0.1 1];

%% rozwi zanie zadania
x = gensinsum(A, phi, f, N, fs);

figure
histogram(x, 100);

for i=1:length(q)
    e = x - quant(x, q(i));
    figure
    histogram(e,100);
    title(['q = ' num2str(q(i))])

```

```
    cov_e = cov(e);  % kowariancja bledu kwantyzacji
    theoretical_variance = q(i)^2/12;  % teoretyczna wariancja kwantyzacji

    % wyswietlanie wynikow
    fprintf(['Kowariancja b   d u kwantyzacji: %.8f; ' ...
        'Wariancja q^2/12: %.8f\n'], ...
        cov_e, theoretical_variance);
end
```

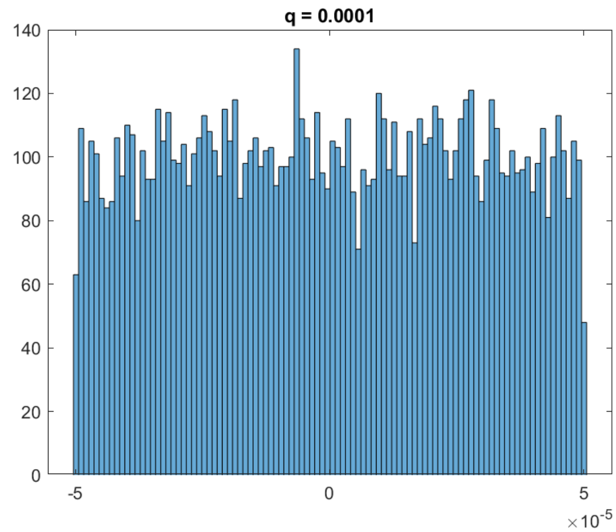
W tym przypadku liczba próbek wynosiła $N = 10000$, natomiast histogramy były nadal rysowane dla liczby przedziałów równej 100. Stąd też histogram wartości sygnału sinusoidalnego wygląda poprawnie:



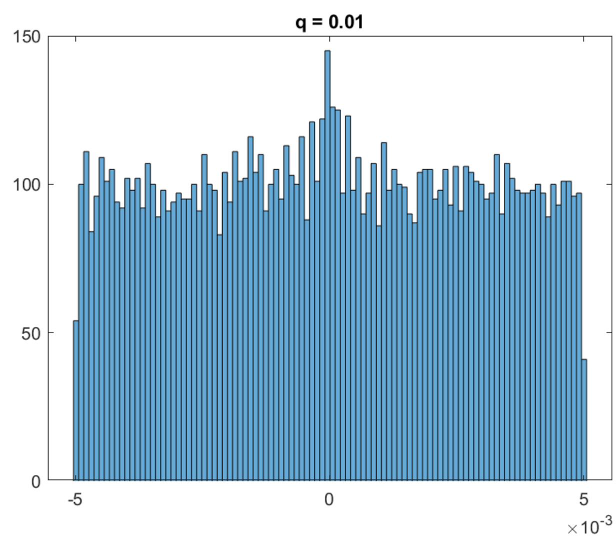
Rysunek 6: Histogram wartości sygnału x

Jego kształt wynika z tego, że sinusoida w okolicach swoich wierzchołku zmienia się najwolniej, a więc największa ilość próbek przyjmuje właśnie wartości z tego zakresu.

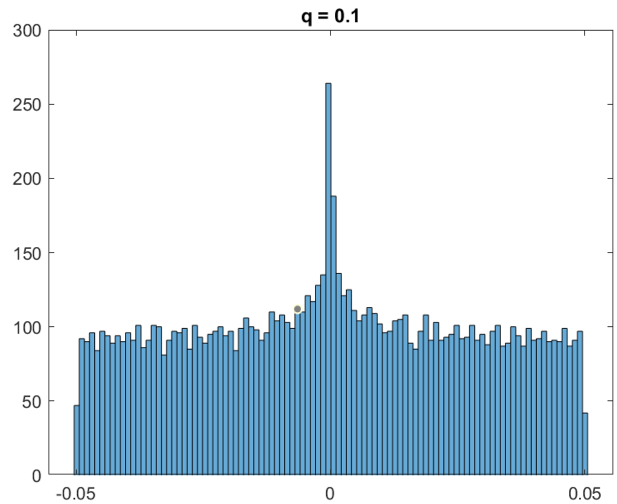
Tym razem wygenerowano dodatkowo jeszcze histogram błędu kwantyzacji dla wartości kwantu $q = 0,0001$:



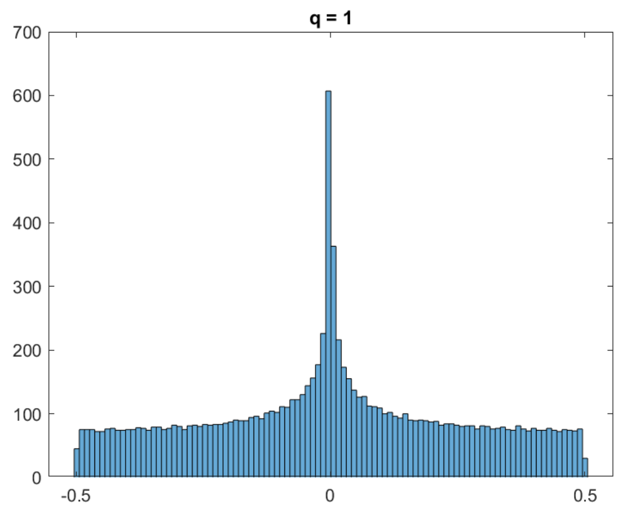
Rysunek 7: Histogram błędu kwantowania dla $q = 0.0001$



Rysunek 8: Histogram błędu kwantowania dla $q = 0.01$



Rysunek 9: Histogram błędu kwantowania dla $q = 0.1$



Rysunek 10: Histogram błędu kwantowania dla $q = 1$

Tym razem przez to, że częstotliwość próbkowania nie jest całkowitą wielokrotnością częstotliwości sygnału, wszystkie przedziały na histogramach zostały obsadzone. Jest to logiczne, bo tym razem próbkowanie nie zachodzi w tych samych powtarzalnych miejscach sinusoidy, lecz miejsca te są rozłożone losowo.
Dla bardzo dużego kwantu $q = 1$ oraz $q = 0,1$ na histogramie jest dominująca wartość 0. Jest to związane z tym, że dla sinusa najwięcej próbek ma wartości z wierzchołków sygnału, a tam błąd kwantowania dla tak dużych kwantów jest jednym z najmniejszych. Dla bardzo małych kwantów, rozkład błędu kwantowania coraz bardziej przypomina rozkład ciągły, co jest słuszne, bo szum kwantowania w idealnym przypadku jest to szum biały.

Dodatkowo skrypt zwrócił następujący wynik:

Listing 3: Wynik działania skryptu z zadania 2.

```
Kowariancja bledu kwantyzacji: 0.00000000; Wariancja q^2/12: 0.00000000
```

```
Kowariancja bledu kwantyzacji: 0.00000812; Wariancja q^2/12: 0.00000833
Kowariancja bledu kwantyzacji: 0.00077023; Wariancja q^2/12: 0.00083333
Kowariancja bledu kwantyzacji: 0.06402240; Wariancja q^2/12: 0.08333333
```

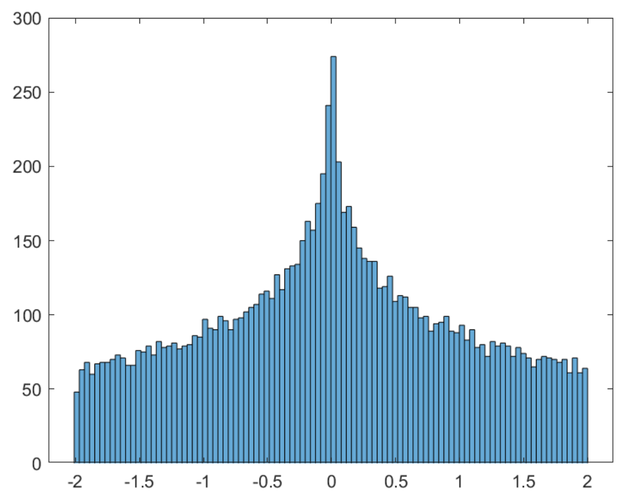
Jak widać, kowariancja błędu kwantyzacji wyznaczona na podstawie rzeczywistego sygnału nie odbiega za mocno od wariancji wyliczonej ze wzoru teoretycznego. W rzeczywistości kowariancja zależy od kilku czynników, nie tylko od błędów kwantowania

3 Zadanie 3

W tym zadaniu w skrypcie nastąpiła tylko ta zmiana, że parametry sygnału podano jako wektor, a częstotliwości sygnałów składowych to parametr f_c pomnożony przez odpowiednie niecałkowite współczynniki:

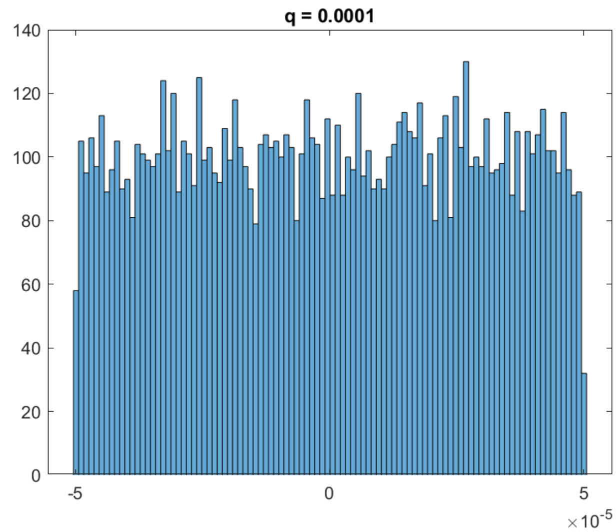
Listing 4: Skrypt MATLAB do zadania 3.

```
%% parametry
A = [1; 1];
phi = [0; 0];
fc = 0.1;
f = fc * [0.9912331; 1.00323];
N = 10000;
fs = 2;

q = [0.0001 0.01 0.1 1];
end
```

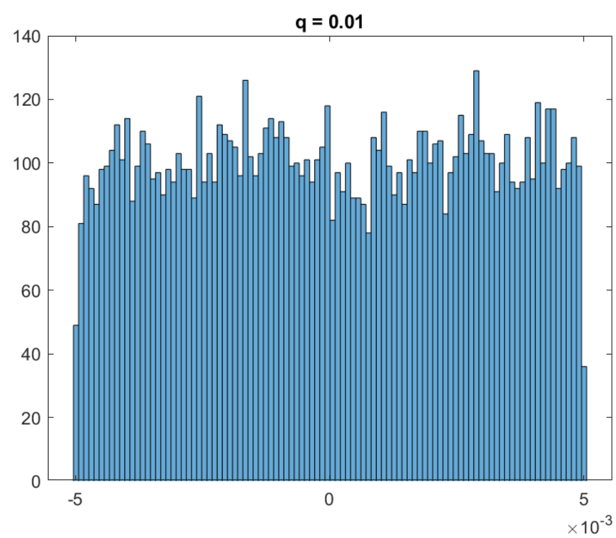


Rysunek 11: Histogram wartości sygnału x

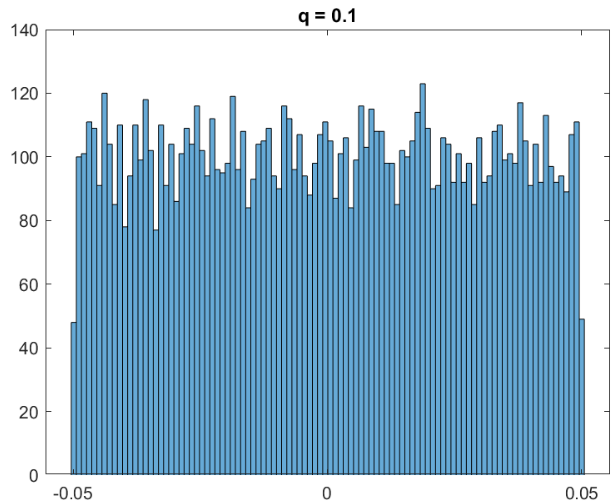
Tym razem wartości sygnału x mają inny rozkład niż w poprzednim zadaniu. Wynika to z tego, że sygnał jest wolnozmienny, a więc bardzo dużo próbek ma wartości bliskie 0, stąd "pik"na histogramie. Druga różnicą jest to, że większe jest też obsadzenie pozostałych zakresów wartości na histogramie, co wynika z tego, że sygnał po zsumowaniu nie jest już czystą sinusoidą, lecz ma nieregularny kształt.



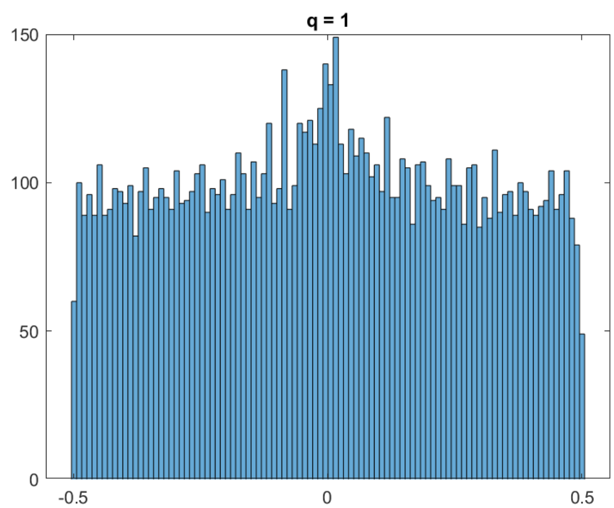
Rysunek 12: Histogram błędu kwantowania dla $q = 0.0001$



Rysunek 13: Histogram błędu kwantowania dla $q = 0.01$



Rysunek 14: Histogram błędu kwantowania dla $q = 0.1$



Rysunek 15: Histogram błędu kwantowania dla $q = 1$

W tym przypadku rozkład błędu kwantyzacji jest bardziej jednolity i mało zależy od wielkości kwantu. Wynika to z tego, że po zsumowaniu obu sinusiod, sygnał nie jest już sinusoidą, zawiera dodatkowe prążki częstotliwości przez co zakresy wartości są obsadzane bardziej losowo.

Wartości kowariancji i wariancji błędu kwantyzacji są następujące:

Listing 5: Wynik działania skryptu z zadania 3.

```
Kowariancja bledu kwantyzacji: 0.00000000; Wariancja q^2/12: 0.00000000
Kowariancja bledu kwantyzacji: 0.00000828; Wariancja q^2/12: 0.00000833
Kowariancja bledu kwantyzacji: 0.00083095; Wariancja q^2/12: 0.00083333
Kowariancja bledu kwantyzacji: 0.07897309; Wariancja q^2/12: 0.08333333
```

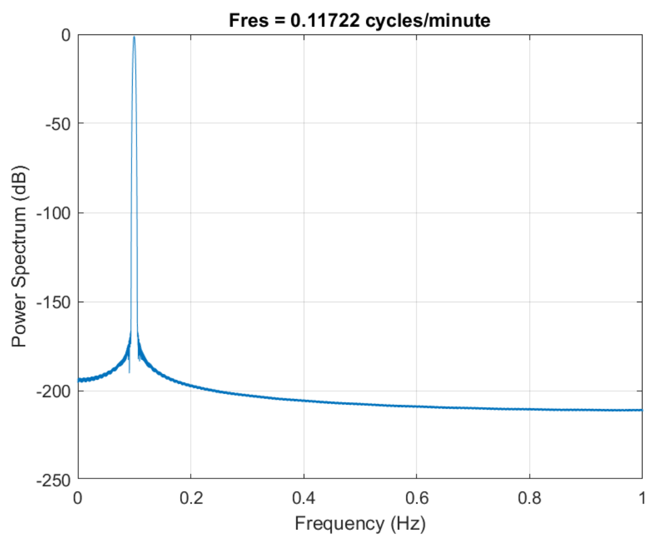
Jak widać wariancja się nie zmieniła, wszakże cały czas używane były te same wielkości kwantów. Natomiast wartość kowariancji stała się bliższa wartości wariancji wyliczonej ze wzoru.

4 Zadanie 4

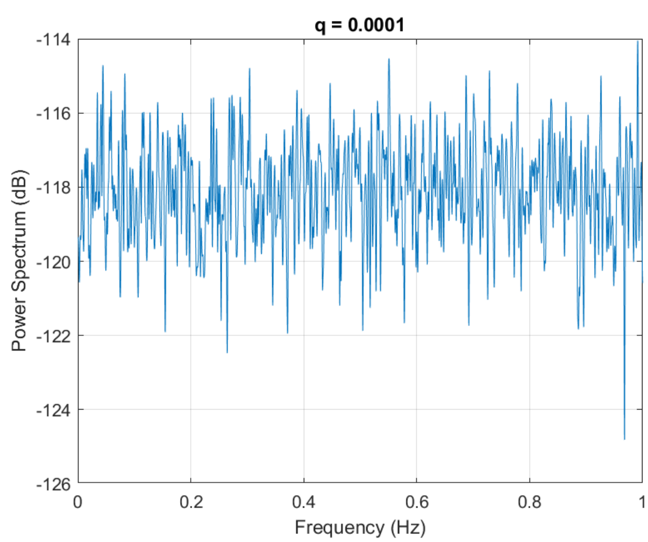
W zadaniu tym dodano fragment kodu odpowiadający za wygenerowanie widma sygnału oraz widma błędu kwantyzacji:

Listing 6: Fragment kodu odpowiadający za wyznaczanie widma szumu kwantyzacji

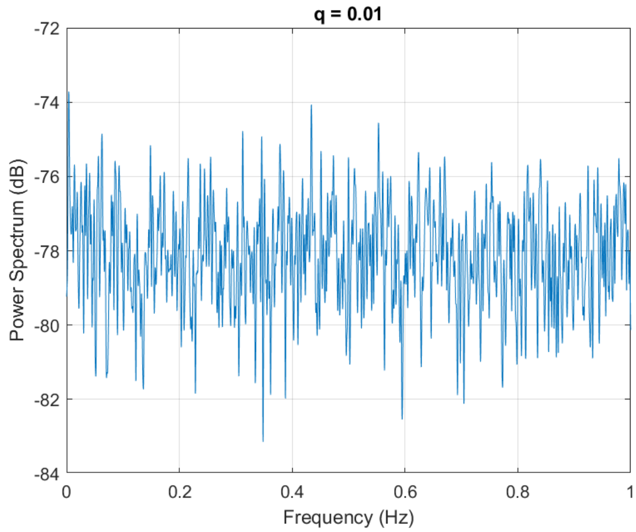
```
figure;
pspectrum(e, fs, 'power');
title(['q = ' num2str(q(i))])
```



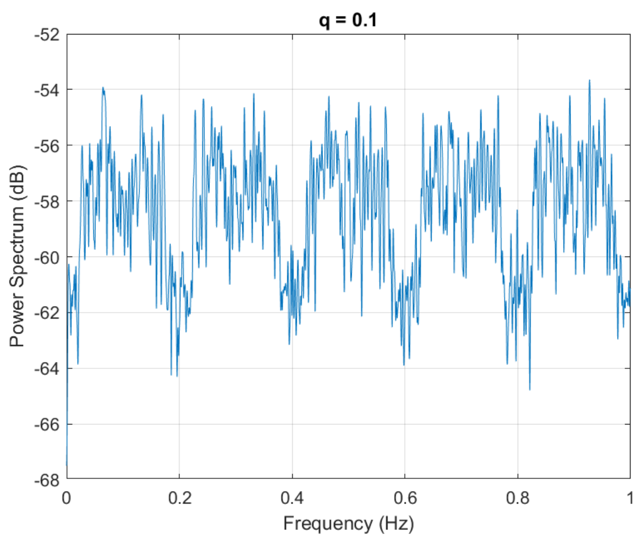
Rysunek 16: Widmo sygnału x



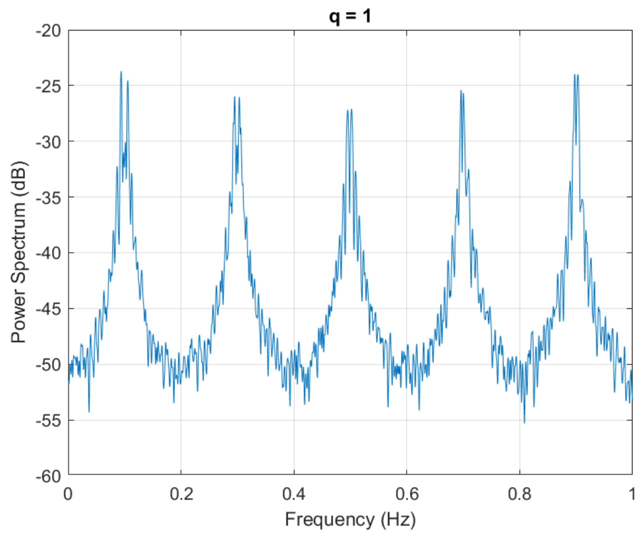
Rysunek 17: Widmo błędu kwantyzacji dla $q = 0,0001$



Rysunek 18: Widmo błędu kwantyzacji dla $q = 0,01$



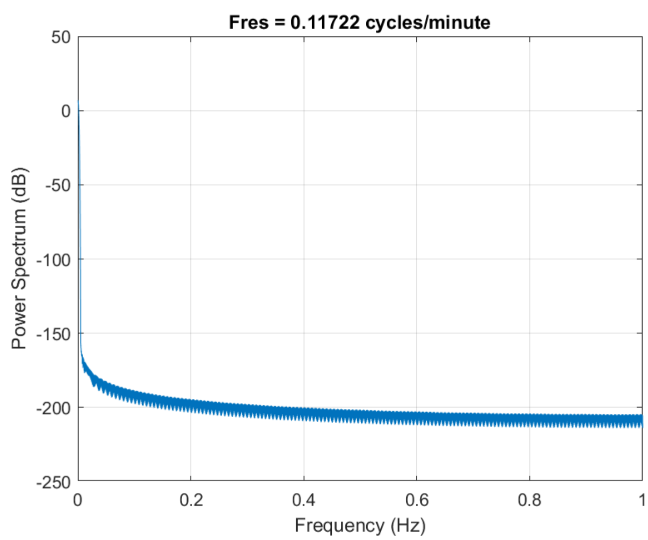
Rysunek 19: Widmo błędu kwantyzacji dla $q = 0,1$



Rysunek 20: Widmo błędu kwantyzacji dla $q = 1$

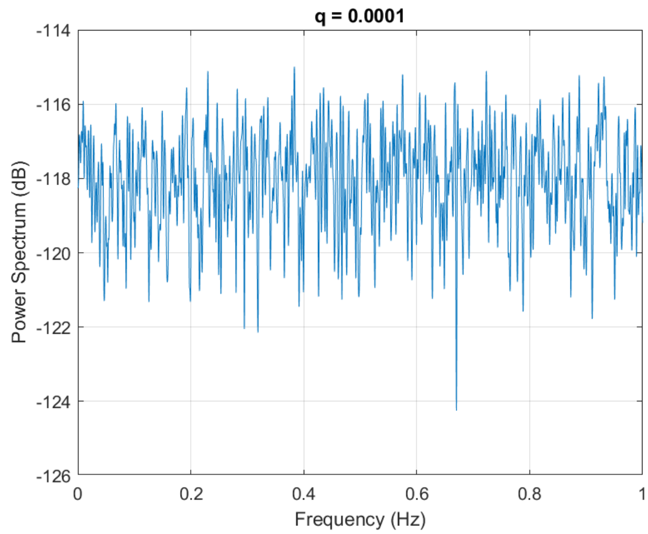
Jak widać, dla małych wartości kwantu widmo błędu kwantyzacji bardziej przypomina szum biały i ma małe wartości.

Tym razem wykonano doświadczenie dla $f_c = 0,0001$. Widma sygnału oraz szumu kwantyzacji są następujące:

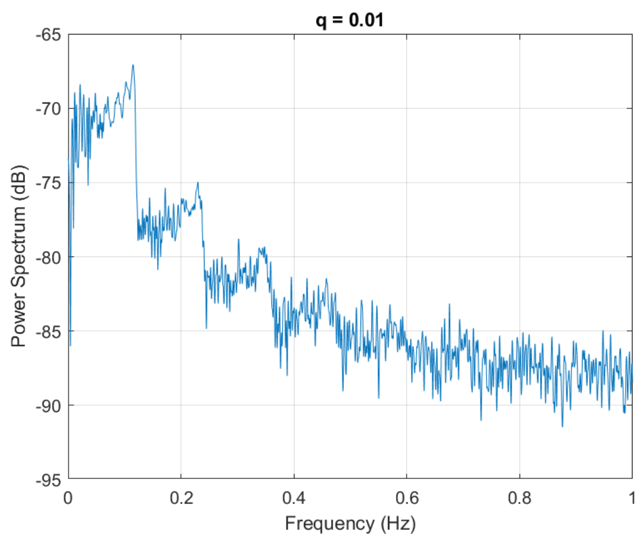


Rysunek 21: Widmo sygnału x

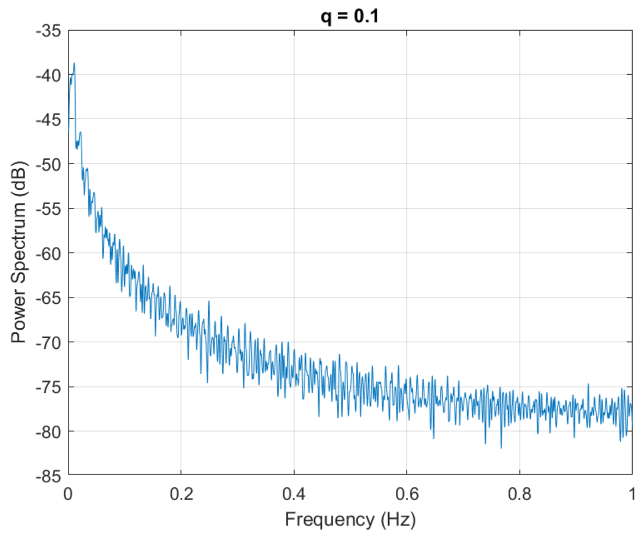
Wystąpiło tutaj bardzo duże nadpróbkowanie, które sprawia, że poziom szumów jest niezwykle mały.



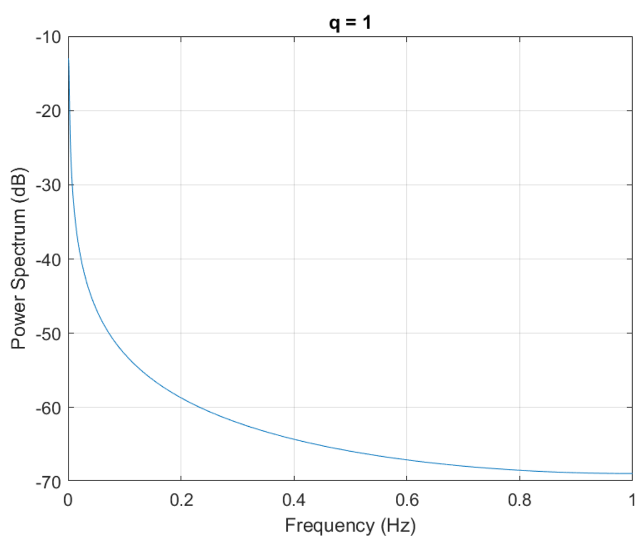
Rysunek 22: Widmo błędu kwantyzacji dla $q = 0,0001$



Rysunek 23: Widmo błędu kwantyzacji dla $q = 0,01$



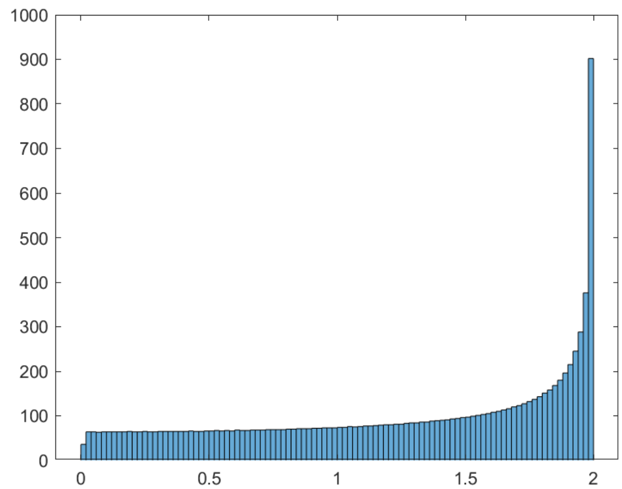
Rysunek 24: Widmo błędu kwantyzacji dla $q = 0,1$



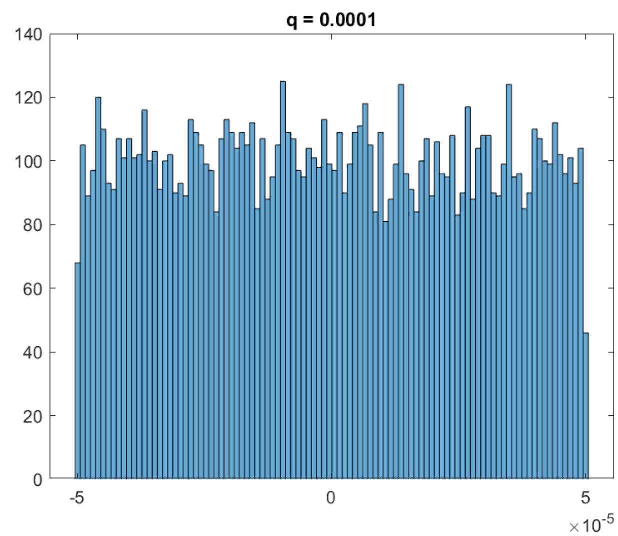
Rysunek 25: Widmo błędu kwantyzacji dla $q = 1$

W tym przypadku widma błędów kwantyzacji wyglądają lepiej i wartości mocy szumów są mniejsze.

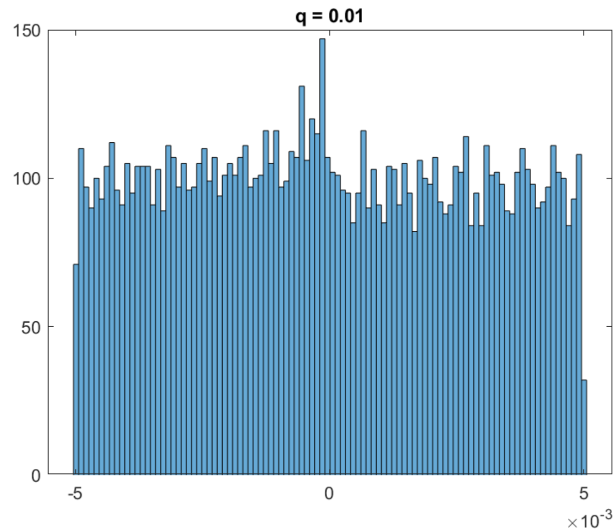
Wykonano dodatkowo histogramy sygnałów dla tej częstotliwości. Tym razem otrzymano pewne anomalie wynikające z tego, że badany sygnał ma bardzo małą częstotliwość i w zasadzie jest bliski składowej stałej.



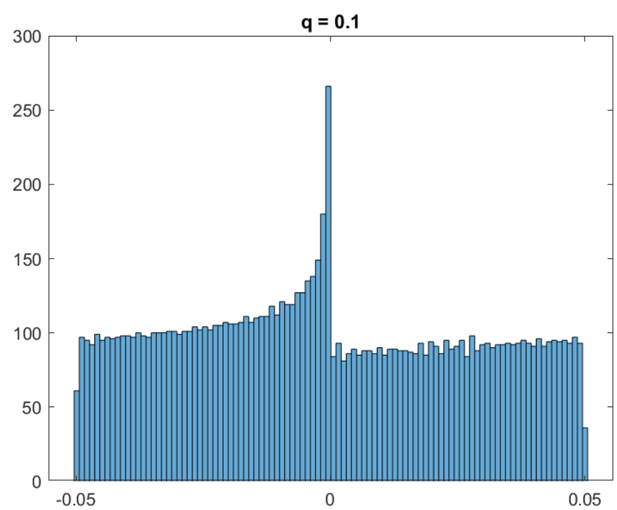
Rysunek 26: Histogram wartości sygnału x



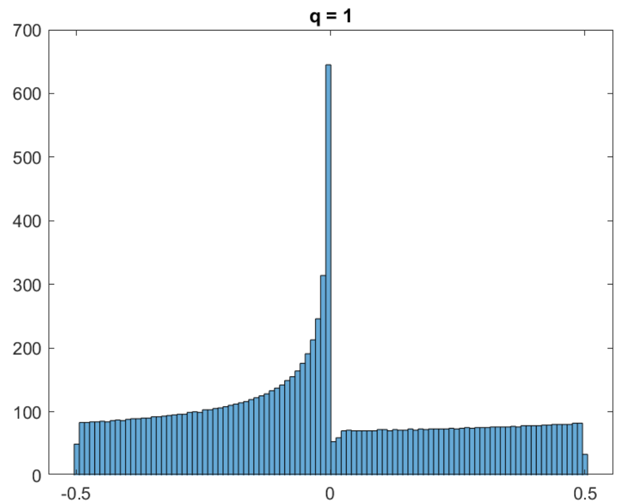
Rysunek 27: Histogram błędu kwantowania dla $q = 0.0001$



Rysunek 28: Histogram błędu kwantowania dla $q = 0.01$



Rysunek 29: Histogram błędu kwantowania dla $q = 0.1$



Rysunek 30: Histogram błędu kwantowania dla $q = 1$

5 Zadanie 5

Napisano poniższy skrypt:

Listing 7: Skrypt MATLAB do zadania 5.

```
%% Zad5
clear variables
clc

%% parametry
M = [4 9 16];
A = [1; 1];
phi = [0; 0];
fs = 1;
q = 0.1;

%% rozwiazanie zadania
for m=M
    fc = 0.1/m;
    f = fc * [0.9912331; 1.00323];
    N = 10000*m;

    x = gensinsum(A, phi, f, N, fs);

    [b,a] = butter(5,(1/m)*fs/2);

    y1 = filter(b,a,quant(x,q));
    y2 = filter(b,a,x);

    e = y1-y2;

    qeff = sqrt(12*cov(e));

    fprintf(['M: %.0f; ' ...
```

```
        'Kwant efektywny: %.8f\n'], ...
        m, qeff);
end
```

Dzięki podzieleniu f_c przez M oraz pomnożeniu liczby próbek razy M, uzyskano ten sam czas sygnału dla różnych nadpróbkowań. Program zwórcił następujący wynik:

Listing 8: Wynik działania skryptu z zadania 5.

```
M: 4; Kwant efektywny: 0.03702986
M: 9; Kwant efektywny: 0.02588504
M: 16; Kwant efektywny: 0.02054442
```

Jak widać, nadpróbkownanie sprawia, że efektywny kwant maleje, czyli rośnie efektywna liczba bitów. Wynika to z tego, że widmo szumu kwantyzacji jest równomiernie rozłożone na całym próbkowanym paśmie. Dzięki nadpróbkowaniu, można zmniejszyć pasmo spróbkowanego sygnału do pożądanej wartości, w ten sposób zmniejszając moc szumu. Między $M = 4$ a $M = 16$ mamy różnicę 4 razy jeżeli chodzi o zmniejszenie mocy w wyniku filtrowania. Jako że wartość sygnału jest pierwiastkiem z mocy, to amplituda szumu zmalała dwukrotnie, a więc i kwant efektywny stał się dwa razy mniejszy. W tym przykładzie nie jest to idealnie dwa razy mniej, co może wynikać, że dla takiej liczby próbek i takiego nadpróbkowania, rozkłady szumów nie są idealnie rozkładem ciągłym.

6 Zadanie 6

W tym zadaniu przerobiono poprzedni skrypt tak, aby mierzyć kwant efektywny dla różnych nadpróbkowań w sytuacji, gdy występuje zjawisko jitteru:

Listing 9: Skrypt MATLAB do zadania 6.

```
%% Zad6
clear variables
clc

%% parametry
M = [4 9 16];
A = [1; 1];
phi = [0; 0];
fs = 1;
jitt = 0.001;
q = 0.1;

for m=M
    fc = 0.1/m;
    f = fc * [0.9912331; 1.00323];
    N = 10000*m;

    x1 = gensinsum(A, phi, f, N, fs);
    x2 = gensinsumjitt(A, phi, f, N, jitt, fs);

    [b,a] = butter(5,(1/m)*fs/2);

    y1 = filter(b,a,x1);
    y2 = filter(b,a,x2);

```

```
    e = y1-y2;

    qeff = sqrt(12*cov(e));

    fprintf(['M: %.0f; ' ...
        'Kwant efektywny: %.8f\n'], ...
        m, qeff);
end
```

Program zwrócił:

Listing 10: Wynik działania skryptu z zadania 6.

```
M: 4; Kwant efektywny: 0.00019020
M: 9; Kwant efektywny: 0.00005792
M: 16; Kwant efektywny: 0.00002336
```

Tak małe wartości kwantu efektywnego wynikają z faktu, że bazując na rysunku z instrukcji, nie zastosowano funkcji `quant`. Mimo to widać, że zwiększenie nadpróbkowania pozwala ograniczyć błąd związany z jitterem.

Dla wartości jitteru $jitt = 0,1$ program zwrócił takie wyniki:

Listing 11: Wynik działania skryptu z zadania 6.

```
M: 4; Kwant efektywny: 0.01936855
M: 9; Kwant efektywny: 0.00577898
M: 16; Kwant efektywny: 0.00240727
```

Uwidacznia się zależność taka, że rząd pogorszenia jitteru i pogorszenia kwantu efektywnego są takie same.