SPRAWOZDANIE - LABORATORIUM NR 11
Szybka transformata sinusowa

Tomasz Kasprzak, 18 maja 2020

1 Wstęp teoretyczny

Metoda szybkiej transformacji sinusowej (DST - ang, Discrete Sine Transform) to bliźniacza metoda szybkiej dyskretnej transformacji cosinusowej. Obie te metody są rodzajami algorytmów szybkiej dyskretnej transformacji Fouriera (FFT - ang. Fast Fourier Transform), którą opisuje wzór (1). Są one wykorzystywane w przetwarzaniu sygnałów (odszumanie, analiza widma częstotliwości), kompresji danych, całkowaniu, interpolacji, aproksymacji, czy też mnożeniu wielomianów. W odróżnieniu od pozostałych algorytmów FFT (takich jak radix-2, PFT, czy split-radix), DST oraz DCT operują na sygnałach składających się z funkcji rzeczywistych - użycie ich w sytuacji kiedy dane są rzeczywiste znacznie zmniejsza kosz obliczeń.

$$f(x) = P(x) = \sum_{k=0}^{n-1} \langle f, E_k \rangle E_k \tag{1}$$

2 Zadanie

Na zajęciach laboratoryjnych zastosowaliśmy szybką transformację sinusową do odszumienia sygnału periodycznego. Sygnał zaszumiony wygenerowaliśmy zgodnie z poniższym algorytmem:

- sygnał okresowy nie zaszumiony ma postać:

$$y_0(i) = sin(\omega \cdot i) + sin(2\omega \cdot i) + sin(3\omega \cdot i) \tag{2}$$

  gdzie: i - numer próbki sygnału (numer elementu w wektorze).

$$\omega = 2\frac{2\pi}{n} \tag{3}$$

  gdzie: n - ilość próbek

- tworzymy zmienną losową imitującą szum:

$$a = 2sign \cdot X \tag{4}$$

  gdzie:

$$X = \frac{rand()}{RAND\_MAX + 1.} \tag{5}$$

  jest liczbą pseudolosową o rozkładzie równomiernym w przedziale $(0, 1)$. Znak zmiennej określamy następująco: losujemy drugą zmienną losową Y (podobnie jak X) a następnie dokonujemy wyboru:

$$y = \begin{cases} +1, & Y > \frac{1}{2} \\ -1, & Y \leqslant \frac{1}{2} \end{cases} \tag{6}$$

- sygnał zaszumiony konstruujemy następująco:

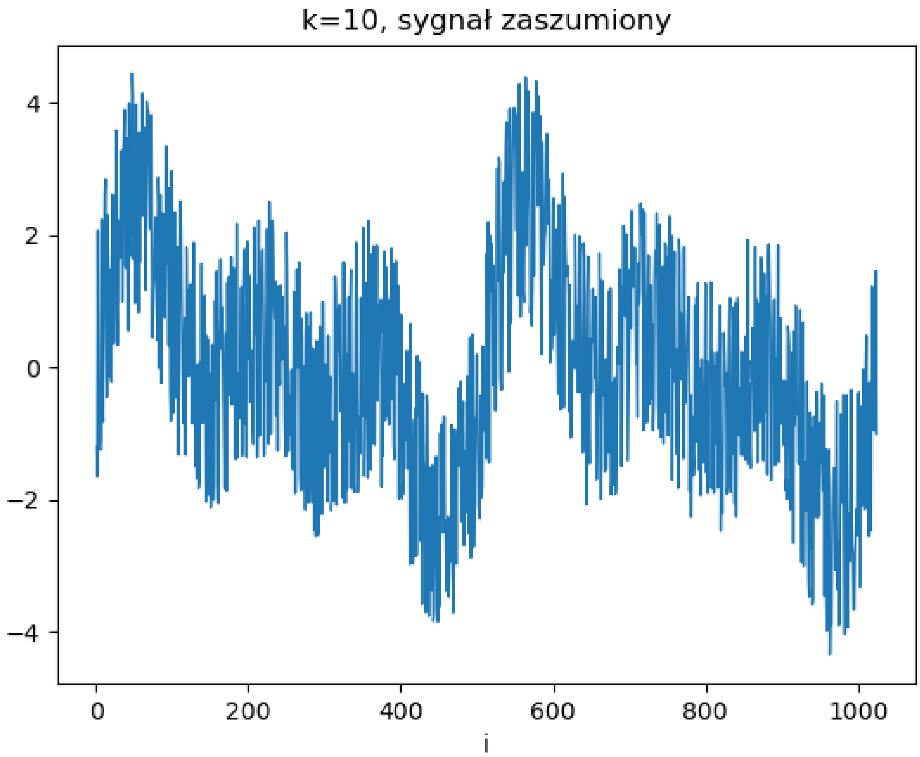
$$y(i) = y_0(i) + a \tag{7}$$

  wyznaczając wartość a dla każdego indeksu i z osobna.

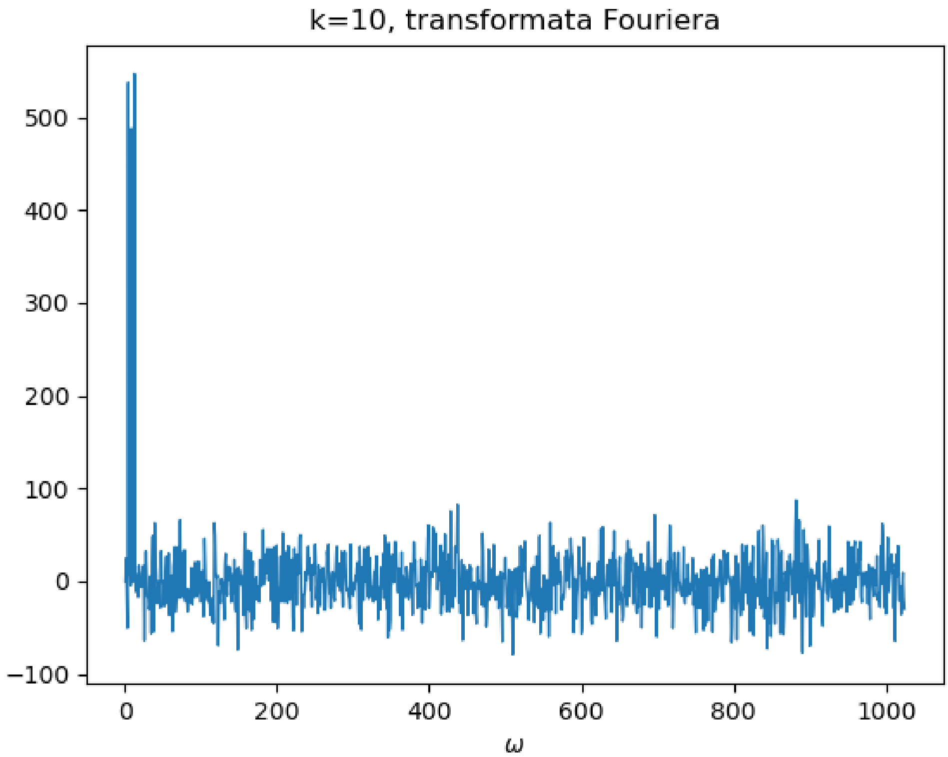
Wykonaliśmy następujące zadania:

- Zapisaliśmy zaszumiony sygnał do jednowymiarowej tablicy typu float. Jej długośc wynosi $2^k = n$
- Wykonaliśmy na tej tablicy transformację sinusową korzystając z funkcji $sinft$ z biblioteki $NumericalRecipes$.
- Dokonaliśmy w tablicy dyskryminacji na poziomie 25% wartości maksymalnej.
- Po dyskryminacji wyznaczyliśmy transformatę odwrotną (użyliśmy znowu procedury $sinft$ ale sygnał przemnożyliśmy przez $\frac{2}{n}$)

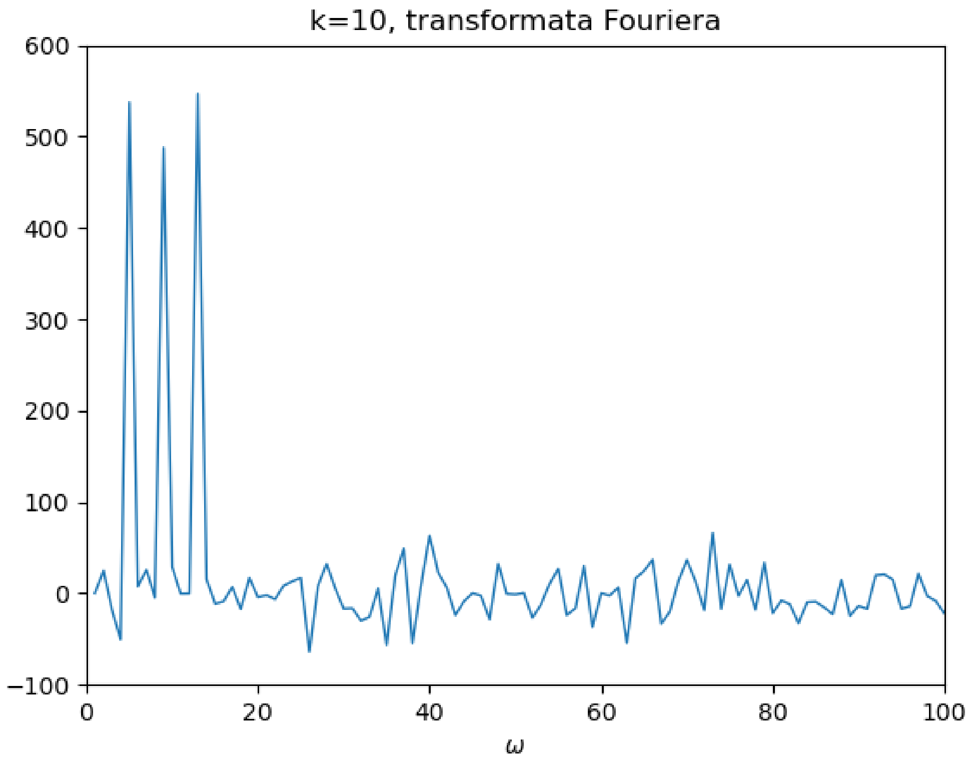
3 Wyniki



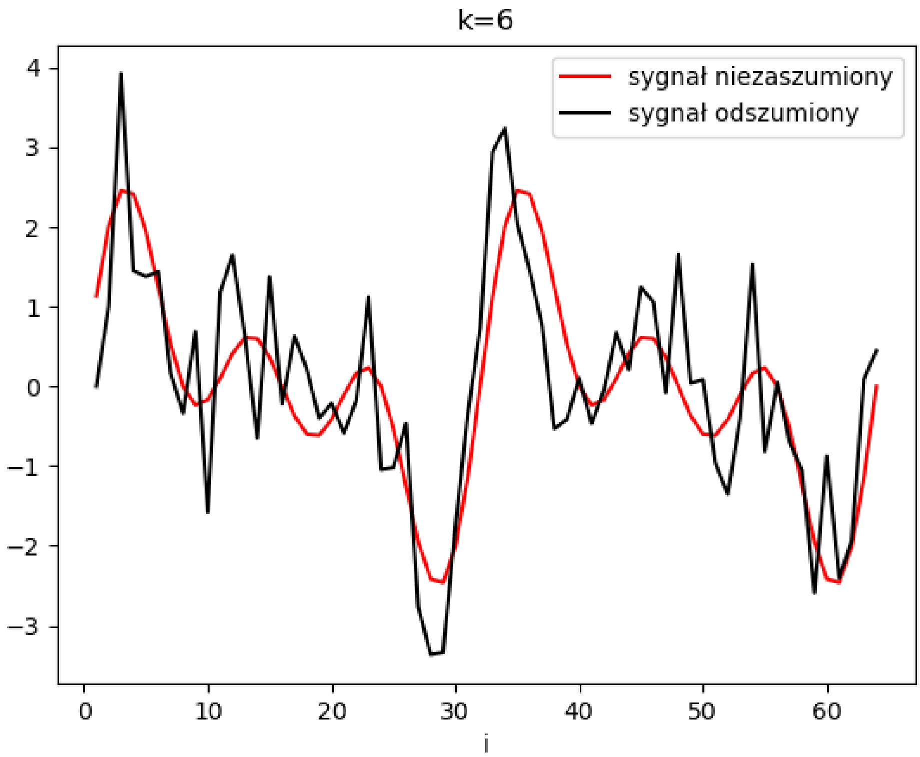
Rysunek 1: Wykres sygnału zaszumionego dla $k = 10$



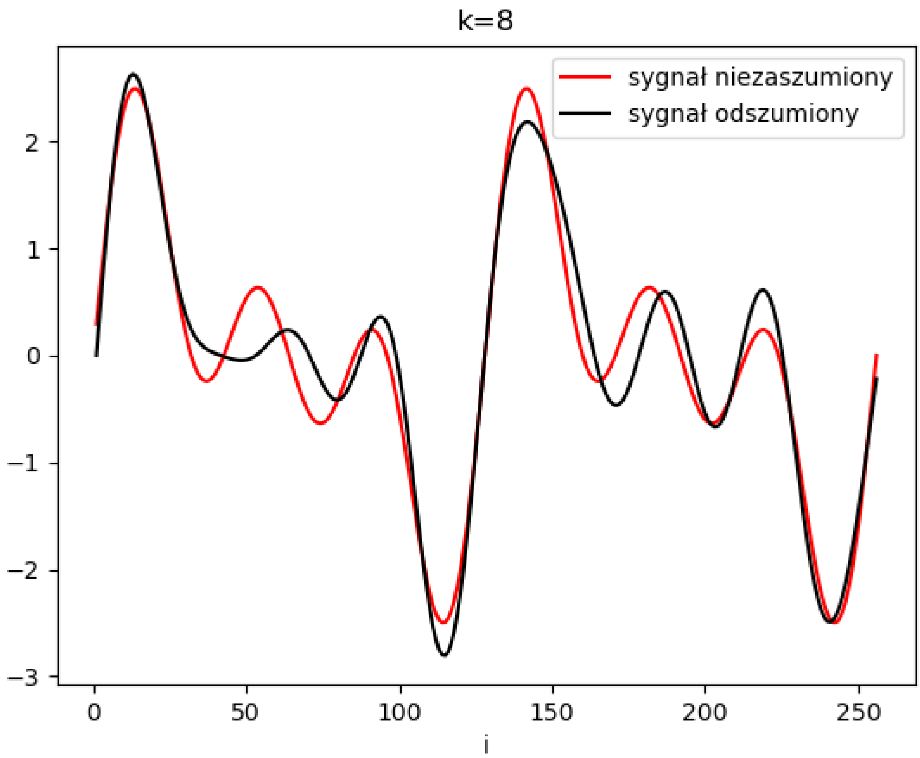
Rysunek 2: Wykres transformaty Fouriera dla $k = 10$



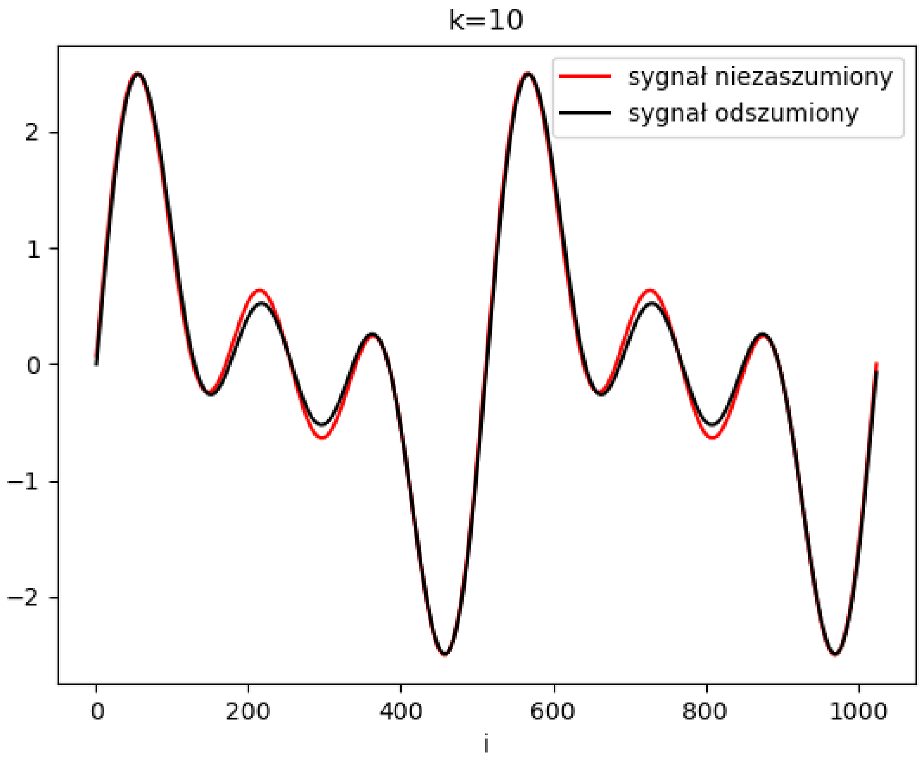
Rysunek 3: Wykres fragmentu transformaty Fouriera dla $k = 10$ dla największych pików wartości



Rysunek 4: Wykres sygnału odszumionego i niezaszumionego dla $k = 6$



Rysunek 5: Wykres sygnału odszumionego i niezaszumionego dla $k = 8$



Rysunek 6: Wykres sygnału odszumionego i niezaszumionego dla $k = 10$

4 Wnioski

Dla każdej wartości k widać prawidłowość - wartości odpowiadające udziałowi w wykresie funkcji y_0 części $sin(3\omega \cdot i)$, czyli miejsca o największej amplitudzie bardziej odpowiadają wykresowi bazowemu. Dzieje się tak prawdopodobnie dlatego, że szum jest niezależny od wartości funkcji (dla większych wartości funkcji, do tego rosnących szybko będzie on mniej znaczący. Na podstawie uzyskanych wykresów można określić wartość parametru k, dla którego odszumienie sygnału można uznać za satysfakcjonująco dokładne. Dla $k = 6$ odszumiony sygnał jest niedokładny - taka długość wektora skutkuje zbyt małą ilością wartości do dokładnego odszumenia sygnału. Poprawę widać już dla $k = 8$, a dla $k = 10$ dopasowanie wykresu odszumionego do bazowego wydaje się satysfakcjonujące dla większości zastosowań, w szczególności dla możliwości odczytu tego sygnału.