

# Sprawozdanie 11.

## Aproksymacja sygnału okresowego przy użyciu FFT

Mirosław Kołodziej

20.05.2021

### 1. Wstęp teoretyczny

#### 1.1 Szybka transformacja Fouriera (FFT)

**Szybka transformacja Fouriera** to algorytm wyznaczania dyskretnej transformaty Fouriera oraz transformaty do niej odwrotnej.

Najprostszym algorytmem FFT jest radix-2, czyli algorytm Cooley-Tukey. Został on opracowany w latach 60. dwudziestego wieku. Powstał on w celu szybkiej analizy danych sejsmologicznych.

Algorytm polega na znalezieniu współczynników transformaty Fouriera (DFT)  $c_k$  wykonując jak najmniejszą ilość obliczeń. Zakładamy, że całkowita liczba węzłów to potęga liczby 2:

$$x_j = \frac{2\pi}{N}j, \quad j = 0, 1, 2, \dots, N-1, \quad N = 2^r, \quad r \in \mathbb{N}$$

Zatem:

$$c_k = \sum_{j=0}^{N-1} f_j \exp\left(-I \frac{2\pi}{N}jk\right)$$

Grupujemy osobno składniki:

- parzyste ( $j = 2m$ ):

$$c_k = \sum_{m=0}^{\frac{N}{2}-1} f_{2m} \exp\left(-I \frac{2\pi}{N}(2m)k\right) + \sum_{m=0}^{\frac{N}{2}-1} f_{2m+1} \exp\left(-I \frac{2\pi}{N}(2m+1)k\right)$$

- nieparzyste ( $j = 2m+1$ ):

$$c_k = \sum_{m=0}^{\frac{N}{2}-1} f_{2m} \exp\left(-I \frac{2\pi}{N}mk\right) + \exp\left(-I \frac{2\pi}{N}k\right) \sum_{m=0}^{\frac{N}{2}-1} f_{2m+1} \exp\left(-I \frac{2\pi}{N}mk\right)$$

Można zauważyć, że  $p_k = \sum_{m=0}^{N/2-1} f_{2m} \exp\left(-I \frac{2\pi}{N/2}mk\right)$  i  $q_k = \sum_{m=0}^{N/2-1} f_{2m+1} \exp\left(-I \frac{2\pi}{N/2}mk\right)$  są okresowe, więc:

$$p_{k+N/2} = p_k, \quad q_{k+N/2} = q_k,$$

czyli mamy do wyznaczenia o połowę mniej współczynników. Natomiast czynnik fazowy  $\varphi_k = \exp\left(-I \frac{2\pi}{N} k\right)$  ma następującą własność:

$$\varphi_{k+\frac{N}{2}} = \exp\left(-I \frac{2\pi}{N} \left(k + \frac{N}{2}\right)\right) = -\exp\left(-I \frac{2\pi}{N} k\right) = -\varphi_k$$

## 2. Problem

Naszym zadaniem było zastosowanie FFT do odsumienia sygnału periodycznego. Sygnał zaszumiony generowaliśmy zgodnie z poniższym wzorem:

$$y(i) = \sin(\omega \cdot i) + \sin(2\omega \cdot i) + \sin(3\omega \cdot i) + \Delta,$$

gdzie:  $i = 0, 1, 2, \dots, N - 1$  to numer próbki sygnału,  $\omega = 2\frac{2\pi}{N}$ ,  $N = 2^k$  to ilość wygenerowanych próbek sygnału,  $\Delta = 2 \cdot \left(\frac{\text{rand}()}{\text{RAND\_MAX}+1.0}\right) - 1$  to liczba pseudolosowa o rozkładzie równomiernym w przedziale  $(-1, 1]$ .

Najpierw musieliśmy wygenerować zaszumiony sygnał (część rzeczywistą) i zapisać go do wektora typu double. Długość wektora wynosiła  $N = 2k$ , kolejno dla  $k = 8, 10, 12$ . Następnie wyznaczyliśmy transformatę sygnału korzystając z funkcji biblioteki GSL:

```
gsl_fft_complex_radix2_forward (double dane[], size_t sride=1, size_t N);
```

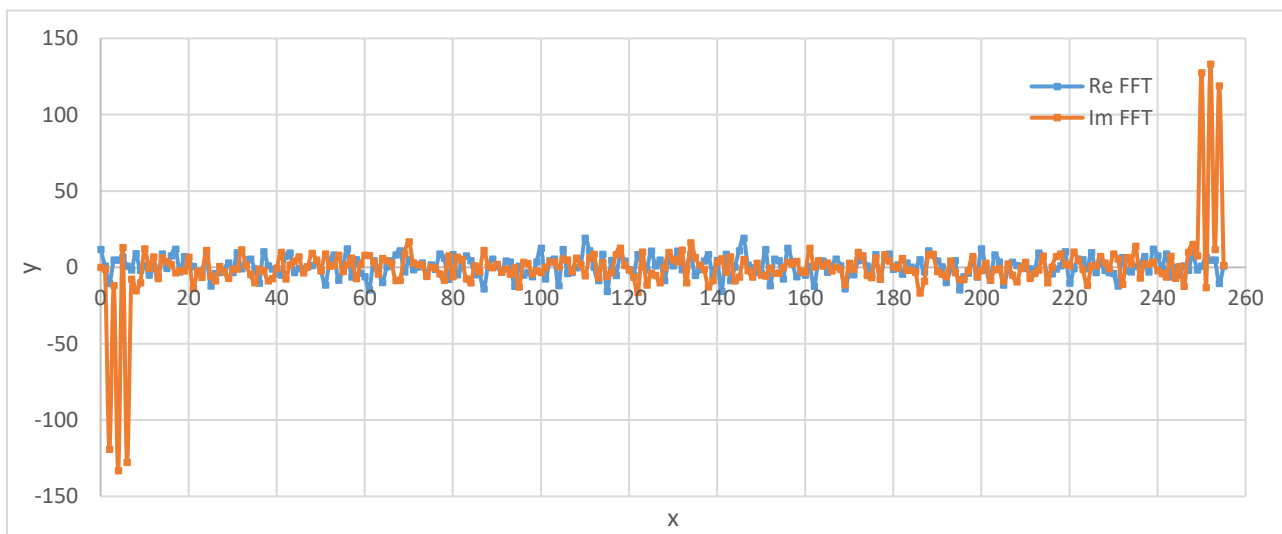
Później, dla każdego  $k$  przeprowadziliśmy dyskryminację sygnału na poziomie  $\max \frac{|c_k|}{2}$ , czyli wyzerowaliśmy te współczynniki transformaty (części rzeczywiste i urojone), które nie przekraczają tego progu. Po dyskryminacji wyznaczyliśmy transformatę odwrotną. Skorzystaliśmy przy tym z funkcji:

```
gsl_fft_complex_radix2_backward (double dane[], size_t sride, size_t N);
```

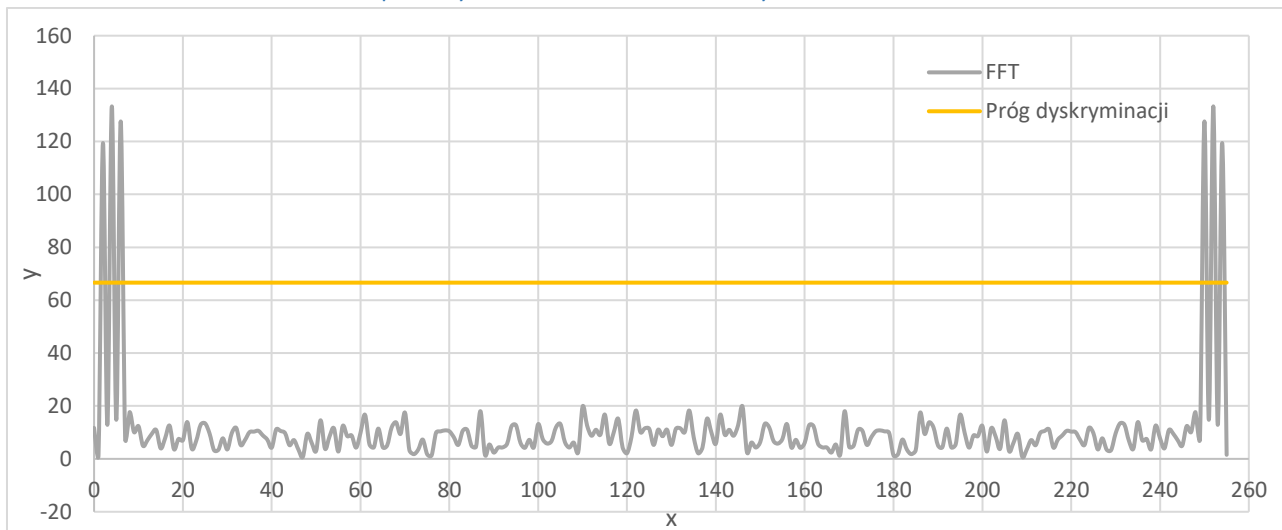
Otrzymany sygnał unormowaliśmy dzieląc go przez  $N$ . Na koniec, dla każdego  $k$  wykonaliśmy po dwa rysunki: sygnału zaburzonego i odsumionego oraz sygnału niezaburzonego i odsumionego. Dodatkowo, dla  $k = 8$  sporządziliśmy dwa rysunki: pierwszy pokazujący część rzeczywistą i urojoną transformaty oraz drugi pokazujący wartości modułów współczynników transformaty.

### 3. Wyniki

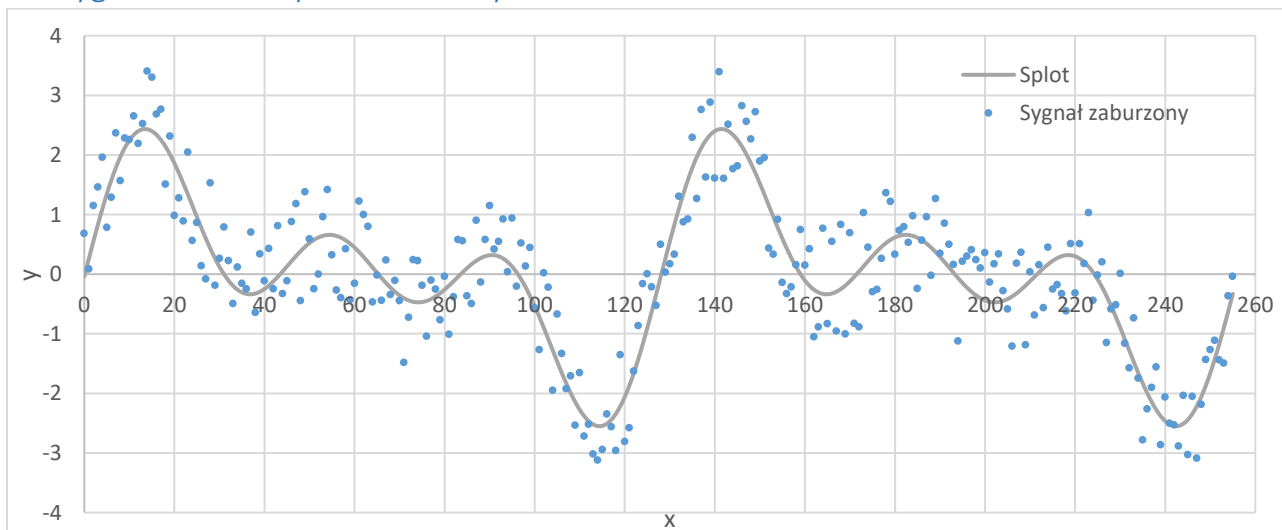
#### 3.1 Część rzeczywista i urojona transformaty dla $k = 8$



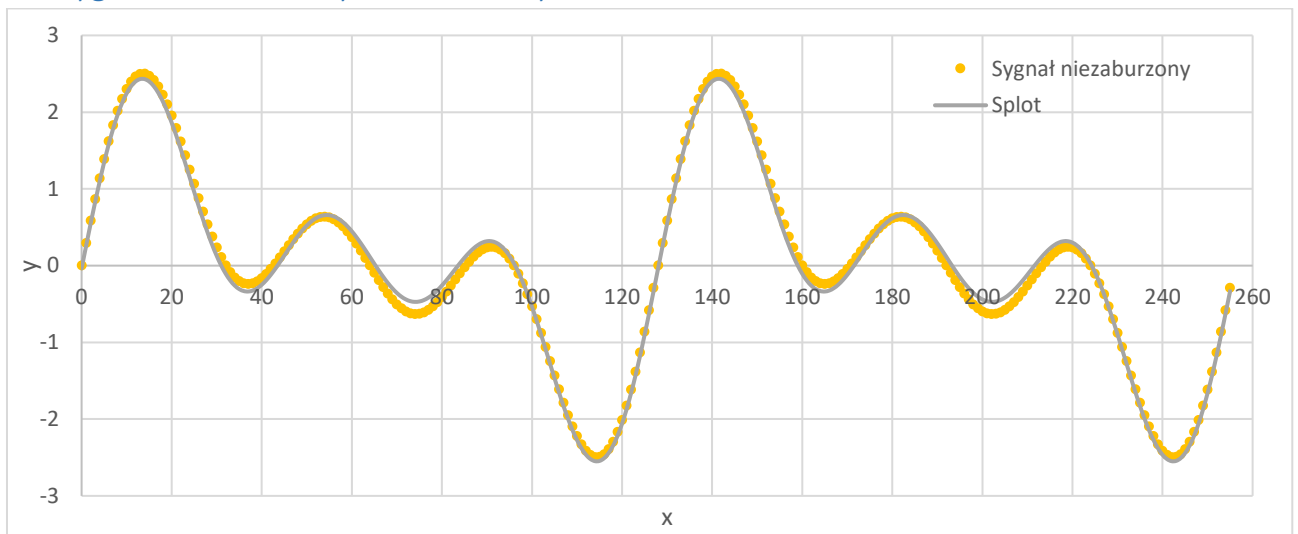
#### 3.2 Wartości modułów współczynników transformaty dla $k = 8$



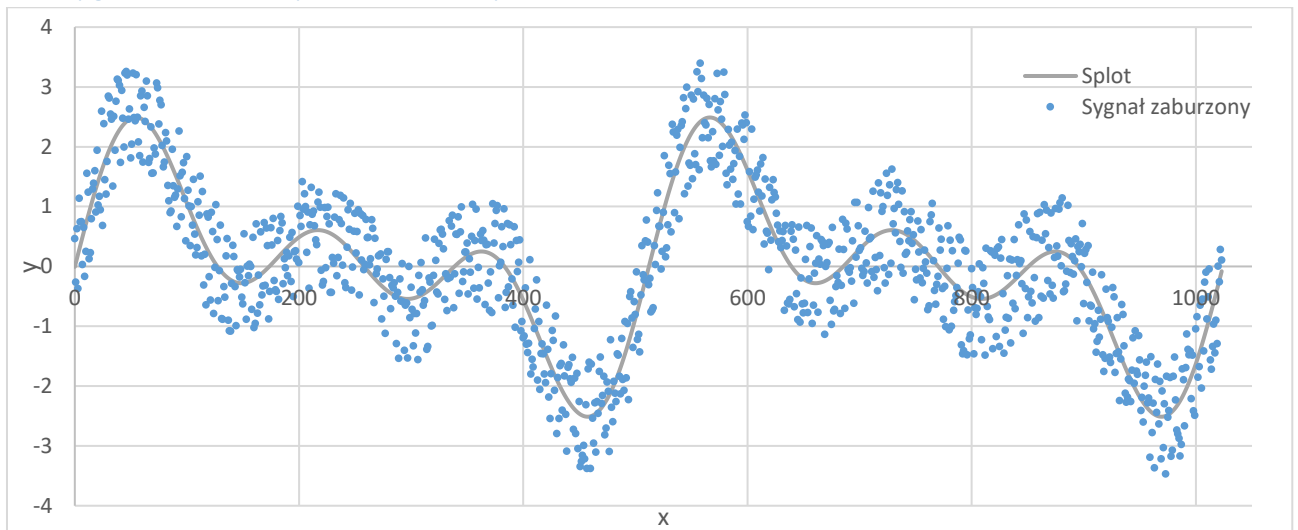
#### 3.3 Sygnał zaburzony i odszumiony dla $k = 8$



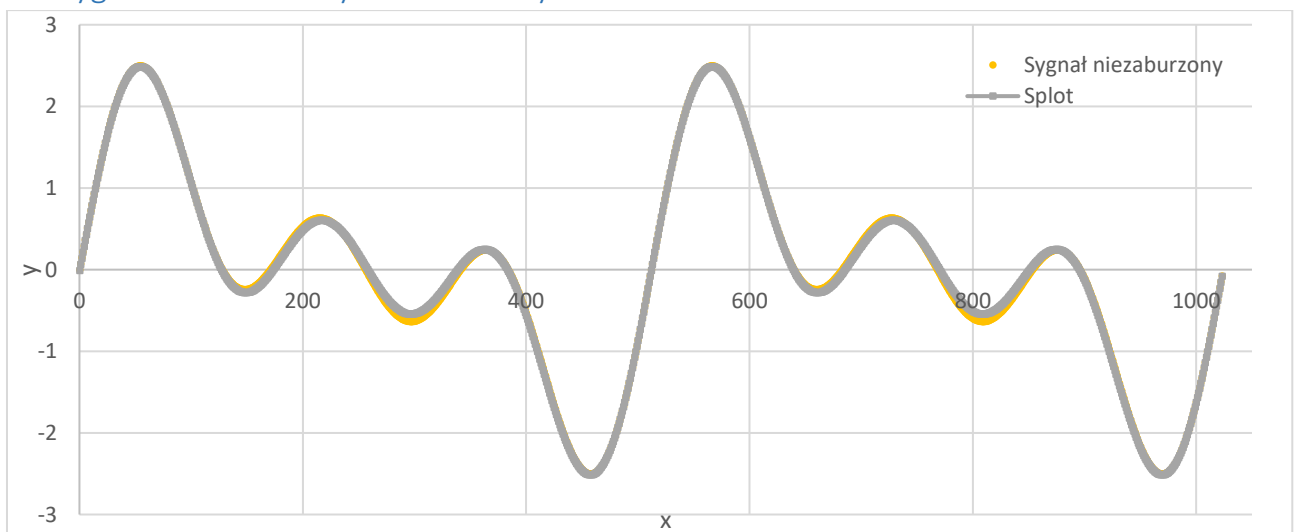
### 3.4 Sygnał niezaburzony i odszumiony dla $k = 8$



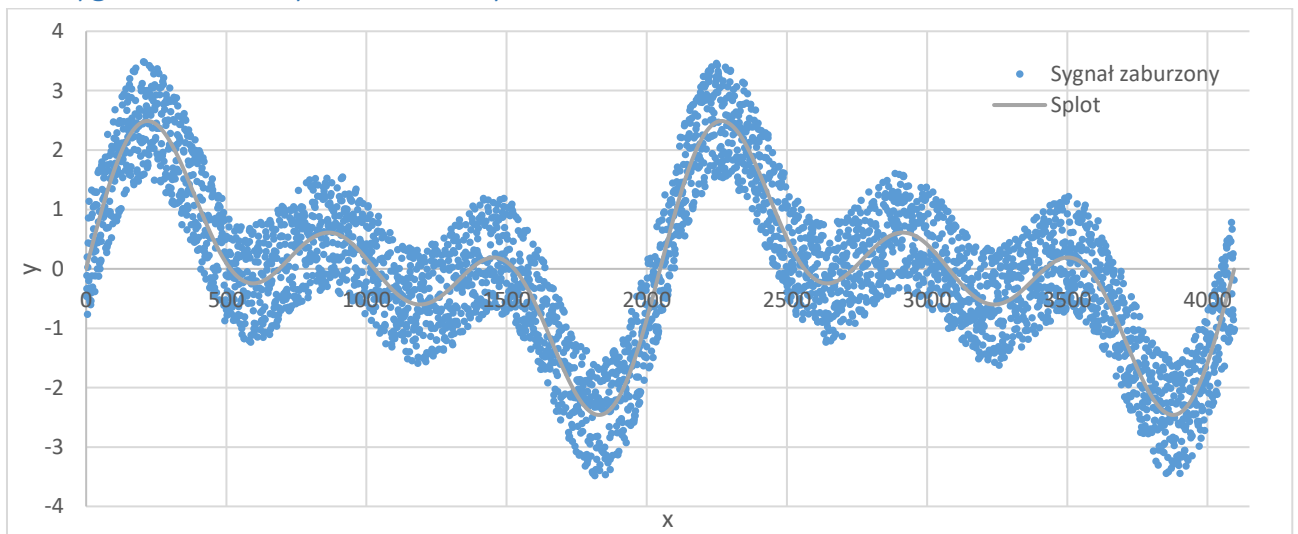
### 3.5 Sygnał zaburzony i odszumiony dla $k = 10$



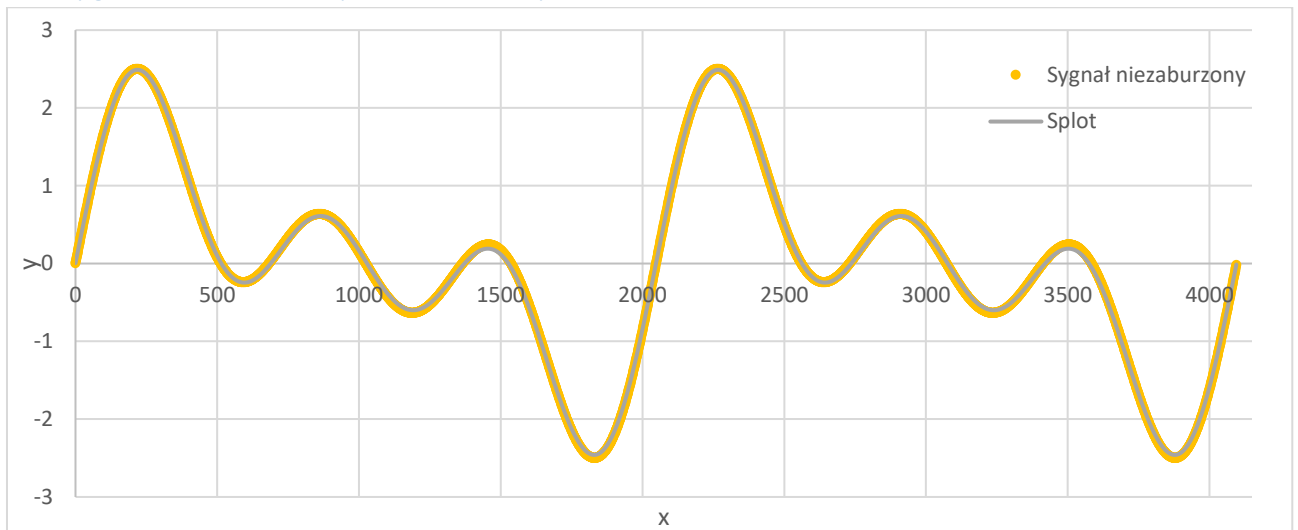
### 3.6 Sygnał niezaburzony i odszumiony dla $k = 10$



### 3.7 Sygnał zaburzony i odszumiony dla $k = 12$



### 3.8 Sygnał niezaburzony i odszumiony dla $k = 12$



## 4. Wnioski

Odszumiliśmy sygnał za pomocą FFT. Możemy zauważyć, że dla większej ilości wygenerowanych próbek sygnału otrzymane wyniki pokrywają się bardziej z sygnałem niezaburzonym. Częstotliwość próbkowania ma zatem wpływ na dokładność otrzymywanych wyników.