[redacted] – MN ćw.09

Data wykonania: 12-12-2012

1 Wstęp

Celem zajęć było zastosowanie szybkiej transformaty kosinusowej do odszumienia danego sygnału okresowego. Sygnał niezaszumiony miał postać

$$y_0(i) = \cos(\omega i) + \cos(2\omega i) + \cos(3\omega i) \tag{1}$$

gdzie i to numer próbki sygnału, a $\omega = \frac{4\pi}{n}$, $n = 2^k$ to ilość próbek. Sygnał ten został zaszumiony poprzez zmienną losową wygenerowaną następująco:

$$a = \pm \frac{2rand()}{RAND\_MAX + 1.0} \tag{2}$$

Należało obliczyć transformatę Fouriera przy użyciu procedury `cosft2` z biblioteki Numerical Recipes, następnie dokonać w niej dyskryminacji na poziomie 25% wartości maksymalnej i na końcu obliczyć transformatę odwrotną. Końcowy sygnał należało dodatkowo znormalizować, czyli pomnożyć przez $\frac{2}{n}$. Całą procedurę należało powtórzyć dla $k = 6, 8, 10$ i dla każdej z tych wartości narysować sygnał niezaszumiony oraz sygnał po odszumieniu. Dodatkowo dla $k = 10$ należało wykonać dodatkowe wykresy: sygnału zaszumionego, transformaty oraz transformaty z widocznymi pikami.

Procedura `cosft2` implementuje algorytm radix-2 (algorytm Cooley'a-Tukey'a). Zadaniem jest obliczenie współczynników transformaty Fouriera c_k w zależności między szeregiem Fouriera rzeczywistym a zespolonym. Zakładając, że całkowita liczba węzłów N jest potęgą liczby 2 możemy zapisać:

$$\begin{aligned} x_j &= \frac{2\pi j}{N} \\ j &= 0, 1, 2, \ldots, N-1 \\ N &= 2^r, r \in \mathbb{N} \end{aligned} \tag{3}$$

Wtedy:

$$c_k = \langle E_k, f \rangle = \sum_{j=0}^{N-1} f(x_j) E_k(x_j) = \sum_{j=0}^{N-1} f(x_j) e^{-i x_j k} = \sum_{j=0}^{N-1} f_j e^{-\frac{i2\pi jk}{N}} \tag{4}$$

Następnie osobno grupujemy składniki parzyste ($j = 2m$) oraz nieparzyste ($j = 2m + 1$):

$$\begin{aligned} c_k &= \sum_{m=0}^{\frac{N}{2}-1} f_{2m} e^{-i\frac{2\pi}{N}2mk} + \sum_{m=0}^{\frac{N}{2}-1} f_{2m+1} e^{-i\frac{2\pi}{N}(2m+1)k} \\ c_k &= \sum_{m=0}^{\frac{N}{2}-1} f_{2m} e^{-i\frac{2\pi}{N/2}mk} + e^{-i\frac{2\pi}{N}k} \sum_{m=0}^{\frac{N}{2}-1} f_{2m+1} e^{-i\frac{2\pi}{N/2}mk} \\ c_k &= p_k + \phi_k q_k \\ p_k &= \sum_{m=0}^{\frac{N}{2}-1} f_{2m} e^{-i\frac{2\pi}{N/2}mk} \\ q_k &= \sum_{m=0}^{\frac{N}{2}-1} f_{2m+1} e^{-i\frac{2\pi}{N/2}mk} \\ \phi_k &= e^{-i\frac{2\pi}{N}k} \end{aligned} \tag{5}$$

Korzystamy teraz z okresowości funkcji p_k oraz q_k, oraz wyznaczamy $\phi_{j+N/2}$:

$$\begin{aligned} p_{k+N/2} &= p_k \\ q_{k+N/2} &= q_k \end{aligned} \tag{6}$$

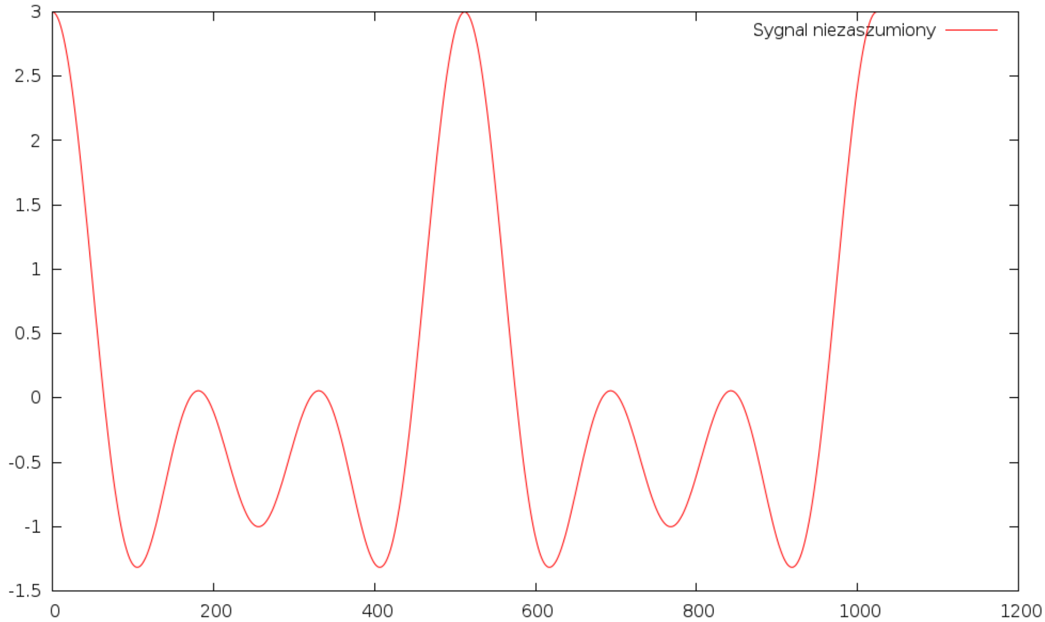
$$\phi_{k+N/2} = e^{-i\frac{2\pi}{N}(k+\frac{N}{2})} = e^{-i\frac{2\pi}{N}k} e^{-i\frac{2\pi}{N}\frac{N}{2}} = e^{-i\frac{2\pi}{N}k}(\cos\pi + i\sin\pi) = -e^{-i\frac{2\pi}{N}k} = -\phi_k \tag{7}$$

Całą procedurę powtarzamy osobno dla p_k i q_k, wtedy liczba elementów w każdej z dwóch powstałych sum jest dwukrotnie mniejsza niż w elemencie macierzystym. Proces ten powtarzamy rekurencyjnie i kończymy, gdy liczba elementów jest równa 1.

2 Wyniki

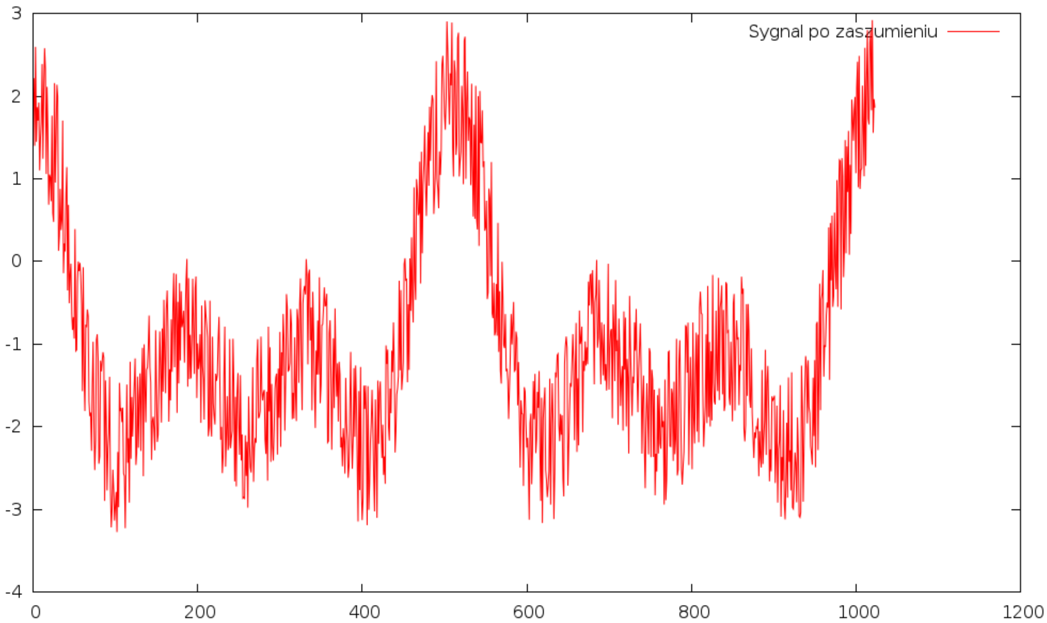
Dla $k = 10$ wygenerowałem sygnał wyjściowy, składający się z $n = 2^{10} = 1024$ punktów. Ma on następującą postać:

Rysunek 1: Wykres sygnału wyjściowego, k=10



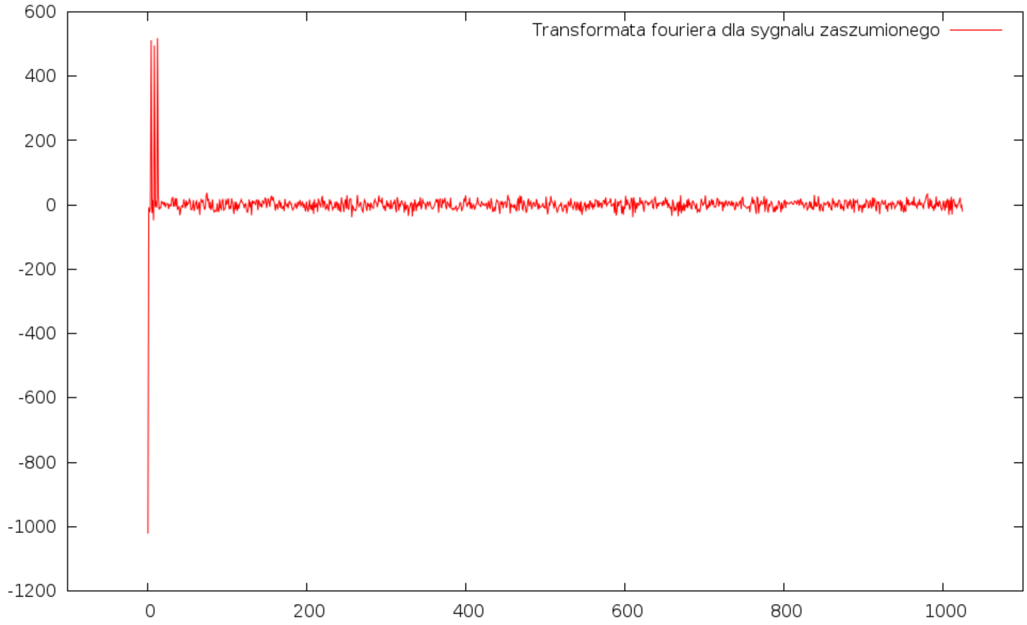
Następnie zaszumiłem ten sygnał losową zmienną a opisaną we wstępie:

Rysunek 2: Wykres sygnału wyjściowego po zaszumieniu, k=10



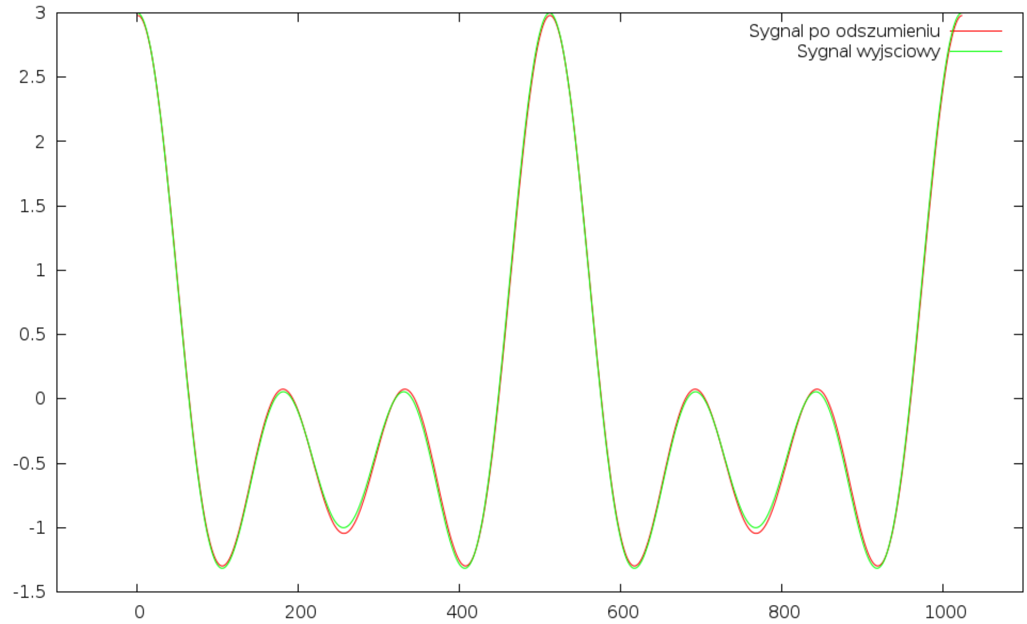
Po obliczeniu transformaty fouriera procedurą `cosft2` zrobiłem jej wykres, przesuwając go tak, aby widoczne były piki:

Rysunek 3: Transformata fouriera dla sygnału zaszumionego, k=10



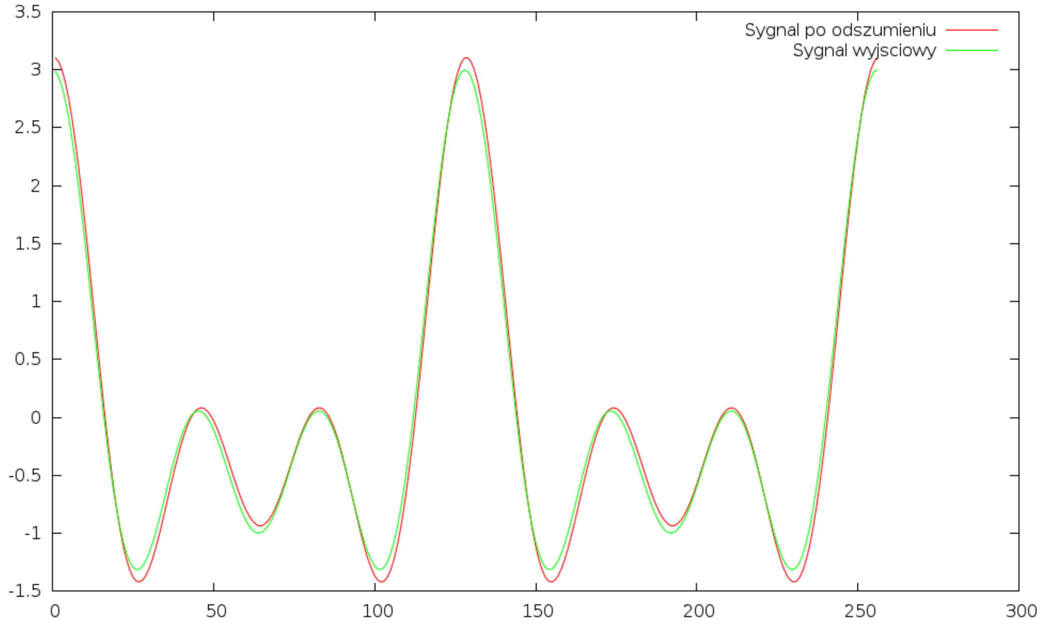
Widać wyraźnie, że blisko 0 występują znaczące odchylenia od pozostałej części wykresu. Dokonałem więc dyskryminacji na poziomie 25% wartości maksymalnej i wywołałem procedurę `cosft2` w celu przeprowadzenia transformaty odwrotnej. Na jednym wykresie umieściłem sygnał wyjściowy oraz sygnał po odszumieniu:

Rysunek 4: Sygnał wejściowy i po odszumieniu, k=10

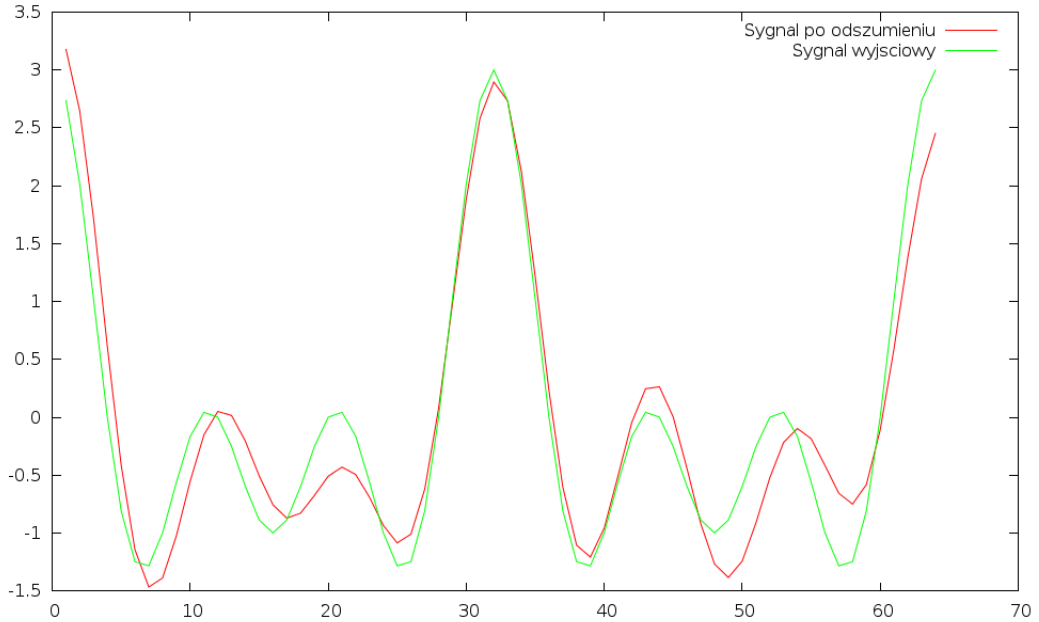


Całą procedurę powtórzyłem dla $k = 6$ oraz $k = 8$:

Rysunek 5: Sygnał wejściowy i po odszumieniu, k=8



Rysunek 6: Sygnał wejściowy i po odszumieniu, k=6



3 Wnioski

Dla $k = 10$ (1024 próbki) sygnał po odszumieniu nie różni się znacząco od sygnału wyjściowego. Jednak im mniej próbek sygnału, tym odszumianie za pomocą FFT jest mniej precyzyjne. Dla $k = 8$ (256 próbek) sygnał po odszumieniu jest jeszcze akceptowalny, jednak dla $k = 6$ (64 próbki) sygnał po odszumieniu bardzo znacząco odbiega od wyjściowego. Największe różnice występują w lokalnych ekstremach sygnału.