Sprawozdanie - Laboratorium nr 11

Odszumianie sygnału przy użyciu FFT - splot funkcji

Hubert Wdowiak, 21.05.2020

1. Wstęp Teoretyczny

Podczas trwania zajęć poznaliśmy sposób na odszumianie sygnału, którego wartości zdały się mieć dodatkowy losowy czynnik, który sprawia, że wykres sygnału jest "zanieczyszczony". Aby tego dokonać, posłużyliśmy się szybką transformacją Fouriera, oznaczoną skrótem FFT.

Korzysta ona ze splotu funkcji *f* (sygnał), oraz *g* (funkcji wagowej), który to można potraktować jako uśrednienie funkcji *f*. Taki splot określony jest wzorem:

$$(f * g)(t) = \int_{-\infty}^{\infty} f(\tau)g(t-\tau)d\tau \quad (1)$$

2. Opis problemu

W trakcie laboratorium naszym zadaniem było oczyszczenie z szumu, sygnału danego wzorem:

$$f(t) = f_0(t) + \Delta \quad (2)$$

$$f_0(t) = sin(1 \cdot \omega t) + sin(2 \cdot \omega t) + sin(3 \cdot \omega t) \quad , \quad (3)$$

przy czym $\omega = 2\pi / T$, Δ oznacza losową wartość z przedziału [-0.5, 0.5] ($T = 1.0$, oznacza okres). Zadanie zostało wykonane trzykrotnie - dla każdej z liczb węzłów N = : 2^8, 2^{10}, 2^{12}. Maksymalny okres czasu trwania sygnału - t_{max}, został ustalony na 3T, a krok czasowy - *dt*, wynosił t_{max}/N.

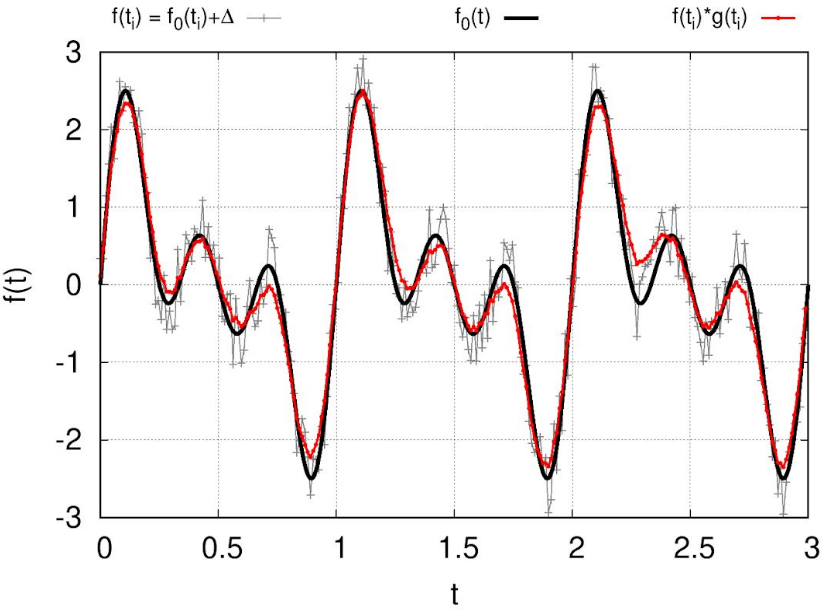
Funkcja wagowa została określona wzorem:

$$g(t) = \frac{1}{\sigma\sqrt{2\pi}} exp\left(-\frac{t^2}{2\sigma^2}\right) \quad , \quad (4)$$

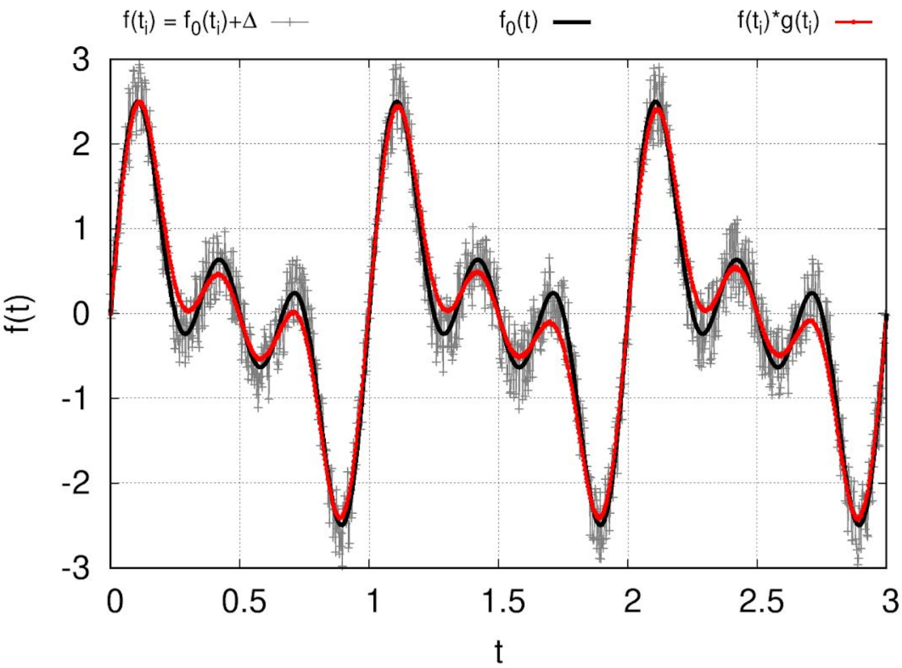
gdzie $\sigma = T/20$.

Do wykonania ćwiczenia, wykorzystaliśmy funkcję *four1(float [], int, isign)*, znajdującą się w bibliotece *Numerical Recipes*, w której pierwszy argument, oznacza wektor o długości 2 * [drugi argument] , którego komórki o nieparzystych indeksach, przechowują rzeczywiste wartości sygnału w kolejnych chwilach czasowych, a komórki o parzystych numerach indeksów, przechowywały (w naszym przypadku) zawsze zerową część urojoną wartości sygnału. Trzeci argument, decyduje czy algorytm FFT, powinien zostać przeprowadzony w sposób normalny (wartość 1), czy odwrócony (wartość -1). Cały program potrzebny do wykonania zadania, został zapisany w języku C++.

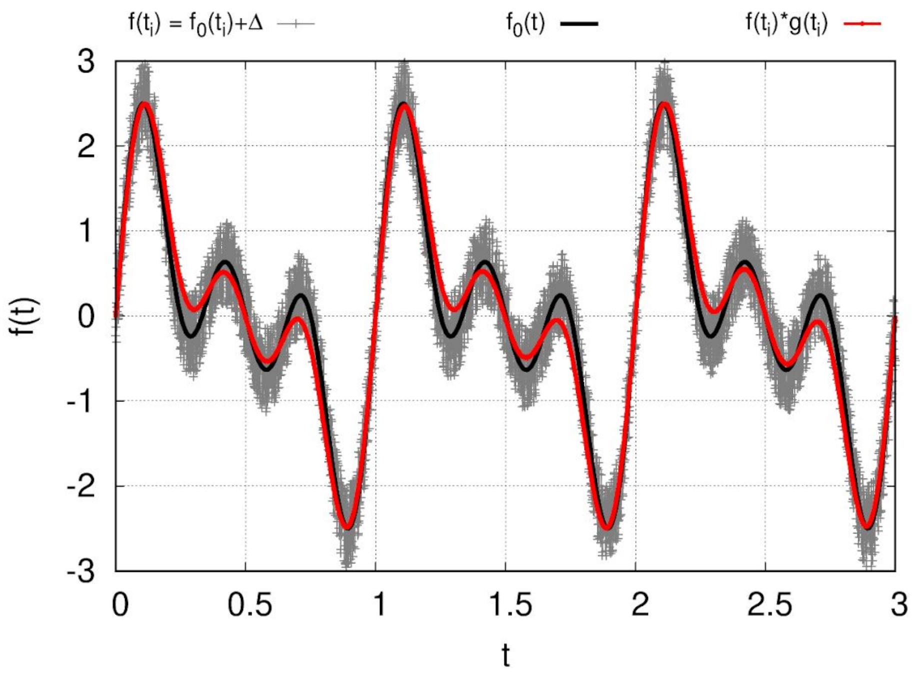
3. Wyniki



Wykres 1. Znormalizowany, odszumiony sygnał, oraz sygnał niezaburzony, dla parametru k=8



Wykres 2. Znormalizowany, odszumiony sygnał, oraz sygnał niezaburzony, dla parametru k=10



Wykres 3. Znormalizowany, odszumiony sygnał, oraz sygnał niezaburzony, dla parametru k=12

Powyższe wykresy wskazują na powiązanie pomiędzy gładkością odszumionego sygnału, a liczbą punktów wejściowych. Na pierwszym wykresie, gdzie owa liczba wynosi 2^8, krawędzie odszumionej funkcji wydają się nie być gładkie, a wykresy różnią się znacząco w okolicach ekstremów lokalnych.

Na *wykresie 2,* po zwiększeniu ilości punktów początkowych do 2^{10}, zauważyć można wyraźne wygładzenie otrzymanych krzywych, a także delikatnie lepsze dopasowanie funkcji niezaburzonej i wygładzonej. Kontynuacja zwiększania punktów wejściowych, do wartości 2^{12}, przyniosła pozytywne skutki, i sygnał odszumiony okazał się, w jeszcze lepszym stopniu przypominać oryginalny, niezaburzony sygnał. Otrzymana krzywa została dodatkowo wygładzona, jednak mimo wszystko widoczne są wyraźne różnice pomiędzy obiema krzywymi, nawet dla parametru k=12.

4. Wnioski

Algorytm FFT jest w stanie odnaleźć przybliżenie funkcji, która została zanieczyszczona pewnym szumem. Wedle naszych doświadczeń, nie jesteśmy w stanie w ten sposób otrzymać funkcji w 100% pokrywającej się z funkcją poszukiwaną. Jakość przybliżenia jest jednoznacznie powiązana z ilością próbek na wejściu algorytmu. Im większa jest ta wartość, tym dokładniejsze i gładsze wykresy odszumionej funkcji jesteśmy w stanie uzyskać. W każdym z analizowanych przez nas przykładów, największe odchyły były uwidaczniane w okolicach ekstremów lokalnych funkcji. Pomimo to ich ogólny kształt był podobny, podczas całego ich przebiegu.