Sprawozdanie - laboratorium nr 11

Odszumianie sygnału przy użyciu FFT – splot funkcji.

Damian Płóciennik

15 maja 2019

1 Wstęp teoretyczny

1.1 Szybka transformacja Fouriera

Szybką transformacją Fouriera (FFT) nazywamy algorytm wyznaczania dyskretnej transformaty Fouriera oraz transformaty do niej odwrotnej. Dzięki niej praktycznie możliwe stało się cyfrowe przetwarzanie sygnałów (DSP), a także zastosowanie dyskretnych transformat kosinusowych (DCT) do kompresji danych audio-wideo (JPEG, MP3, XviD itd.).

1.2 Algorytm radix-2

Najprostszy algorytm FFT to radix-2 (Cooley-Tukey) opracowany w latach 60 XX wieku w celu szybkiej analizy danych sejsmologicznych. W algorytmie tym zakładamy, że całkowita liczba węzłów jest potęga 2, to jest:

$$N = 2^r, \quad r \in \mathbf{R}. \tag{1}$$

Węzły oznaczamy, więc:

$$x_j = \frac{2\pi}{N} j, \quad j = 0, 1, \dots, N-1, \tag{2}$$

a współczynniki wyznaczamy następująco:

$$c_k = \langle E_k, f \rangle = \sum_{j=0}^{N-1} f(x_j) E_k(x_j) = \sum_{j=0}^{N-1} f_j \exp\Big(- I \frac{2\pi}{N} jk \Big). \tag{3}$$

Grupując osobno składniki parzyste ($j = 2m$) i nieparzyste ($j = 2m + 1$), otrzymujemy:

$$c_k = \sum_{m=0}^{\frac{N}{2}-1} f_{2m} \exp\Big(- I \frac{2\pi}{N} (2m) k \Big) + \sum_{m=0}^{\frac{N}{2}-1} f_{2m+1} \exp\Big(- I \frac{2\pi}{N} (2m+1) k \Big), \tag{4}$$

z czego możemy otrzymać:

$$c_k = \sum_{m=0}^{\frac{N}{2}-1} f_{2m} \exp\Big(- I \frac{2\pi}{N/2} mk \Big) + \exp\Big(- I \frac{2\pi}{N} k \Big) \sum_{m=0}^{\frac{N}{2}-1} f_{2m+1} \exp\Big(- I \frac{2\pi}{N/2} mk \Big). \tag{5}$$

Przyjmując kolejno:

$$p_k = \sum_{m=0}^{\frac{N}{2}-1} f_{2m} \exp\Big(- I \frac{2\pi}{N/2} mk \Big), \tag{6}$$

$$q_k = \sum_{m=0}^{\frac{N}{2}-1} f_{2m+1} \exp\Big(- I \frac{2\pi}{N/2} mk \Big), \tag{7}$$

$$\varphi = \exp\Big(- I \frac{2\pi}{N} k \Big) \tag{8}$$

możemy zapisać:

$$c_k = p_k + \varphi_k q_k. \tag{9}$$

Korzystając z okresowości, możemy określić:

$$p_{k+N/2} = p_k \qquad q_{k+N/2} = q_k \qquad \varphi_{k+N/2} = -\varphi_k \tag{10}$$

gdzie współczynniki p_k oraz q_k można wyliczyć dzięki DFT nakładem $\mathrm{O}(N^2/4)$ oraz można oszczędzić dodatkowy czas wyznaczając tylko współczynniki dla $k < \frac{N}{2}$, ponieważ:

$$c_k = \begin{cases} p_k + \varphi q_k & k < \frac{N}{2} \\ p_{k-\frac{N}{2}} - \varphi_k q_{k-\frac{N}{2}}, & k \geqslant \frac{N}{2} \end{cases} \tag{11}$$

Kolejnym krokiem FFT jest podział sum w p_k oraz q_k na sumy zawierające tylko elementy parzyste i nieparzyste, po którym liczba elementów w dwóch z pozostałych sum jest dwukrotnie mniejsza niż w oryginalnym elemencie. Tak powtarzany rekurencyjnie podział kończymy, gdy liczba elementów osiągnie 1.

2 Zadanie do wykonania

2.1 Opis problemu

Splot dwóch funkcji definiujemy jako:

$$(f * g)(t) = \int_{-\infty}^{\infty} f(\tau)g(t-\tau)\mathrm{d}\tau. \tag{12}$$

Jeśli funkcję $f(t)$ potraktujemy jako sygnał a funkcję $g(t)$ jako wagę, to splot tych dwóch funkcji można potraktować jako uśrednienie funkcji f pewną ustaloną funkcją wagową g. Wykorzystano ten fakt do wygładzenia zaszumionego sygnału. Aby przeprowadzić efektywnie obliczenia, do obliczenia splotu wykorzystano FFT:

$$\mathrm{FFT}\{f(t) * g(t)\} = \mathrm{FFT}\{f\} \cdot \mathrm{FFT}\{g\} = f(k) \cdot g(k), \tag{13}$$

$$f * g = \mathrm{FFT}^{-1}\{f(k) \cdot g(k)\}. \tag{14}$$

Jako sygnał przyjęto:

$$f(t) = f_0(t) + \Delta, \tag{15}$$

gdzie:

$$f_0(t) = \sin(\omega t) + \sin(2 \cdot \omega t) + \sin(3 \cdot \omega t). \tag{16}$$

Jako funkcję wagową przyjęto funkcję gaussowską:

$$g(t) = \frac{1}{\sigma\sqrt{2\pi}} \exp\left(-\frac{t^2}{2\sigma^2}\right). \tag{17}$$

Ze względu na operowanie na chwilach czasowych, które są dodatnie, to jest $t \in [0, t_{max}]$ funkcja $g(t)$ przedstawia tylko połowę oryginalnej funkcji gaussowskiej (środek wypada w $t = 0$). Dla dodatnich wartości t licząc $g_1(k)$ zastosujemy wzór:

$$g_1(k) = \mathrm{FFT}\{g(t > 0)\} = \frac{1}{N}\sum_{i=1}^{N} g(t_i) \exp\left(-\frac{2 \cdot \pi \cdot \cdot k \cdot i}{N}\right). \tag{18}$$

By policzyć transformatę dla drugiej połówki zmieniamy znak przy t, to jest $g(t) = g(-t)$ ze względu na symetrię:

$$g_2(k) = \mathrm{FFT}\{g(t < 0\} = \frac{1}{N}\sum_{i=1}^{N} g(t_i) \exp\left(\frac{2 \cdot \pi \cdot \cdot k \cdot i}{N}\right) = \mathrm{FFT}^{-1}\{g(t > 0)\}. \tag{19}$$

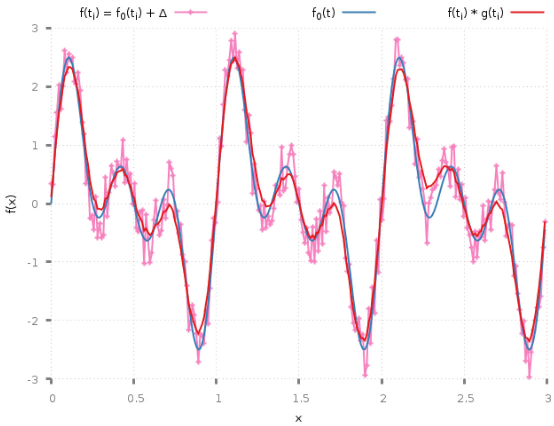
Do liczenia splotu, zamiast $g(k) = \mathrm{FFT}\{g(t)\}$, można użyć więc sumy dwóch transformat:

$$g(k) = \mathrm{FFT}\{g(t)\} + \mathrm{FFT}^{-1}\{g(t)\}. \tag{20}$$

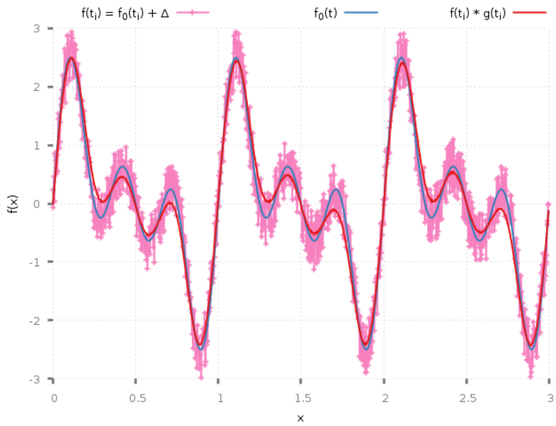
W celu wykonania zadania przyjęto parametry: $N^k = 2^k$, $k = 8, 10, 12$ - liczba węzłów, $T = 1.0$, $t_{max} = 3T$ - maksymalny okres czasu trwania sygnału, $dt = t_{max}/N_k$ - krok czasowy, $\sigma = T/20$. Δ jest liczbą pseudolosową, $\Delta \in [-0.5, 0.5]$

2.2 Wyniki

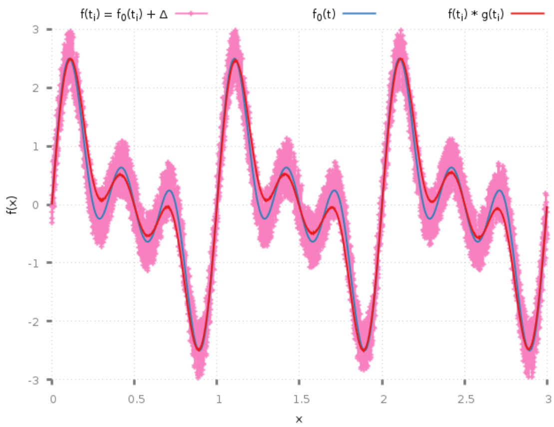
Do wykonania zadani wykorzystano program napisany w języku C z wykorzystaniem biblioteki Numerical Recipes. Na wykresach dokonano porównania oryginalnego sygnału niezaburzonego, sygnału zaburzonego oraz sygnału wygładzonego (odszumionego) kolejno dla $k = 8, 10, 12$.



(a) $k = 8 \Longrightarrow N_k = 2^8$ próbek wejściowych



(b) $k = 10 \Longrightarrow N_k = 2^{10}$ próbek wejściowych



(c) $k = 12 \Longrightarrow N_k = 2^{12}$ próbek wejściowych

Rysunek 1: Wynik odszumiania sygnału przy użyciu FFT; $f_0(t)$ - oryginalny sygnał niezaburzony, $f(t) = f_0(t) + \Delta$ - sygnał zaburzony, $f(t) * g(t)$ – sygnał wygładzony (odszumiony), tj. splot funkcji.

Na wykresie (a) łatwo dostrzec, że funkcja splotu nie jest gładka, a ekstrema sygnału wygładzonego i niezaburzonego w większości nie pokrywają się, natomiast ogólny kształt funkcji odszumionej przypomina kształt funkcji oryginalnej. W przypadku, gdy ilość punktów wejściowych została zwiększona do $N_k = 2^{10}$, możemy dostrzec, że splot jest dużo gładszy niż w poprzednim przypadku. Niektóre z ekstremów praktycznie pokrywają się z ekstremami sygnału niezaburzonego. W ostatnim przypadku splot już jest najgładszy, a sygnał odszumiony najbardziej przypomina sygnał niezaburzony. Jednak nadal nie jest to odwzorowanie idealne, ciągle nie wszystkie ekstrema się pokrywają. Między wykresami (b) i (c) różnica nie jest już tak bardzo widoczna, jak pomiędzy (a) i (b).

3 Wnioski

Szybka transformacja Fouriera nie pozwoliła na na uzyskanie dokładnego wyglądu początkowej funkcji, a jedynie na jego przybliżenie. Zwiększenie ilości próbek wejściowych pozwoliło na zwiększenie jakości odwzorowania. Największe pokrycie sygnału wygładzonego z niezaburzonym można zaobserwować w ekstremach globalnych. W przypadku mniejszego k funkcja splotu nie była gładka, co mogło być pozostałością po szczątkowych szumach. Jednak w każdym z rozważanych przypadków kształt wykresu funkcji odszumionej był w dużym stopniu zbliżony do oryginału.

Reasumując, przy użyciu szybkiej transformacji Fouriera można w skuteczny sposób dokonać odszumienia sygnału i uzyskać przybliżenie funkcji niezaburzonej.

4 Bibliografia

- Szybka transformacja Fouriera. Wikipedia, wolna encyklopedia. [online]. [dostęp: 19.05.2019]. Dostęp w Internecie: `https://pl.wikipedia.org/wiki/Szybka_transformacja_Fouriera`
- Chwiej Tomasz: Szybka transformacja Fouriera (FFT – Fast Fourier Transform). [online]. [dostęp: 19.05.2019]. Dostęp w Internecie: `http://galaxy.agh.edu.pl/~chwiej/mn/fft_1819.pdf`