UMEÅ UNIVERSITY
Department of Physics
Numerical Methods in Physics

March 22, 2015

Fast Fourier transforms

Janis Erdmanis
akels14@gmail.com

code directory **/home/jaer0031/Desktop/lab1-fourier** Program description is given in last section.

1 GSL FFT routines

The GSL defines forward Fourier transform from input signal $x_k = x(t_k)$ in following manner:

$$X_j = \sum_{k=0}^{N-1} x_k \exp(-2\pi ijk/N) \qquad k = 0, \dots, N-1; j = -\frac{N}{2}, \dots, \frac{N}{2} \tag{1}$$

from which follows that inverse transform is:

$$x_k = \frac{1}{N} \sum_{j=-N/2}^{N/2} X_j \exp(2\pi ijk/N) \tag{2}$$

We can also give interpretation for X_j as approximation to continious Fourier transform since:

$$\hat{x}(f_j) = \int_{-L/2}^{+L/2} x(t) e^{-2\pi itf_j} = e^{+2\pi iLf_j/2} \int_0^L x(t-L/2) e^{-2\pi itf_j} = \Delta t e^{+2\pi iLf_j/2} X_j \tag{3}$$

where frequency f_j because of GSL notation is defined as:

$$f_j = \frac{j}{\Delta t N} \qquad j = -\frac{N}{2}, \dots, \frac{N}{2}, \tag{4}$$

2 The Fourier transform of a Gaussian

The programm calculates Fourier transform of Gaussian function:

$$x(t) = \frac{1}{\sqrt{\pi\sigma^2}} e^{-t^2/\sigma^2} \tag{5}$$

with a sample of size $N = 1024$ for values of t:

$$t_k = \frac{k - N/2}{N} \qquad k = 0, \dots, N-1 \tag{6}$$

so the $\Delta t = 1./N = 1./1024$.

After applying Fourier transform for this data with GSL routine we obtain that spacing between frequencies for transformed data according toeq. (4) is given as:

$$\Delta f = \frac{1}{\Delta t N} = \frac{1}{(1./1024)1024} = 1.Hz \tag{7}$$

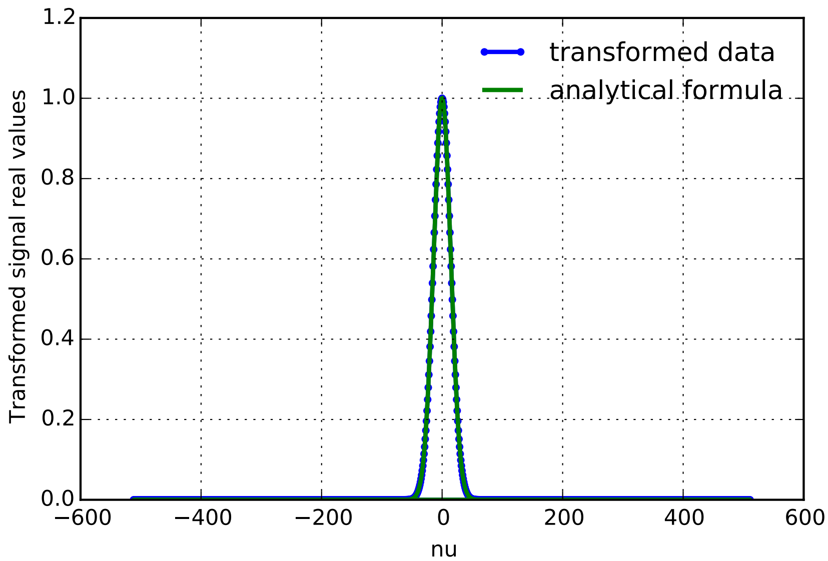

Figure 1: Frequency spectrum real values of given gaussian signal. We see that it agrees with given formula $e^{\frac{1}{4}(2\pi\sigma f)^2}$.

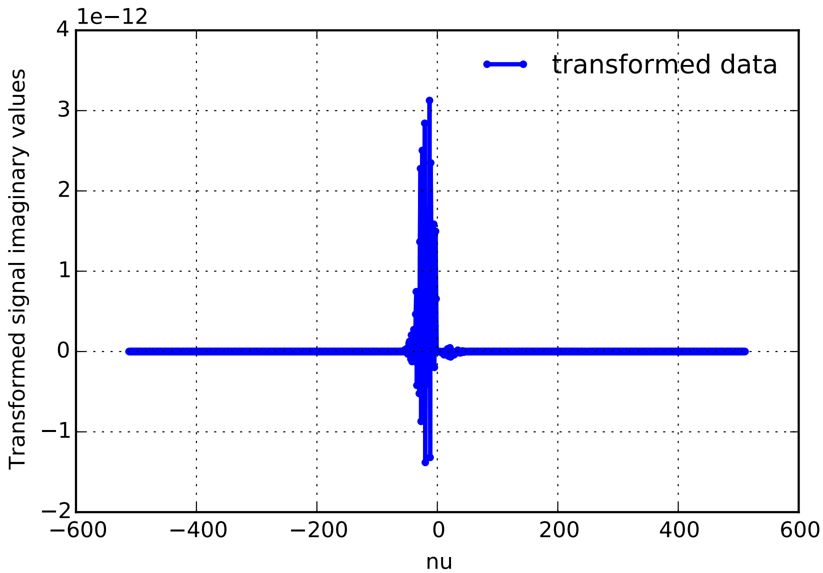

Figure 2: Frequency spectrum imaginary values. Since input was given symmetric then it is expected that imaginary part is 0 so the implementation is correct

3 The spectrum of a simple AM wave

The input signal is given as:

$$u(t) = \bar{u}(t)\sin(2\pi f_c t) \tag{8}$$

where $\bar{u}(t)$ resembles encoded binary signal. The sample size is $N = 1024$ and $f_c = 1/(128\Delta t) = 8Hz$ so there are 8 available bits for $\bar{u}(t)$.

The test signal which is used is:

$$01010101 \tag{9}$$

where position of each bit is given as $t mod 1/f_c$ but 0 and 1 represent when the $\bar{u}(t)$ has a low value $\bar{u}_0$ or high value $\bar{u}_1$.

For band width estimation we need to express the amplitude signal $\bar{u}(t)$ as fourrier transform. Because of special form of input signal we can represent it as:

$$\bar{u}(t) = \{sawtooth\} + \{constant\ part\} \tag{10}$$

The *constant part* is responsible for the main peak of the frequency spectrum because from it $u(t)$ is harmonic signal with frequency $f_c = 8Hz$. The *sawtooth* however is responsible to the sidebands because if we epress it with fourier series we get the terms for $u(t)$ like:

$$\sin(2\pi f_{saw} t)\sin(2\pi f_c t) \tag{11}$$

Conecting with discussion in the instruction we would get two frequencies $f_c + f_{saw}$ and $f_c - f_{saw}$. Since the *sawtooth* signal can be approximated with harmonic signal which as frequency $4Hz$ then it is also the frequency for the sideband.

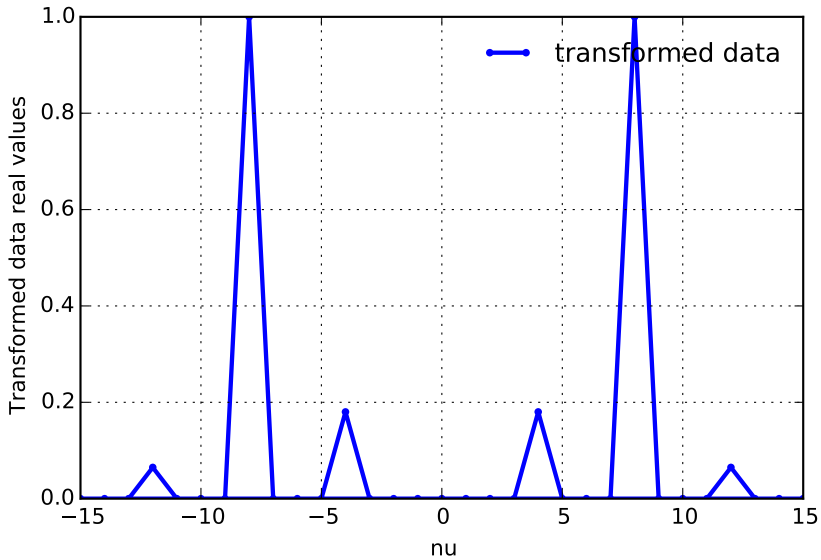

Figure 3: Modulated signal eq. (9) with $\bar{u}_0 = 1$ and $\bar{u}_1 = 3$.

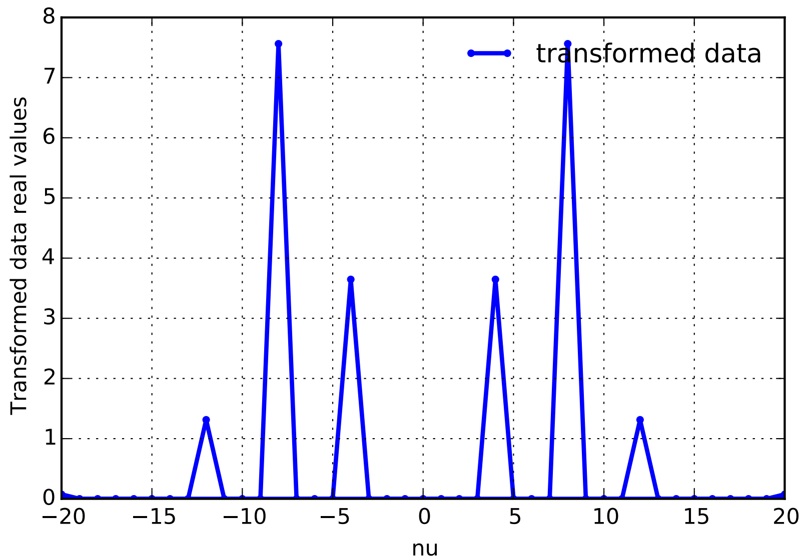

Figure 4: Modulated signal eq. (9) with $\bar{u}_0 = 1$ and $\bar{u}_1 = 10$. We see that sidebands are seperated with the same frequency as before, the amplitude of overall signal is increased, and the sidebands have become proportionaly higher.

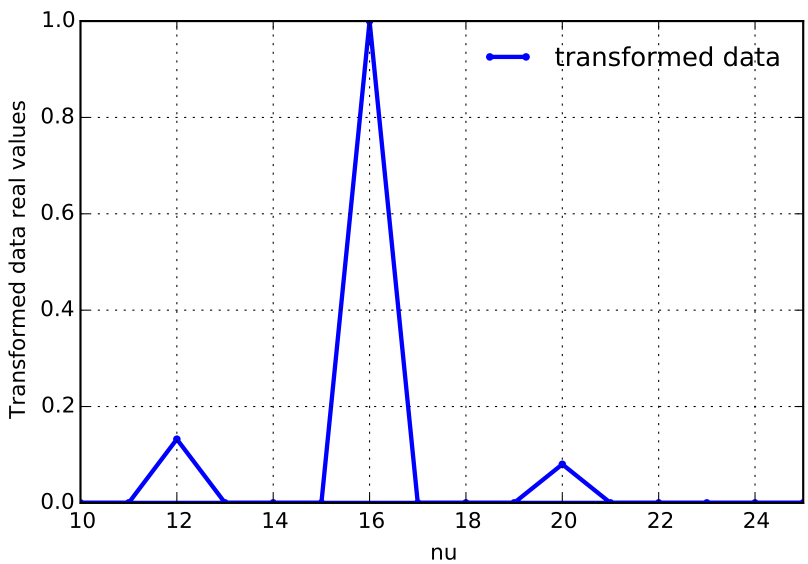

Figure 5: Modulated signal eq. (9) with $\bar{u}_0 = 1$ and $\bar{u}_1 = 3$ but with $f_c = 16Hz$.

4 Extracting information of noisy signal

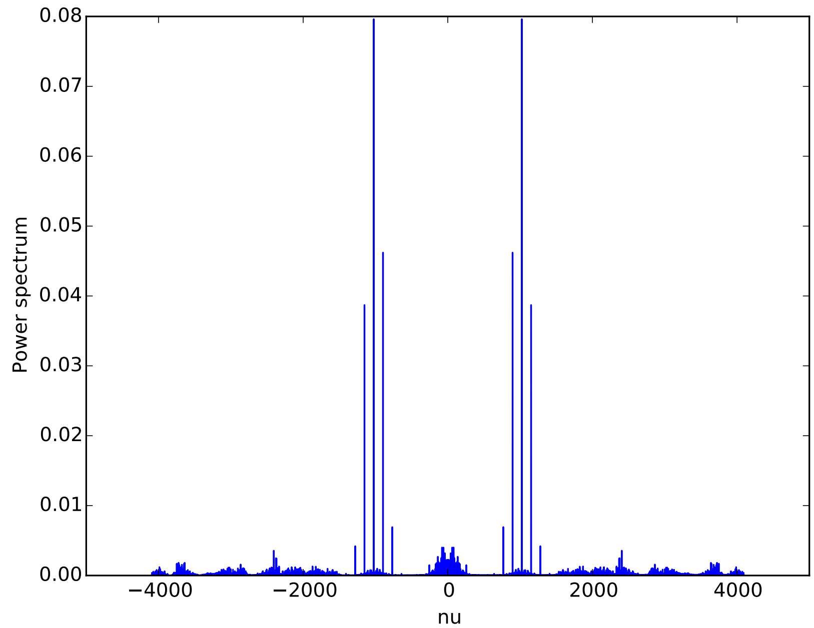

Figure 6: The power spectrum of given signal **am04.dat**. We see that the signal has quite periodic structure arround $1000Hz$ but it's sideband width about $\Delta f = 100Hz$

Due to large figure for noisy and extracted signal, they are available under `decode/results.pdf` The representation of filtered signal in binary format is:

$$01000110010011100100110100110001001101010110001001 11 \tag{12}$$

$$010000111011010010100101001101101100110010010010 \tag{13}$$

$$11011000110100101000101101 \tag{14}$$

which according to `http://www.roubaixinteractive.com/PlayGround/Binary_Conversion/Binary_To_Text.asp` gives the decoded signal:

$$FNM15bt; JSiQ \tag{15}$$

5 Programming specificaction

`gaussian/fourier`
type program
output creates two files `transformed-signal` and `original-signal`
purpose gives the results used in section section 2
compilation execute `make` in the programs directory

`afterproc.py`
type python program
execution run `python afterproc.py` in main directory. Note that python packages `pandas` and `matplotlib` must be installed for the python environment from which program is executed
input text files generated by C programs
output figures
purpose to make the figures for tabulated output data. Purpose similar as spreadsheet application.

`modulation/fourier`
type program
input given in the preprocessor
output creates two files `transformed-signal` and `original-signal`
purpose gives the results for section section 3
compilation execute `make` in the working directory

`decode/fourier`
type program
input the signal file `am04.dat` in the working directory
output creates two files `transformed-signal` and `original-signal`
purpose applies window of the input data frequency spectrum and gives back filtered signal. Program was used to produce the results in section section 4.
compilation execute `make` in the working directory