Computational Physics III: Fourier transforms and analysis

Due on March 30, 2023

Salomon Guinchard

Contents

Problem 1

Discrete Fourier transform

The Fourier transform is a mathematical operator that changes the way a complex valued function defined over the real space $\mathbb{R}^n$ is described, by describing the function in terms of the frequencies that it contains. Formally, the Fourier transform of a function f is defined as follows:

$$\mathcal{F}(f)(\mathbf{k}) := \hat{f}(\mathbf{k}) = \int_{\mathbb{R}^n} f(x) \cdot e^{-2\pi i \mathbf{k}\cdot\mathbf{x}} d\mathbf{x}, \tag{1}$$

where $\mathbf{k}$ is an element from the so-called Fourier space, or frequency space, and $\mathbf{x} \in \mathbb{R}^n$. For the sake of simplicity, in what follows, we will focus only on the 1-dimensional case, so the Fourier transform reads

$$\mathcal{F}(f)(k) := \hat{f}(k) = \int_{R} f(t) \cdot e^{-2\pi i k t} dt. \tag{2}$$

The discrete Fourier transform aims at computing the integral from Eq.(2) over a discrete set of points. To do so, one needs to discretize the real and the Fourier spaces, that is describe f over a discrete set of points $f \to f(t_n)$, $n \in \mathbb{N}$. However, in order for the discrete transform to be implemented numerically, the function can only be described on a finite set of points. Choosing the same number of grid points (N) for both the real and Fourier spaces, the discrete formulation of Eq.(2) reads

$$\hat{f}[m] = \sum_{n=0}^{N-1} f[n] e^{-2\pi i m n/N}, \tag{3}$$

where $f(t_n) = f[n]$ and $\hat{f}(k_m) = \hat{f}[m]$. Then, the numerical implementation of the above Eq.(3) is straightforward, and the algorithm is shown below:

1- Discrete Fourier transform algorithm

Listing 1: Matlab script for the discrete Fourier transform algorithm

```
function discrete_fourier = mydft(F)
    N    = length(F);    % spatial domain
    temp = zeros(size(F));
    for m = 0:(N-1)      % Fourier space variable
        for n = 0:(N-1) % real space variable associated to domain of F
            temp(m+1)= temp(m+1) + F(n+1)*exp(-2*pi*1j*m*n/N);
        end
    end
    discrete_fourier = temp;
end
```

Problem 2

NMR spectroscopy

The structure of molecules and solids can be determined by mean of NMR spectroscopy. To do so, the transitions between different quantum states of nuclear spins (of protons) in an externally applied magnetic field are studied. More precisely, the spins are excited by radio-frequency waves and their relaxation is recorded. The recorded signal is called the free-induction decay (FID). The Fourier transform of the FID yields the NMR spectrum.

The NMR spectrum relates the intensity of the signal as a function of the chemical shift δ, that we defines as follows:

$$\delta = \frac{f - f_0}{\nu_0}, \tag{4}$$

where f denotes the frequency, that will be of course discretized, f_0 the internal reference, and ν_0 the operating frequency of the NMR spectrometer. All frequencies are expressed in the same order of magnitude for the units (e.g. Hz).

1 - FID of $C_2H_6O_2$

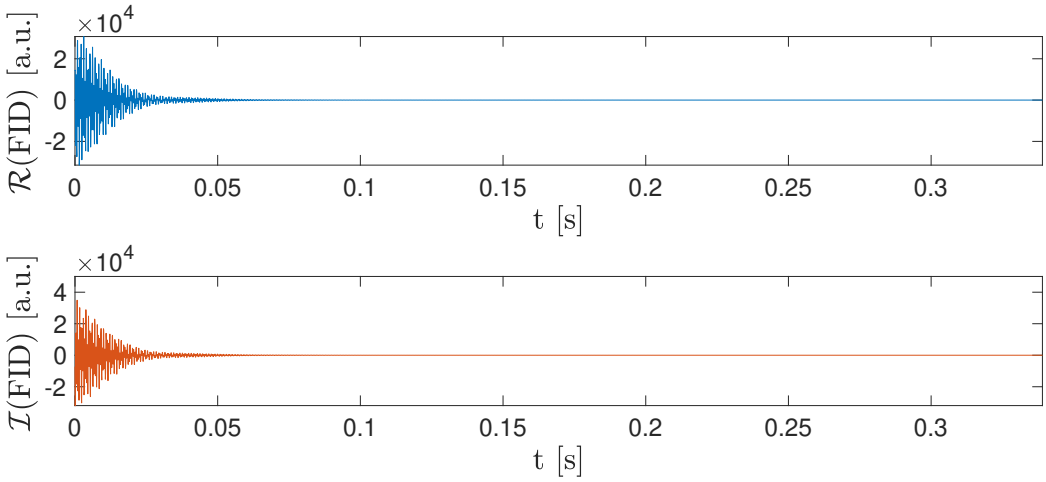

Figure 1: Upper: plot of the real part $\mathcal{R}(FID)$ of the free-induction decay over time. Lower: plot of the imaginary part $\mathcal{I}(FID)$.

2 - Nyquist and discrete frequencies for Fourier transform

Recall that the Nyquist-Shannon sampling theorem states that the sampling rate Δt has to be at most $\Delta t = 1/(2f_c)$, where f_c is the Nyquist frequency, for the signal to be properly reconstructed, and to avoid aliasing issues. Thus, for $\Delta t = 83.2 \cdot 10^{-6}$s, one obtains that $f_c = 6.01 \cdot 10^3$Hz. The discrete frequencies spanning the interval $[-f_c, f_c]$ are then defined by $f_n := (-N/2 + n) \cdot (2/N \cdot f_c)$, with $n \in \{0, \dots, N-1\}$.

Using the previously defined discrete frequencies, it is possible to discretize the chemical shift δ_n from Eq.(4), and changing f by f_n. In our problem, the internal reference is taken as $f_0 = 1067.93$Hz and $\nu_0 = 800.224 \cdot 10^6$Hz.

3 - Protons and chemical environment

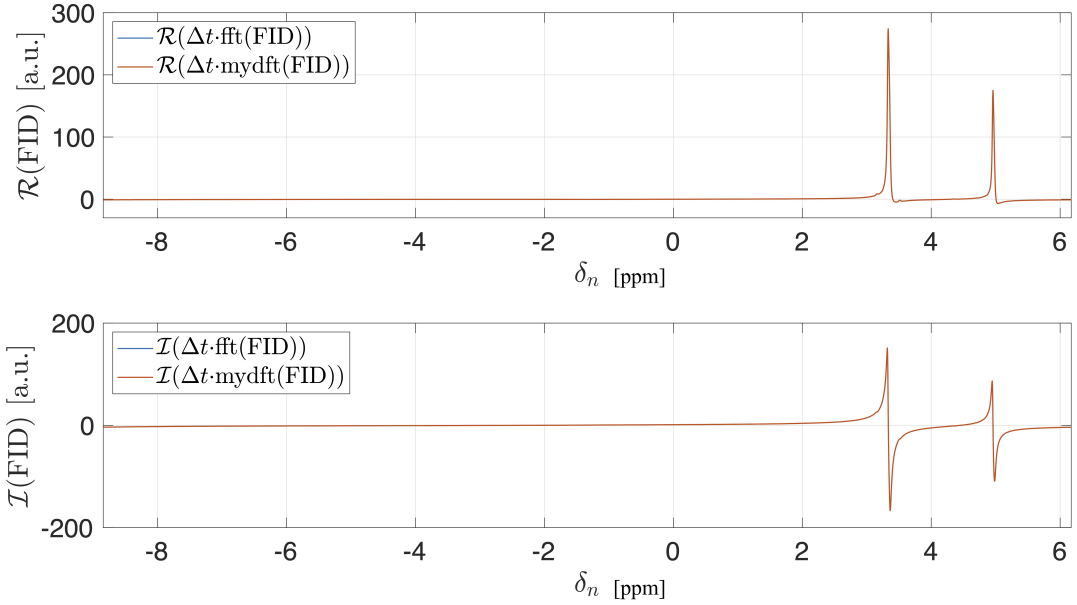

Figure 2: Upper: plot of the real part $\mathcal{R}(FID)$ of the free-induction decay over time. Lower: plot of the imaginary part $\mathcal{I}(FID)$.

Fig.(2) shows the real and imaginary parts of the Fourier transform of the FID, as a function of the chemical shift. It displays the comparative results for both our implementation of the discrete Fourier transform *mydft*, and the Matlab function *fft*. Note the perfect overlap of the curves. The obtained results are identical, but the difference between the two algorithms lays in the fact that the computational complexities are not the same, and so neither are the computational times. This will be dealt with later, when introducing the **Radix-2 algorithm** for the Fast Fourier transform.

Fig.(3) shows the power spectrum of the Fourier transform of the FID as well as the relative error ϵ_r over the power spectrum between *mydft* and *fft*, taking *fft* as the reference:

$$\epsilon_r = \left| \frac{\text{mydft} - \text{fft}}{\text{fft}} \right|. \tag{5}$$

One notes that as said earlier, the results are identical. The error is slightly more important in the region of the peaks, and the average relative error seems to increase with δ.

As of the chemical environment of the protons in that molecule, one notes that the upper plot from Fig.(3) has two very distinct peaks at $\delta_1 = 3.33$ ppm and $\delta_2 = 4.96$ ppm respectively. Thus those two peaks are characteristics of two different chemical environment for the protons. The latter represents all the atoms the proton is related to. It is not only the particles it is chemically bonded with, but all the particles that it is close to. The area under the peaks in the power spectrum is proportional to the square of the number of

protons in each environment. Since we're not able with those data only to determine the scaling law, we'll take the ratio between the two areas, both defined by

$$\mathcal{A}_i = \int_{\Omega_i} d\delta \mathcal{P}(\delta), \tag{6}$$

with $\Omega_i = [\delta_i - \Delta\delta, \delta_i + \Delta\delta]$ $i = 1, 2$ for an appropriate $\Delta\delta$. Thus, taking the ratio $r = \mathcal{A}_1/\mathcal{A}_2$, one gets $r = 1.73$. Then, there are approximately 1.73 more protons in the first chemical environment than in the second.

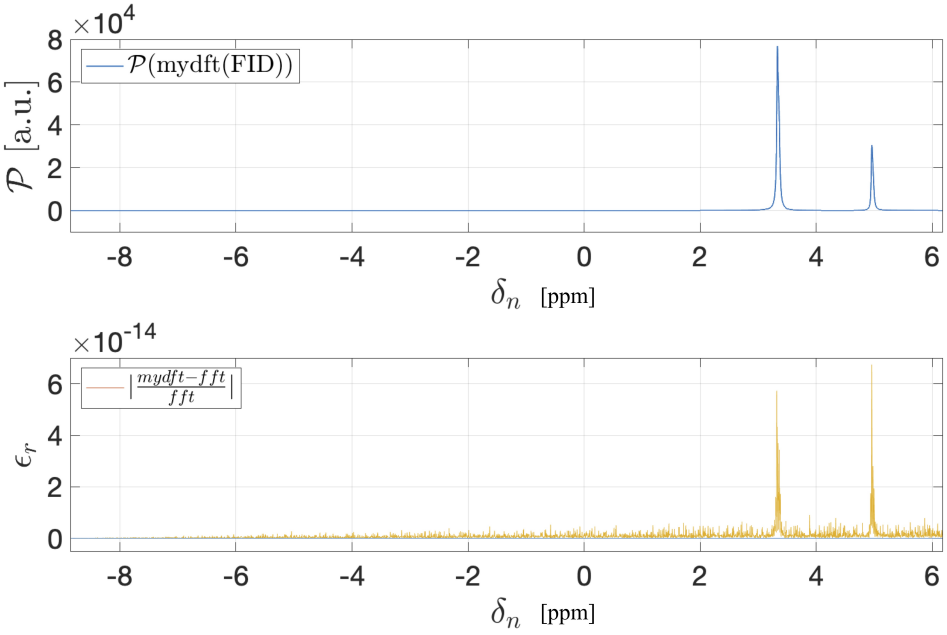

Figure 3: Upper: plot of the real part $\mathcal{R}(FID)$ of the free-induction decay over time. Lower: plot of the imaginary part $\mathcal{I}(FID)$.

Considering that C, O and H atoms are able to form covalent bonds with 4, 2 and 1 neighbors respectively, it is possible to suggest the structural formula of the molecule $C_2H_6O_2$. Indeed, recall from the previous part that the molecule had two different chemical environment for protons, with about $1.73 \sim 2$ times more protons in one environment than the other. This means that for a molecule with 6 H atoms, equivalent to protons after forming a covalent bond, there has two be 4 H atoms with the same type of covalent bond, and 2 with a different one. Therefore, we can come up with the two following possible structures:

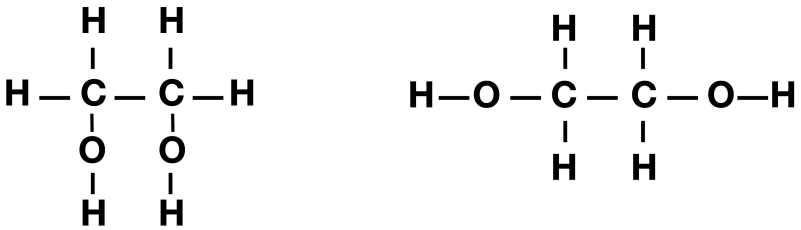

Figure 4: Two possible molecular structures for $C_2H_6O_2$

Problem 3

Radix-2 FFT algorithm

1- Radix-2 Fast Fourier transform algorithm

Listing 2: Matlab script for the Radix-2 Fast Fourier transform algorithm

```
function result=myfft(f)% f is an array input
    result = zeros(size(f));
    N = length(f);

    r=log2(N); % checking whether N is a power of two or not. Mandatory for
    % this algo.
    if abs(r-uint64(r))>1e-5 %arbitrary threshold
        error('N has to be a power of 2.');
    end

    if N==1
        result = f;%this is computed directly from the formula
    elseif N==2
            result(1) = f(1) + f(2);%this is computed directly from the
            % formula
            result(2) = f(1) - f(2);
    else
        %recursive part, divide to conquer
        e = myfft(f(1:2:end)); %even part
        o = myfft(f(2:2:end)); %odd part
        for k=0:N/2-1
            W = exp(-2*1i*pi*k/N);
            result(k+1) = e(k+1) + W*o(k+1);
            result(k+1+N/2) = e(k+1)-W*o(k+1); %as usual, careful with the
            % indices in Matlab, not the same as python
        end
    end
end
```

Problem 4

Fourier Ptychography

Fourier ptychograpghy enables to overcome the problem of limited resolution in conventional optical microscopy by generating a wider aperture, that is by allowing a wider range of angles under which the light can be captured by the microscope. Indeed, denoting the resolution by $\mathcal{R}$, one gets the following relation:

$$\mathcal{R} \propto \frac{\lambda}{n \sin(\theta)}, \tag{7}$$

where λ is the wavelength of the incoming light, n the refractive index of the lens and θ the maximum angle of acceptance. It is then trivial that increasing θ will increase $\mathcal{R}$.

In order to understand the Fourier ptychography algorithm, let us recall how a typical optical microscope works. Considering that the studied object can be represented by a 2-dimensional complex function O, that is the real space image is given by $O(x, y)$, where (x, y) is an appropriate coordinate choice. Thus, the reciprocal (Fourier space) image is given by $\mathcal{F}(O)(k_x, k_y) \cdot a(k_x, k_y)$, where $\mathcal{F}$ represents the Fourier transform operator and a an amplitude transfer function of finite radius. The latter can be approximated by a low-pass filter of radius r_c, mapping to 1 below r_c and 0 otherwise. The real image is then reconstructed after the light passes through the second lens, and the detector will record an intensity $I \propto |\mathcal{F}^{-1}\Big(\mathcal{F}(O)(k_x, k_y) \cdot a(k_x, k_y)\Big)|^2$.

In Fourier ptychography, the process is slightly different since instead of using perpendicular plane waves $\propto e^{ik_{zi}z}$ as illumination sources, several images are taken with tilted light sources $\propto e^{ik_{xi}x} e^{ik_{yi}y} e^{ik_{zi}z}$. Then by the transfer theorem from Eq.(**??**), the Fourier transform is shifted as follows:

$$\mathcal{F}(O)(k_x - k_{xi}, k_y - k_{yi}), \tag{8}$$

and hence, the intensity recorded by the microscope verifies

$$I \propto \left|\mathcal{F}^{-1}\Big(\mathcal{F}(O)(k_x - k_{xi}, k_y - k_{yi}) \cdot a(k_x, k_y)\Big)\right|^2. \tag{9}$$

The cutoff circle has then been shifted by (k_{xi}, k_{yi}) and some higher frequencies were involved in the image formed. Thus, taking several images under tilting angles enables to reconstruct the Fourier transform with a higher cutoff radius, and enhance the resolution.

Numerical implementation of Fourier ptychography algorithm

In the next part, the following reconstruction algorithm will be used:[IS16]

1. Start with a (complex) guess function I and compute $\mathcal{F}(I)$.
2. Restrict $\mathcal{F}(I)$ to a circle in Fourier space, centered around (k_{xi}, k_{yi}) multiplying it by a cutting function: $\mathcal{F}(I) \to \mathcal{F}(I)_c = \mathcal{F}(I) \cdot C_{(k_{xi}, k_{yi})}$

3. Take its inverse Fourier transform $\mathcal{F}^{-1}(\mathcal{F}(I)_c)$
4. Retain the phase but replace the magnitude by the experimental one
5. Perform the Fourier transform of the resulting object and use it to replace the values of $\mathcal{F}(I)$ inside the cutoff circle
6. Repeat the previous steps for the the circles in Fourier space, that is for different values of (k_{xi}, k_{yi})
7. Repeat the previous algorithm till convergence is reached

1 - Phase and Intensity

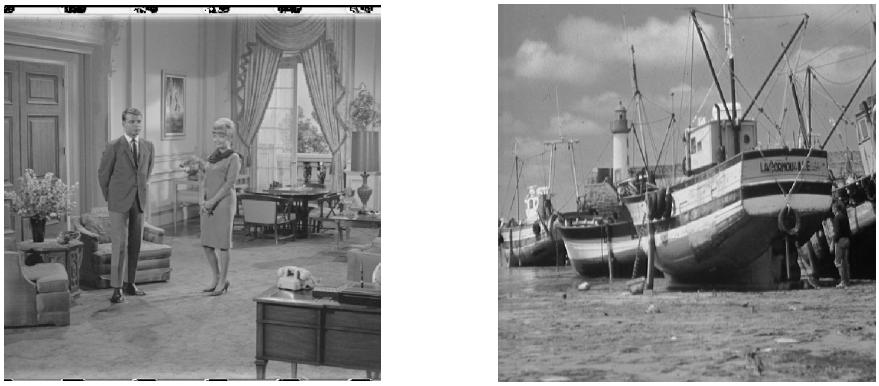

Figure 5: Left: Original intensity image - Right: Original phase image

Fig.(5) displays both the original intensity and the original phase. By mean of Fourier ptychography, on some simulated microscope images, we will try to retrieve the phase and enhance the resolution. The implementation of the above algorithm will be described and the results will be presented below.

Microscope images

The microscope images were obtained by taking the Fourier transform of the original objects, and by applying circular filters in Fourier space, before taking the inverse Fourier transform and record the new intensity. Fig.(**??**) shows an example of such a 'microscope' image. One notes that the result obtained in Fourier space is not contained in the original circular low pass filter. This is due to the fact that when

References

[IS16] Toshitaka Idehara and Svilen Petrov Sabchevski. Gyrotrons for high-power terahertz science and technology at FIR UF. *Journal of Infrared, Millimeter, and Terahertz Waves*, 38(1):62–86, oct 2016.