Università degli Studi di Trento – Dipartimento di Fisica

Computational Physics 1

Recitation class, April 28, 2021

Fast Fourier Transform

Indice

1 A code

What does the following code do? In particular, what is the relation between the plot of `f[]` defined in the first `for(n=0;n<N;n++)` cycle and that of `f[]` after the last cycle? Why?

```
#define N 256
double f[2*N];

for(n=0;n<N;n++)
  {
    x[n] = n * dx;
    sigma = 1.0;
    f[2*n] = exp(-pow((x[n] - R/2.0),2.0)/(2.0*sigma*sigma));
    sigma = 3.0;
    f[2*n] += 0.25*exp(-pow((x[n] - 2.0*R/3.0),2.0)/(2.0*sigma*sigma));
    f[2*n+1] = 0.0;
  }

gsl_fft_complex_radix2_forward (f, 1, N);

for(k=0;k<N;k++)
  {
    double c, s, zre, zim;

    c = cos((2.0*M_PI/N)*k*64.0);
```

```
    s = sin((2.0*M_PI/N)*k*64.0);
    zre = f[2*k];
    zim = f[2*k+1];

    f[2*k]   = zre * c - zim * s;
    f[2*k+1] = zre * s + zim * c;
  }

gsl_fft_complex_radix2_inverse (f, 1, N);

for(n=0;n<N;n++)
  {
    printf("%g \t %g\n",n*dx,f[2*n]);
  }
```

2 Gravity waves

The dispersion relation of surface gravity waves in a liquid is

$$\omega^2(k) = gk\tanh(kh), \tag{1}$$

where $g = 9.81$ m/s^2 is the gravity acceleration and h the depth. Consider a wavepacket of surface waves moving in a region of depth $h = 10$ m, whose shape at time $t = 0$ is the real part of

$$f(x) = \exp\left[-\frac{(x-x_0)^2}{8\lambda^2}\right]\exp\left(i\frac{2\pi(x-x_0)}{\lambda}\right), \tag{2}$$

with $\lambda = 1$ m. Compute the temporal evolution in an interval $T = 1000$ s. If the wavepacket is moving towards a beach, which wavelengths will arrive first: longer or shorter?

You will notice that during the temporal evolution, the wavepacked will disperse, moving in the positive x direction and re-entering from the opposite side. In order to better visualize the dispersion, you can consider a reference system where the packet is still. The average velocity of a wave packet is the group velocity, given by

$$v_{\mathrm{g}} = \left.\frac{d\omega}{dk}\right|_{k=k_{\max}}, \tag{3}$$

where $k_{\max}$ is the wavevector of the maximum of the Fourier transform of (2). Use the results of exercise 1 to visualize the wavepacket in the reference system moving with speed v_{g}.

3 Numerical derivatives using FFT

Write a routine that computes the numerical first derivative of a function using FFT techniques. We recall that if we represent a function as

$$f_n = \sum_{k=0}^{N-1} F_k \exp\left(i\frac{2\pi kn}{N}\right)$$

its derivative is given by

$$df_n = \sum_{k=0}^{N-1} F_k K_k \; \exp\left(i\frac{2\pi kn}{N}\right)$$

where

$$K_k = \begin{cases} i\dfrac{2\pi k}{N} & \text{se } k \leq \dfrac{N}{2} \\[2ex] i\dfrac{2\pi}{N}\left(k-N\right) & \text{se } k > \dfrac{N}{2} \end{cases}$$

Extend it to compute the second derivative.

4 Transmission and reflection: time-dependent approach

How would you compute transmission and reflection coefficients of a quantum particle impinging on a 1D barrier by solving the time-dependent Schrödinger equation using the split-operator techique? Test your ideas and the effectiveness of your computer code with the exact solution for the rectangular barrier and the numerical solution for the Gaussian barrier using the time-independent (Numerov) approach.

Before writing and running the program: how do you envision the temporal evolution of a Schrödinger wavepacket encountering a repulsive barrier?