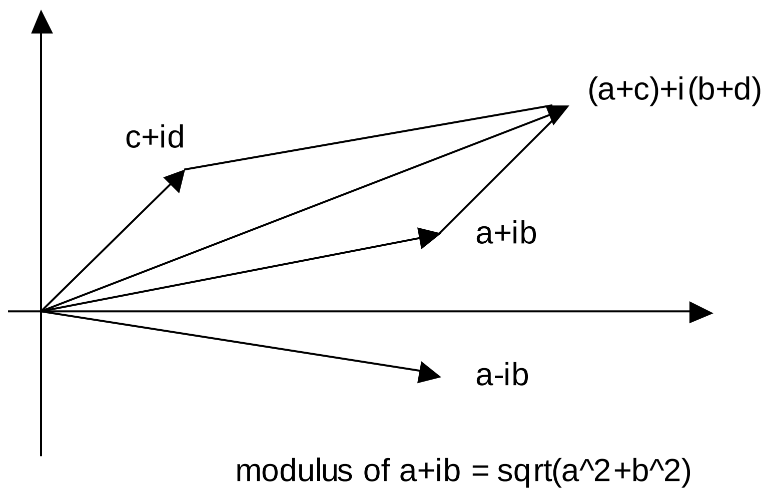
Lecture 5: Complex Numbers and Fourier Transforms

The imaginary number ***i = sqrt(-1)*** is not a real number because $i^2 = -1$ is not positive. The **complex numbers** are the set whose elements $a + bi$ where a and b are any real numbers, called the real and imaginary parts of $a + bi$. One can view them as vectors in the complex plane where a is on the horizontal axis and b is on the vertical axis. Then define **addition** of two complex numbers as vector addition, the **modulus** of the complex number as the length of the vector and the **conjugate** of the complex number as a complex number corresponding to the reflection about the horizontal axis.



The product and division of numbers are defined by using $i^2 = -1$:

$$(a+ib)(c+id) \equiv (ac - bd) + i(ad+bc) \text{ and}$$
$$(a+ib)/(c+id) \equiv (a+ib)/(c+id)\,(c-id)/(c-id)$$
$$= (ac + bd)/(c^2 + d^2) + i(-ad+bc)/(c^2 + d^2).$$

Using these definitions one can compute more complicated functions of complex variables. A very important function is the exponential function and we use its series representation

$$\begin{aligned} e^{ix} &= 1 + (ix)^1/1! + (ix)^2/2! + (ix)^3/3! + (ix)^4/4! + \cdots \\ &= 1 + (ix)^2/2! + (ix)^4/4! + \cdots \\ &\quad + (ix)^1/1! + (ix)^3/3! + \cdots \\ &= \cos(x) + i \sin(x) \quad (Euler's\ Formula). \end{aligned}$$

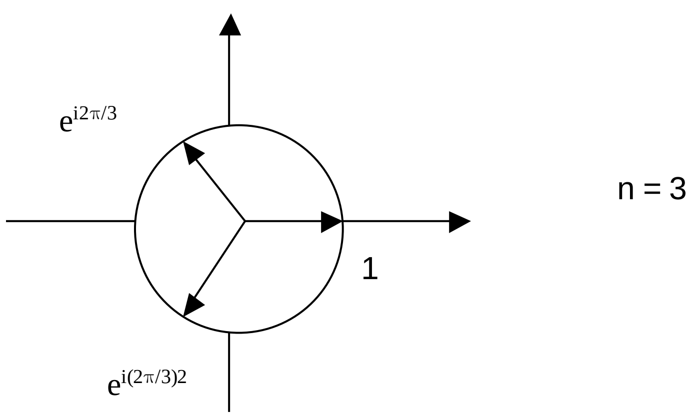
The modulus of e^{ix} is equal to one because $sin^2(x) + cos^2(x) = 1$. Also x can be viewed as the angle from the real axis to the vector representing the complex number e^{ix}. An important case is when $x = 2\pi/n$ so that $w \equiv e^{i2\pi/n}$ and $z \equiv w^* = e^{-i2\pi/n}$ satisfy:

$$w^n = 1,\ z^k = (w^*)^k = w^{-k}\ and\ 1 + w + w^2 + \cdots + w^{n-1} = 0.$$

The last property is established by using $w^n = 1$ and noting **w** is not equal to 1:

$$\begin{aligned} w(1 + w + w^2 + \cdots + w^{n-1}) &= w + w^2 + \cdots + w^n \\ &= w + w^2 + \cdots + w^{n-1} + 1 \\ (w-1)(1 + w + w^2 + \cdots + w^{n-1}) &= 0 \text{ so that for } w-1 \text{ not equal to zero} \\ 1 + w + w^2 + \cdots + w^{n-1} &= 0. \end{aligned}$$

The complex numbers w^j with $j = 0,..,n-1$ are called the n^{th} roots of unity because they solve $x^n = 1$. For $n = 3$, they the following vectors whose vector sum is zero.



Application of Complex Numbers to Laplace Transform.

Consider the Laplace transform if the complex exponential function e^{ibt}. Here we make the "natural" extension to complex integrals! By Euler's formula

$$\begin{aligned} L(e^{ibt}) &= L(\cos(bt) + i\sin(bt)) \\ &= L(\cos(bt)) + iL(\sin(bt). \end{aligned}$$

Apply the definition of Laplace transform directly to e^{ibt}

$$\begin{aligned} L(e^{ibt}) &= \int_0^\infty e^{-st} e^{ibt}\, dt \\ &= \lim_{N\to\infty} \int_0^N e^{-st} e^{ibt}\, dt \\ &= \lim_{N\to\infty} \int_0^N e^{(-s+ib)t}\, dt \\ &= \lim_{N\to\infty} \left[e^{(-s+ib)N} \frac{1}{-s+ib} - e^{(-s+ib)0} \frac{1}{-s+ib} \right] \\ &= \frac{1}{s-ib} \\ &= \frac{1}{s-ib} \frac{s+ib}{s+ib} \\ &= \frac{s}{s^2+b^2} + i\frac{b}{s^2+b^2}. \end{aligned}$$

By equating the real and imagery parts of $\mathbf{L(e^{ibt})}$, we have derived the Laplace transforms for the sine and cosine function. These rules could also have been derived by the direct application of the definition of the Laplace transform to the sine and cosine functions.

The Fourier Transform.

The Fourier transform of a function *f(t)* could be viewed as a variation on the Laplace transform where the real parameter s is replaced by an imaginary parameter $i\omega$ and the domain of integration is extended to the entire real line.

Defintion. The **Fourier transform** of *f(t)* is

$$F(f) = \int_{-\infty}^{\infty} e^{-i\omega t} f(t)dt.$$

Basic Properties of Fourier Transform.

1. $F(cf) = c\, F(f),$
2. $F(f + g) = F(f) + F(g),$
3. $F(f') = -iw\, F(f)$ *and*
4. $F(f * g) = F(f)\, F(g)$ *where*

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

Application to Solution of –u$_{xx}$ + cu = f.

Replace the variable *t* with *x* and compute the Fourier transform of both sides in the differential equation. By using the third property, the derivative property, twice we obtain

$$F(-u_{xx} + cu) = F(f)$$
$$-(-i\omega)^2 F(u) + cF(u) = F(f)$$
$$(w^2 + c)F(u) = F(f)$$
$$F(u) = F(f)\frac{1}{w^2 + c}.$$

Apply the fourth property, the convolution property, to conclude

$$u = f * g \quad where \quad F(g) = \frac{1}{\omega^2 + c}.$$

Discrete Fourier Transform.

The discrete Fourier transform is derived by truncating the integral to [0 1], replacing ω by 2πk and letting $f_j = f(j/n)$. The **discrete Fourier** transform of $f = [f_0\, f_1\, \ldots\, f_{n-1}]$ is complex vector whose k^{th} component is

$$\sum_{j=0}^{n-1} e^{-i(2\pi k)j/n} f_j = \sum_{j=0}^{n-1} e^{-i(2\pi/n)kj} f_j$$

$$= \sum_{j=0}^{n-1} z^{kj} f_{j.}$$

Another way to view this is as a matrix product from vectors in real n dimensional space to vectors in complex n dimensional space

$$\mathbb{F}f \quad \textit{where } \mathbb{F} \textit{ has kj component } z^{kj}.$$

For example, if $n = 3$, then

$$\mathbb{F} = \begin{bmatrix} z^0 & z^0 & z^0 \\ z^0 & z^1 & z^2 \\ z^0 & z^2 & z^4 \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 \\ 1 & z^1 & z^2 \\ 1 & z^2 & z^1 \end{bmatrix}.$$

If $f = [1\ 7\ 2]'$, $n = 3$ and $z = e^{-i2\pi/3} = -1/2 - \sqrt{3}/2i$, then

$$\mathbb{F}f = \begin{bmatrix} 1 & 1 & 1 \\ 1 & z^1 & z^2 \\ 1 & z^2 & z^1 \end{bmatrix} \begin{bmatrix} 1 \\ 7 \\ 2 \end{bmatrix} = \begin{bmatrix} 10 \\ 1 + 7z + 2z^2 \\ 1 + 7z^2 + 2z \end{bmatrix} = \begin{bmatrix} 10 \\ -3.50 - 4.33i \\ -3.50 + 4.33i \end{bmatrix}.$$

In applications n is typically very large and an efficient method to do these computations is required. In Matlab the command fft() is an implementation of the fast Fourier transform. This requires $(n/2)\log_2(n)$ operations, which is much less than the $2n^2$ operations for the matrix-vector product. The following are some simple examples.

```
>> fft([1 7 2])   % n = 3

    10.0          -3.5000 - 4.3301i  -3.5000 + 4.3301i
```

Matlab Code ffttrig.m

```
t = 0:.001:1;    % n = 1001
freq = 1:1:1001;
fftsin = fft(4*sin(2*pi*40*t));
fftcos = fft(8*cos(2*pi*100*t));
subplot(2,1,1);
plot(freq,real(fftsin),freq,real(fftcos));
subplot(2,1,2);
plot(freq,imag(fftsin),freq,imag(fftcos));
```

The spikes in the Fourier transform plots identify the frequencies *f* and *n*-*f*.