Fourier Optics Classes 1-6: The Basics

Anand Sivaramakrishan, STScI, 2018 Oct-Dec

Prerequisites:

```
astroconda with python 3 on your laptop

matrixDFT from https://github.com/mperrin/poppy/blob/master/poppy/matrixDFT.py
(there is a maintained version inside STScI, but we are only using this one
routine from poppy.  Get it any way that's convenient.)

Get the python sources utils.py and telclass.py.

Basic familiarity with complex numbers

Familiarity with DS9, image display stretches (linear, log,..) and matplotlib's
imshow

Time: 8 hours per week: preparatory work prior to the weekly class, and, post-
class, for expanding and writing up work.
```

Course organization

```
Pre-class work will get you acquianted with the ideas/questions and try out some
exercises.  It is not expected that you will "finish" the problems set for this
prep work, but the prep work is essential for the class.  At the start of the
class you will all discuss this prep work with each other, and to air your
confusion (or clarity) on the concepts and details.  This is a major vehicle of
learning in the class.

After the in-class discussion I will go over points that need clarification or
more in-depth understanding.  That, and setting the stage for the next pre-class
prep work will take the remaning part of the class.

After class you should finish the exercises associated with the prep work (e.g.
in an annotated Jupyter notebook).  This notebook will serve as your course
notes, and hopefully a valuable reference for you and a lasting legacy of the
course.
```

Class Schedule

Date	Class #	Topics
Oct 24	#1	Intro, class software, EM waves, scalar field, far field, the Fraunhofer approximation behind Fourier optics. The Fourier transform.
Oct 31	#2	Simulating the image plane of a given telescope pupil. Limits to discrete Fourier transforms: effects of sampling, and finite information input. Equivalent Width Theorem.
Nov 7	#3	Matching theory and numerics, PSFs and imaging, asymptotic behavior, apodization, Nyquist sampling and aliasing, imperfect images, tilt, speckles
Nov 14	#4	Wavefront sensing: focus sweeps, Gerchberg Saxton focus-diverse phase retrieval basics
Nov 21	-	No class
Nov 28	#5	
Dec 5	#6	
Dec 12	Back-up	

Class 1: Theoretical basis, numerical Fourier machinery

This is a hands-on course, but you should have some idea of the theoretical basis for numerical manipulations are, in case you need to delve deeper in the future. We will go quickly over the basics, and outline how we get to the stage of calculating the response of many optical systems to a point source or a sky scene.

We will cover the mathematical definition of the Fourier transform, and a couple of its fundamental properties.

Theoretical work

Look up **Fourier transforms** and **Fourier optics**, and [re]familiarize yourself with the key concepts/phrases.

The 1-dimensional Fourier transform. What sort of function (real, complex,. ...) is the function

(e.g. f(x)) and its transform (F(k))? Are the domains of f and F (the spaces that x or k live in) real or complex? Are f and F, deep down, different descriptions of the same information?

Does the theoretical Fourier transform of a function lose informaton that was present in the original function?

Incoherent vs. **coherent** emission of light from distinct points an **extended object**. Does an atom on one side of a star emit spectral line radiation in phase (coherent with) another atom of the same element on the other side of the star, emitting light in the same spectral line? Do then two atoms communicate with each other to collaborate on releasing radiation coherent with each other? How is this different from a gas in a laser cavity?

Diffraction limit or **[angular] resolution** of a telescope, lambda/D

The Fraunhofer or **far field** situation, and the **Fresnel length** that governs when the far field limit calculations are valid. Fresnel length ~ D^2 / lambda. Parse out the meaning of this in terms of the angular resolution.

Describing light numerically/mathematically: a plane wave of monochromatic light (in a homogenous or non-dispersive medium or vacuum) is a propagating oscillation of the electric and magnetic fields. In (x,y,z) physical space z is the direction of propagation. (x,y) is the plane transverse to the propagation direction. The wave can be expressed as the real part of a complex number that has a (real) amplitude A(x,y) and a unit-strength "phasor" exp(j(kz - wt + phi).

We average over many periods of the waves' oscillation in a typical optical or IR measurement. We also "follow a wave's crest", and ignore the (kz - wt) in the exponent. One resource is ApJ vol. 552 pp.397-408, 2001 May 1, Section 2.1 equation 1 and surrounding very short text.

What mathematical/numerical operation must you perform on a complex array describing the EM wave's "complex amplitude" to get a real array describing the **intensity** (brightness) that a CCD or IR array might detect?

The concept of the **pupil** (or aperture) plane of an optical system, and its illumination by **monochromatic light** from a distant star.

Describing the pupil quantitatively (eg. equation 2 of the above paper). If you create a numpy array that represents a circular mirror telescope without a secondary obstruction, what are the physical dimensions you assign (in your mind) to this in-memory array --- what physical thing does the numerical array span? What physical quantity is represented by a number (an element) in this numerical array?

Post-class-1 work

```
Look at and run code/telclass.py

See hoe the matrixDFT object is instantiated, a pupil created, and an imasge
complex amplitude ("field") and its intensity are calculated.

Look through the parameters of exer1() and its output in datadir/c1/ex1*.fits

Write up a short description of why the images appear as they do.  Consider
information content in the two domains, pupil and image.
```

I will upload a sample program for creating various phase array creation by the end of today
(Oct 24) to help you work through the following exercises:

```
Create images (fits files) using a 100x100 pupil array with a pupil radius=20.0,
focalplane_size=100, focalplane_npix=100, use a non-zero (always real) phase
array.  Create image intesity fits files corresponding to the following cases:

phase = a non-zero constant - the "reference" on-axis image.

phase = a sloped array of the form a*i + b*j (i,j are indices of the array, a
and b constants).  Keep one of them (eg b) at zero, vary a till you see images
that differ noticeably from the rference image

phase = some sinusoidal function (of i only) - a ripple in the "i" direction.
Use ripples with 2, 3, 4, 5 periods across the "active" pupil (its diameter is
50 pixels, since you used radius=25).  Make the amplitude of the sinusoid 0.1,
0.3, 1, and 3 (these are amplitudes in radians of phase)

phase = the sum of a ripple in the "i" direction and one in the "j" direction,
with amplitudes of e.g. 0.3
```