Installing grtrans

- From a terminal run:
  git clone https://github.com/jadexter/grtrans
- Instructions in grtrans_tutorial.pdf file, briefly here
- Install cfitsio
- Install pyfits
- cp Makefile.top.sample Makefile.top and edit Makefile.top
- make

Install cfitsio

- In a terminal:
- wget http://heasarc.gsfc.nasa.gov/FTP/software/fitsio/c/cfitsio3390.tar.gz
- cd cfitsio
- ./configure; make -j4; make install
- cd ..

Compile grtrans

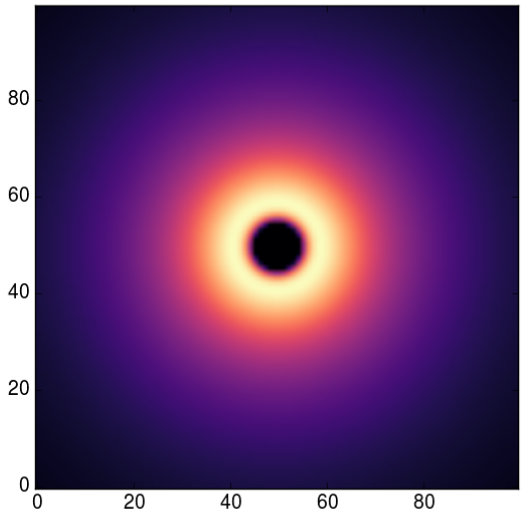
- cp Makefile.top.sample Makefile.top
- Edit Makefile.top to include paths to grtrans and cfitsio directories
- make -j4

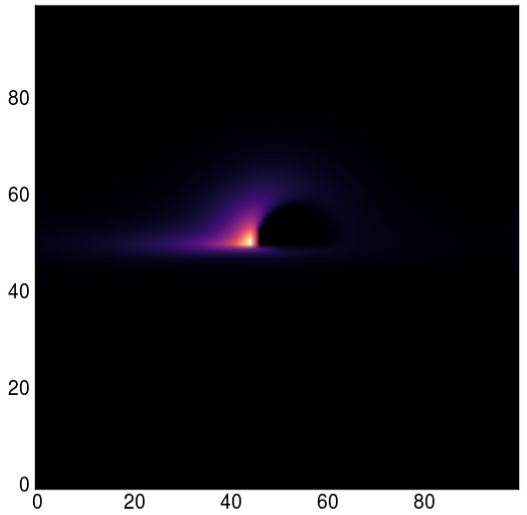
grtrans via python (pgrtrans)

- Best supported / easiest way is to use python 2.7 to run grtrans
- Alternative: inputs.in input file, run grtrans from command line, write to FITS format output
- For python: install pyFITS, pip install pyfits
- Also requires numpy, matplotlib

Running pgrtrans

- in grtrans directory, run ipython (or use a jupyter notebook)
- import grtrans_batch as gr
- x = gr.grtrans()
- x.run_pgrtrans()
- x.disp_pgrtrans_image(-1)
- x.disp_pgrtrans_image(0)
- Default: thin disk total intensity images vs. inclination

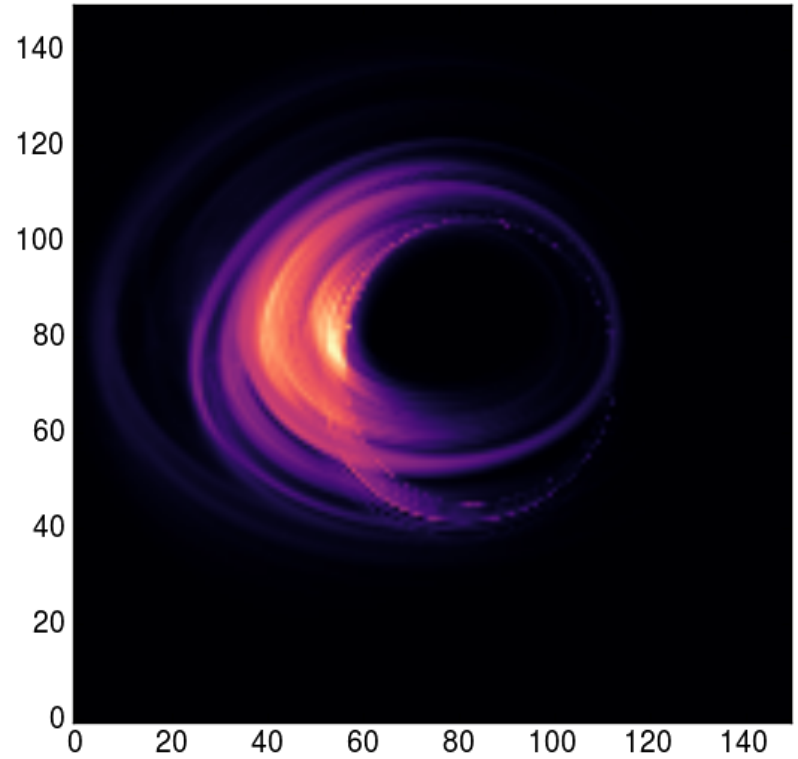




A HARM image

- x.run_pgrtrans(standard=1,nvals=1,fname='HARM',nfreq=1,fmin=2.3e11,fmax=2.3e11,ename='POLSYNCHTH',spin=0.9375,nn=[150,150,400],uout=0.04,mbh=4e6, mdotmin=4e15, mdotmax=4e15,nmdot=1, nmu=1,mumin=.6428,mumax=.6428, gridvals=[-13.,13.,-13.,13.], fhfile='dump040', fdfile='dump',findf=40,fnt=1,muval=1./4.,gmin=1.)

- x.disp_pgrtrans_image(0)



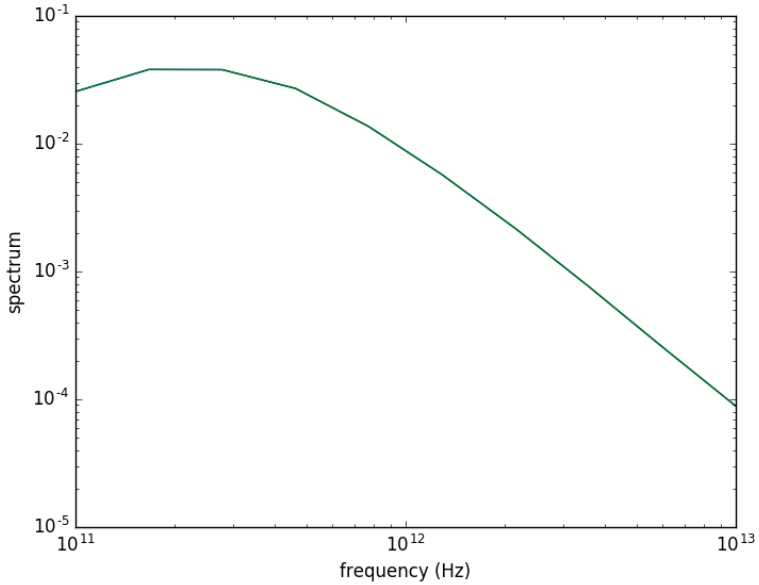
A HARM image

- inputs:
  standard: 1 is normal ray tracing, 2 is equatorial plane only
  nvals: 1 for Stokes I, 4 for full polarization
  fname: fluid model, analytic or simulation
  fmin,fmax,nfreq: observed nu (Hz) for nfreq frequencies
  ename: emissivity (mostly various synchrotron emissivities)
  spin: a/M in Kerr
  mbh: black hole mass in Msun
  nn: camera resolution # of pixels (x,y), # of points along each ray
  uout: 1/r_out, the maximum radius used in rad. trans. calculation
  mumin, mumax, nmu: cos(inclination) for nmu angles
  mdotmin,mdotmax,nmdot: same for mdot
  gridvals: camera size in units of M, [xmin, xmax, ymin, ymax]
  fhfile, fdfile, findf: fluid model specific inputs (e.g. dump file name, header file name)
  fnt: = 1 “fast light”, > 1 load fnt files than one dump file for slow light
  muval: constant fraction of internal energy in e-, 1 / (Tp/Te + 1)

- (good / bad): Many others!

Calculating spectra

- re-run, nfreq = 10, fmin = 5e10, fmax = 1e13
- x.calc_spec_pgrtrans(x.nx)
- import matplotlib.pyplot as plt
- plt.loglog(x.nu,x.spec)



HARM polarization maps

- same as before, but change
  nvals = 4
- x.disp_pol_map(0)
- Now set Faraday
  coefficients = 0: add input
  emiscoefindx=[1,1,1,1,0,0,0]
- x.disp_pol_map(0)