```
import matplotlib
import numpy as np

import matplotlib.pyplot as plt
import math


from novas import compat as novas
from novas.compat import eph_manager
from novas.compat import *


def check_val(x):
    if np.abs(x) > 1:
        y=1
    else:
        y=x
    return y




def toradians(angle):
    pi=3.14159265358979
    x=angle*pi/180.0
    return x

def todegrees(angle):
    pi=3.14159265358979
    x=angle*180.0/pi
    return x

def hms(h, m, s):
    hours=(s+60*(m+60*h))/3600
    return hours
```

```
def dms(d, m, s):
     dms=(d+(m+s/60)/60)
     return dms

def degprint(d):
    dd=int(d)
    min=(d-dd)*60
    mm=int(min)
    sec=(min-mm)*60
    a=str(dd)+':'+str(mm)+':'+str(sec)
    print(a)
    return d


def star_coord(body, year):
    mnum=1
    day=1
    print(body,  year)
    if body=='Hamal':

        star=CatEntry("Hamal".encode(), "HIP".encode(), 0, 2.1195571389, 23.4624175556,188.55,-148.08,49.56,-14.2)
    elif body=='Aldebaran':

        star=CatEntry("Aldebaran".encode(), "HIP".encode(), 1, 4.5986776944, 16.5093023611,63.45,-189.94,49.97,54.26)
    elif body=='Pollux':

        star=CatEntry("Pollux".encode(), "HIP".encode(), 2, 7.7552638528, 28.0261988889, -626.55,-45.80,96.54,3.23)
    elif body=='Regulus':

        star=CatEntry("Regulus".encode(), "HIP".encode(), 3, 10.1395308333, 11.9672083333, -248.73,5.59,41.13,5.9)
    elif body=='Spica':
```

```
    star=CatEntry("Spica".encode(), "HIP".encode(), 4, 13.4198830556, -11.1613194444,-42.35,-30.67,13.06,1.0)
elif body=='Antares':

    star=CatEntry("Antares".encode(), "HIP".encode(), 5, 16.4901276944, -26.4320026111,-12.11,-23.30,5.89,-3.4)
elif body=='Altair':

    star=CatEntry("Altair".encode(), "HIP".encode(), 6, 19.8463884861, 8.8683211944, 536.23,385.29,194.95,-26.1)
elif body=='Fomalhaut':

    star=CatEntry("Fomalhaut".encode(), "HIP".encode(), 7, 22.96084625, -29.6222361111, 328.95,-164.67,129.81,6.5)
elif body=='Markab':

    star=CatEntry("Markab".encode(), "HIP".encode(), 8, 23.0793481806, 15.2052671389, 60.40, -41.30, 24.46, -2.2)
elif body=='Achernar':

#Achernar
    ra=hms(1, 37,42.84548 ) #01h 37m 42.84548
    dec=-dms(57, 14,12.310 )#−57° 14′ 12.3101″
    proper_motion_ra=87.00 #
    proper_motion_dec=-38.24
    parallax=23.39
    radial_velocity=16

    star=CatEntry("Achernar".encode(), "HIP".encode(), 9, ra, dec, proper_motion_ra, proper_motion_dec, parallax, radial_velocity)

elif body=='Alioth':
    #Alioth
    ra=hms(12, 54,1.74959 ) #12h 54m 01.74959s[
    dec=dms(55, 57,35.3627 )#55° 57′ 35.3627
    proper_motion_ra=111.91#111.91
    proper_motion_dec=-8.24#-8.24[
    parallax= 39.51
    radial_velocity=-9.3#-9.3

    star=CatEntry("Alioth".encode(), "HIP".encode(), 10, ra, dec, proper_motion_ra, proper_motion_dec, parallax, radial_velocity)
elif body=='Acrux':
    #Acrux
    ra=hms(12, 26,35.89522 ) #12h 26m 35.89522
    dec=-dms(63, 5,56.7343 )#−63° 05′ 56.7343
    proper_motion_ra=-35.83#
    proper_motion_dec=-14.86
    parallax= 10.13
    radial_velocity=-11.2

    star=CatEntry("Acrux".encode(), "HIP".encode(), 11, ra, dec, proper_motion_ra, proper_motion_dec, parallax, radial_velocity)

elif body=='Alpheratz':
    #Alpheratz
    ra=hms(0, 8,23.25988) #00h 08m 23.25988
    dec=dms(29, 5,25.5520 )#29° 05′ 25.5520
    proper_motion_ra=135.68#135.68
    proper_motion_dec=-162.95#−162.95
    parallax= 33.62
    radial_velocity=-10.6#−10.6

    star=CatEntry("Alpheratz".encode(), "HIP".encode(), 12, ra, dec, proper_motion_ra, proper_motion_dec, parallax, radial_velocity)
elif body=='Arcturus':
#Arcturus
    ra=hms(14,15,39.7) #14h 15m 39.7
    dec=dms(19, 10,56 )#19° 10′ 56
    proper_motion_ra=-1093.45#−1093.45
    proper_motion_dec=-1999.40#−1999.40
    parallax= 88.83
    radial_velocity=-5.19

    star=CatEntry("Arcturus".encode(), "HIP".encode(), 13, ra, dec, proper_motion_ra, proper_motion_dec, parallax, radial_velocity)
elif body=='Betelgeuse':
#Betelgeuse
```

```
    ra=hms(5,55,10.30536) #5h 55m 10.30536
    dec=dms(7, 24, 25.4304 )#07° 24′ 25.4304
    proper_motion_ra=26.42#26.42
    proper_motion_dec=9.60#9.60
    parallax= 4.51
    radial_velocity=21.91

    star=CatEntry("Betelgeuse".encode(), "HIP".encode(), 14, ra, dec, proper_motion_ra, proper_motion_dec, parallax, radial_velocity)
elif body=='Canopus':
#Canopus
    ra=hms(6,23,57.10988) #6h 23m 57.10988
    dec=-dms(52, 41, 44.3810)#−52° 41′ 44.3810″
    proper_motion_ra=19.93#19.93
    proper_motion_dec=23.24#23.24
    parallax= 10.55
    radial_velocity=20.3

    star=CatEntry("Canopus".encode(), "HIP".encode(), 15, ra, dec, proper_motion_ra, proper_motion_dec, parallax, radial_velocity)

elif body=='Capella':
#Capella
    ra=hms(5,16,41.35871) #5h 16m 41.35871
    dec=dms(45, 59, 52.7693)#45° 59′ 52.7693
    proper_motion_ra=75.52#75.52
    proper_motion_dec=-427.11# −427.11
    parallax= 76.20
    radial_velocity=29.9387

    star=CatEntry("Capella".encode(), "HIP".encode(), 16, ra, dec, proper_motion_ra, proper_motion_dec, parallax, radial_velocity)
elif body=='Deneb':
#Deneb
    ra=hms(20,41,25.9) #20h 41m 25.9
    dec=dms(45, 16, 49)#45° 16′ 49
    proper_motion_ra=1.99#1.99
    proper_motion_dec=1.95#23.24
    parallax= 2.29
    radial_velocity=-4.5

    star=CatEntry("Deneb".encode(), "HIP".encode(), 17, ra, dec, proper_motion_ra, proper_motion_dec, parallax, radial_velocity)
elif body=='Dubhe':
#Dubhe
    ra=hms(11,3,43.67152) #11h 03m 43.67152
    dec=dms(61, 45, 3.7249)#61° 45′ 03.7249
    proper_motion_ra=-134.11#−134.11
    proper_motion_dec=-34.70# −34.70
    parallax=       26.54
    radial_velocity=-9#−9

    star=CatEntry("Dubhe".encode(), "HIP".encode(), 18, ra, dec, proper_motion_ra, proper_motion_dec, parallax, radial_velocity)

elif body=='Hadar':
#Hadar
    ra=hms(14,3,49.40535) #14h 03m 49.40535
    dec=-dms(60, 22, 22.9266)#−60° 22′ 22.9266
    proper_motion_ra=-33.27#−33.27
    proper_motion_dec=-23.16#−23.16
    parallax= 8.32
    radial_velocity=5.9

    star=CatEntry("Hadar".encode(), "HIP".encode(), 19, ra, dec, proper_motion_ra, proper_motion_dec, parallax, radial_velocity)
elif body=='Polaris':
#Polaris
    ra=hms(2,31,49.09) #2h 31m 49.09
    dec=dms(89, 15,50.8)#89° 15′ 50.8
    proper_motion_ra=44.48#44.48
    proper_motion_dec=-11.85#−11.85
    parallax= 7.54
    radial_velocity=-17#−17
```

```
    star=CatEntry("Polaris".encode(), "HIP".encode(), 20, ra, dec, proper_motion_ra, proper_motion_dec, parallax, radial_velocity)

elif body=='Procyon':
#Procyon
    ra=hms(7,39,18.11950) #7h 39m 18.11950
    dec=dms(5, 13, 29.9552)#05° 13′ 29.9552
    proper_motion_ra=-714.590#−714.590
    proper_motion_dec=-1036.80#−1036.80
    parallax= 284.56
    radial_velocity=-3.2

    star=CatEntry("Procyon".encode(), "HIP".encode(), 21, ra, dec, proper_motion_ra, proper_motion_dec, parallax, radial_velocity)
elif body=='Rigel':
#Rigel
    ra=hms(5,14,32.27210) #5h 14m 32.27210
    dec=-dms(8, 12, 5.8981)#−08° 12′ 05.8981″
    proper_motion_ra=1.31#1.31
    proper_motion_dec=0.50#0.50
    parallax= 3.78
    radial_velocity=17.8

    star=CatEntry("Rigel".encode(), "HIP".encode(), 22, ra, dec, proper_motion_ra, proper_motion_dec, parallax, radial_velocity)
elif body=='Rigel Kent':
#Rigel Kent
    ra=hms(14,39,36.4940) #14h 39m 36.4940
    dec=-dms(60, 50, 2.3737)#−60° 50′ 02.3737
    proper_motion_ra=-3679.25#−3679.25
    proper_motion_dec=473.67#473.67
    parallax= 754.81
    radial_velocity=-21.4

    star=CatEntry("Rigel Kent".encode(), "HIP".encode(), 23,  ra, dec, proper_motion_ra, proper_motion_dec, parallax, radial_velocity)
elif body=='Sirius':
#Sirius
    ra=hms(6,45,8.91728) #6h 45m 08.91728
    dec=-dms(16, 42, 58.0171)#−16° 42′ 58.0171
        proper_motion_ra=-546.01#−546.01
        proper_motion_dec=-1223.07#−1223.07
        parallax= 379.21
        radial_velocity=-5.50

        star=CatEntry("Sirius".encode(), "HIP".encode(), 24, ra, dec, proper_motion_ra, proper_motion_dec, parallax, radial_velocity)
    elif body=='Vega':
    #Vega
        ra=hms(18,36, 56.33635) #18h 36m 56.33635
        dec=dms(38, 47,1.2802)#38° 47′ 01.2802
        proper_motion_ra=200.94#200.94
        proper_motion_dec=286.23#286.23
        parallax=130.23
        radial_velocity=-13.9

        star=CatEntry("Vega".encode(), "HIP".encode(), 25, ra, dec, proper_motion_ra, proper_motion_dec, parallax, radial_velocity)

    jd_utc = novas.julian_date(int(year), mnum, int(day), 0)
    b_ra, b_dec = novas.app_star(jd_utc, star)
    print(b_ra, b_dec)
    return b_ra,  b_dec



matplotlib.rcParams['xtick.direction'] = 'out'
matplotlib.rcParams['ytick.direction'] = 'out'


jd_start, jd_end, number = eph_manager.ephem_open()
dummy = novas.make_cat_entry('Dummy', 'x', 0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0)

body1='Polaris'
body2='Regulus'
```

```
body3='Dubhe'
year=1945
base_hrs=14
base_min=0
base=base_hrs+base_min/60
span=49
latmin=50
c=math.pi /180
latmin_rad=latmin*c
latmax_rad=(latmin+10)*c
merc_min=math.log(math.tan(latmin_rad/2+ math.pi/4))
merc_max=math.log(math.tan(latmax_rad/2+ math.pi/4))
ra,  dec=star_coord(body1, year)
print(ra,  dec)
dummy=degprint(dec)
sha=360 - 15*ra
dummy=degprint(sha)
dec=dec*c
test=(np.arcsin(np.sin(dec)*np.sin(44*c)+np.cos(dec)*np.cos(44*c)*np.cos(((base-ra))*15*c)))/c
print(test)
dummy=degprint(test)


hr=np.arange(0, span)
#xlist=[]
mlatl=[]
i=0
while i < 51:
    mlatl.append(merc_min+(merc_max - merc_min)*i/50)
    i=i+1
mlat=np.array(mlatl)

X, Y = np.meshgrid(hr, mlat)

Z=(np.arcsin(np.sin(dec)*np.sin((2*np.arctan(np.exp(Y))-np.pi/2))+np.cos(dec)*np.cos((2*np.arctan(np.exp(Y))-np.pi/2))*np.cos(((X/60+base-ra))*15*c)))/c
```

```
V=[]
for x in range(0, 90):
    for y in range(0, 6):
        z=x+y/6
        V.append(z)


fig=plt.figure(figsize = (8.203, 14))
ax = fig.add_subplot(1,1,1)
plt.xlabel("Local Sidereal Time")
plt.ylabel("Latitude")

major_ticksl=[]

i=0
while i < 11:

    z=math.log(math.tan((latmin+i)*c/2+ math.pi/4))
    major_ticksl.append(z)
    i=i+1

major_ticks=np.array(major_ticksl)
minor_ticksl=[]
i=0
while i < 121:

    z=math.log(math.tan((latmin+i/12)*c/2+ math.pi/4))
    minor_ticksl.append(z)
    i=i+1
minor_ticks=np.array(minor_ticksl)

major_x=np.arange(5, span, 5)
```

```
minor_x=np.arange(0, span, 1)
ax.set_xticks(major_x)
ax.set_xticks(minor_x, minor=True)

ax.set_yticks(major_ticks)
ax.set_yticks(minor_ticks, minor=True)

ax2 = ax.twinx()
ax2.set_yticks(major_ticks)
ax2.set_yticks(minor_ticks, minor=True)
ax2.set_ylabel('Latitude')

ax2.set_ylim(bottom=merc_min, top=merc_max)
ax.set_ylim(bottom=merc_min, top=merc_max)

a=[]
i=0
while i < 11:
    a.append(str(latmin+i))
    i=i+1
ax.set_yticklabels(a)
ax2.set_yticklabels(a)

ax3=ax.twiny()
ax3.set_xticks(major_x)
ax3.set_xticks(minor_x, minor=True)
ax.tick_params('both', length=10, width=2, which='major')
ax.tick_params('both', length=5, width=1, which='minor')
ax2.tick_params('both', length=10, width=2, which='major')
ax2.tick_params('both', length=5, width=1, which='minor')
ax3.tick_params('both', length=10, width=2, which='major')
ax3.tick_params('both', length=5, width=1, which='minor')


CS = plt.contour(X, Y, Z, V,colors='black', alpha=0.4, orientation="portrait", )
```

```
CS = plt.contour(X, Y, Z, V[::6],colors='black', alpha=0.4, orientation="portrait")

labels1=plt.clabel(CS, CS.levels,inline=1, fontsize=10,  fmt='%1.0f')

yticks = ax.yaxis.get_ticklines(minor=True)
#ax.yaxis.set_tick_params(yticks[::4], length=8)
plt.setp(yticks[::4], 'markersize', 7)
plt.setp(yticks[::12], 'markersize', 10)

i=0
for tick in ax2.yaxis.get_minor_ticks():
#    tick.tick1line.set_markersize(0)

    tick.tick2line.set_markersize(10)
    if (i%6==0 ) :
      if (i%12 == 0):

        j=i
      else:
          tick.tick2line.set_markersize(10)

    else:
        if (i%2==0):
            tick.tick2line.set_markersize(7)
        else:
            tick.tick2line.set_markersize(5)

    i=i+1
```

```
yticks1 = ax2.yaxis.get_ticklines(minor=True )

plt.setp(yticks1[::4], 'markersize', 7)
plt.setp(yticks1[::12], 'markersize', 10)



ra,  dec=star_coord(body2, year)
print(ra,  dec)
dummy=degprint(dec)
sha=360 - 15*ra
dummy=degprint(sha)

dec=dec*math.pi/180

Z=(np.arcsin(np.sin(dec)*np.sin((2*np.arctan(np.exp(Y))-np.pi/2))+np.cos(dec)*np.cos((2*np.arctan(np.exp(Y))-np.pi/2))*np.cos(((X/60+base-ra))*15*c)))/c

V=[]
for x in range(0,90):
    for y in range(0, 6):
        z=x+y/6
        V.append(z)


CT = plt.contour(X, Y, Z, V, colors='r', alpha=0.6, orientation="portrait")
CT = plt.contour(X, Y, Z, V[::6], colors='purple', alpha=0.6, orientation="portrait")
labels1=plt.clabel(CT,CT.levels,  inline=1, fontsize=10,  fmt='%1.0f')


ra,  dec=star_coord(body3, year)
print(ra,  dec)
dummy=degprint(dec)
sha=360 - 15*ra
dummy=degprint(sha)

dec=dec*math.pi/180

Z=(np.arcsin(np.sin(dec)*np.sin((2*np.arctan(np.exp(Y))-np.pi/2))+np.cos(dec)*np.cos((2*np.arctan(np.exp(Y))-np.pi/2))*np.cos(((X/60+base-ra))*15*c)))/c

V=[]
for x in range(0,90):
    for y in range(0, 6):
        z=x+y/6
        V.append(z)


CU = plt.contour(X, Y, Z, V, colors='green', alpha=0.6, orientation="portrait")
CU= plt.contour(X, Y, Z, V[::6], colors='brown', alpha=0.6, orientation="portrait")
labels1=plt.clabel(CU,CU.levels,  inline=1, fontsize=10,  fmt='%1.0f')

plt.title('Year:'+str(year)+'  '+body1+'(black)  '+body2+'(red)  '+body3+'(green)', y=1.08)

plt.figtext(0.073, 0.072, str(base_hrs), fontsize=16, fontweight='bold')
if base_min==0:
    plt.figtext(0.11, 0.077, '00', fontsize=12, fontweight='bold')
else:
     plt.figtext(0.11, 0.077, '30', fontsize=12, fontweight='bold')
plt.figtext(0.11, 0.062, 'LST', fontsize=10, fontweight='bold')
plt.figtext(0.073, 0.92, str(base_hrs), fontsize=16, fontweight='bold')
if base_min==0:
    plt.figtext(0.11, 1-0.073, '00', fontsize=12, fontweight='bold')
else:
    plt.figtext(0.11, 1-0.073, '30', fontsize=12, fontweight='bold')
plt.figtext(0.11, 1-0.089,'LST', fontsize=10, fontweight='bold')

fname='Weemsp'+str(base)+'y'+str(year)+'.png'
plt.savefig(fname)
fname='Weemsp'+str(base)+'y'+str(year)+'.pdf'
```

```
plt.savefig(fname)
plt.show()
```