visualization

March 12, 2023

1 P46 3.6

alpha beta

1.1 2048

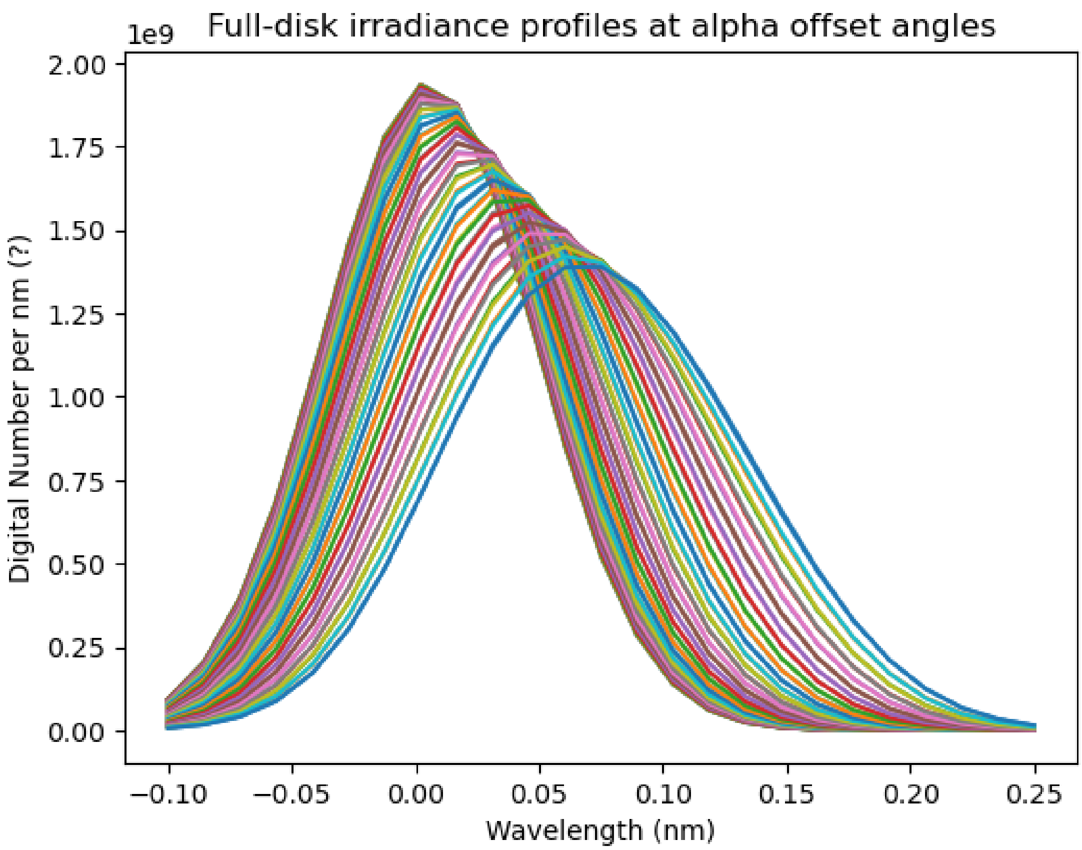
λ range: -0.1,0.25,25

[165]:

```
import numpy as np
import matplotlib.pyplot as plt
from astropy.modeling import models, fitting
import math
from calculating_DN_2048 import wavelength_point_num, wavelength_list
from calculating_DN_2048 import angle_point_num_alpha,offaxis_angle_x_alpha, offaxis_angle_y_alpha
from calculating_DN_2048 import angle_point_num_beta,offaxis_angle_x_beta, offaxis_angle_y_beta

DN = np.load("output_DN/DN_large_lambda_range_2048.npz")

# Initialize
# Cruciformscan in alpha direction
offaxis_angle_x_min_alpha=offaxis_angle_x_alpha*180*60/math.pi

# Cruciformscan in beta direction
offaxis_angle_y_min_beta=offaxis_angle_y_beta*180*60/math.pi

# Fit data in DN??.npz
wavelength_shift_alpha = np.zeros(angle_point_num_alpha)
fit_alpha = []    # List of Gaussian1D
for i in range(angle_point_num_alpha):
    g_init = models.Gaussian1D(amplitude=1E9, mean=0.05, stddev=0.0424)
    # initial value for fitting
    fit_g = fitting.LevMarLSQFitter()
    g = fit_g(g_init, wavelength_list, DN['DN_alpha'][i])
    wavelength_shift_alpha[i] = g.mean.value
    fit_alpha.append(g)
```

[166]:
```python
# Profiles during cruciformscan
fig, ax = plt.subplots()
# Profiles during cruciformscan in alpha direction
for i in range(angle_point_num_alpha):
    ax.plot(wavelength_list, DN['DN_alpha'][i], label='linear')
# ax.set_title("He II          ")
ax.set_title("Full-disk irradiance profiles at alpha offset angles")
ax.set_xlabel('Wavelength (nm)')
ax.set_ylabel("Digital Number per nm (?)")
```

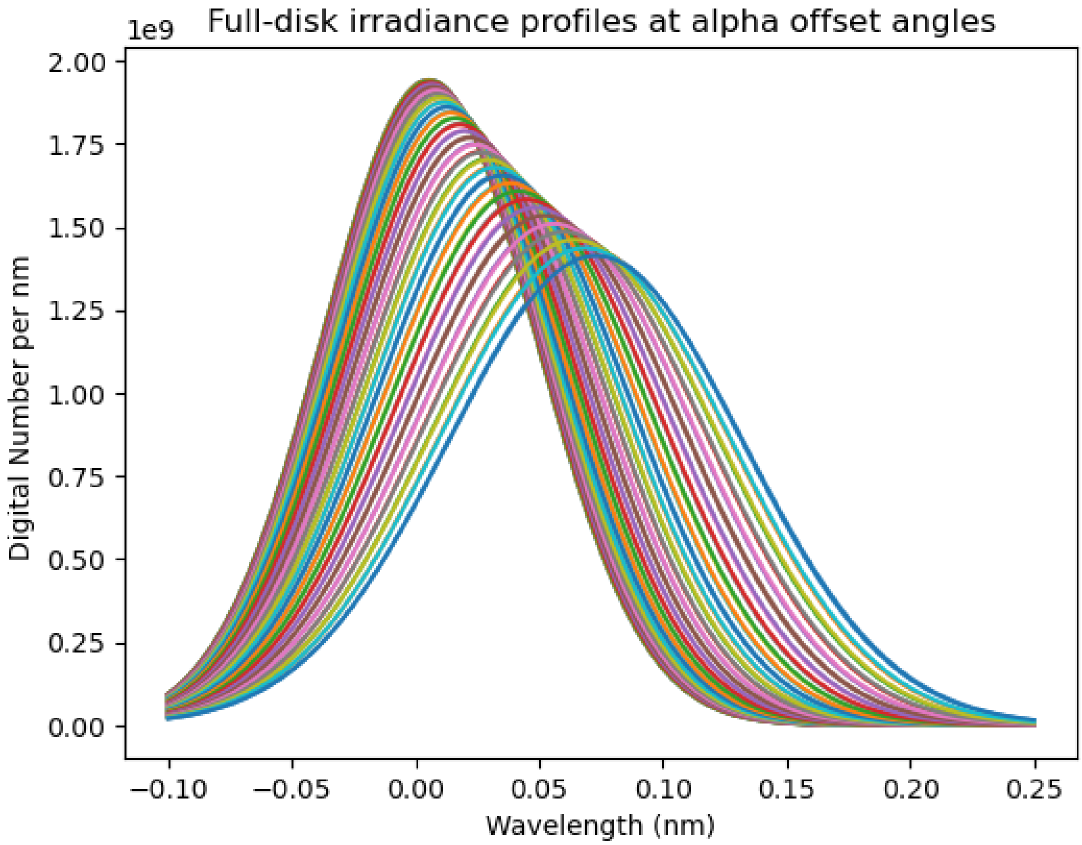
[166]:
```
Text(0, 0.5, 'Digital Number per nm (?)')
```



[167]:
```python
# Fitted profiles during cruciformscan
fig, ax = plt.subplots()
wavelength_densed_list = np.linspace(-0.1, 0.25, 10*wavelength_point_num)
# Profiles during cruciformscan in alpha direction
for i in range(angle_point_num_alpha):
    ax.plot(wavelength_densed_list, fit_alpha[i](
        wavelength_densed_list), label='linear')
```

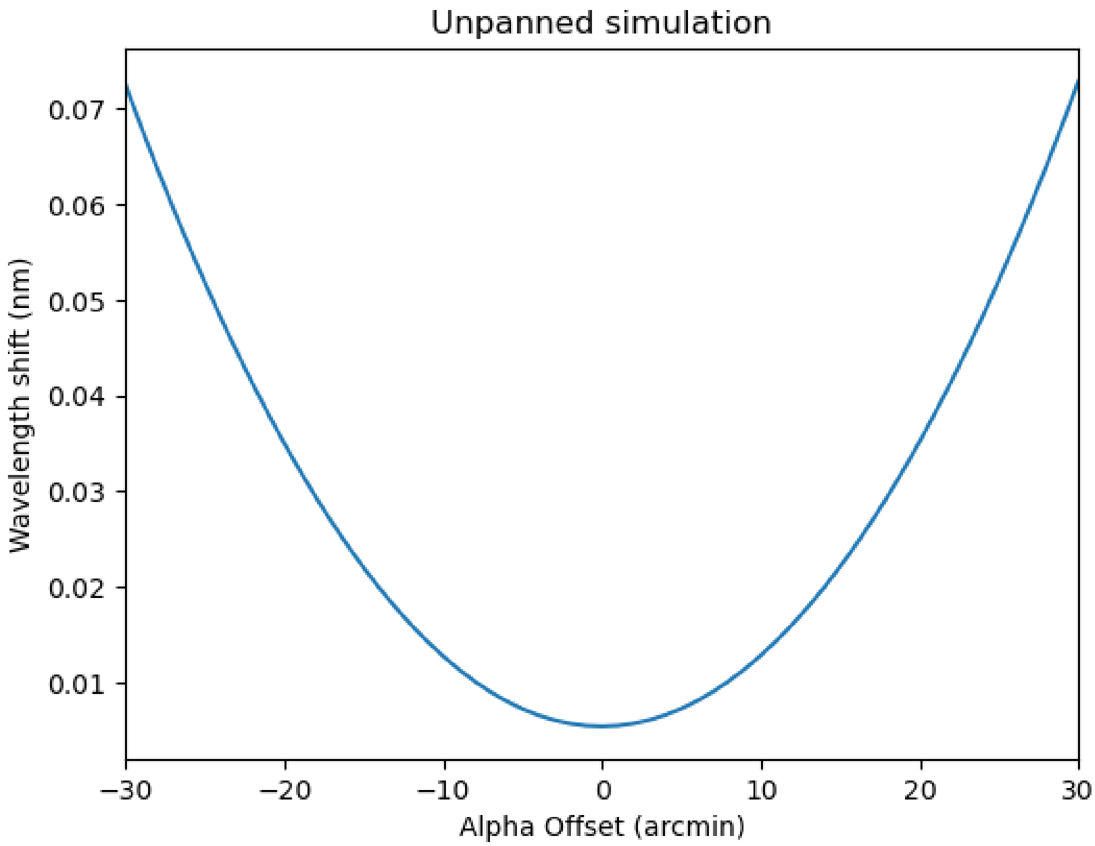
```
ax.set_xlabel('Wavelength (nm)')
ax.set_ylabel("Digital Number per nm ")
ax.set_title("Full-disk irradiance profiles at alpha offset angles")
```

[167]: Text(0.5, 1.0, 'Full-disk irradiance profiles at alpha offset angles')



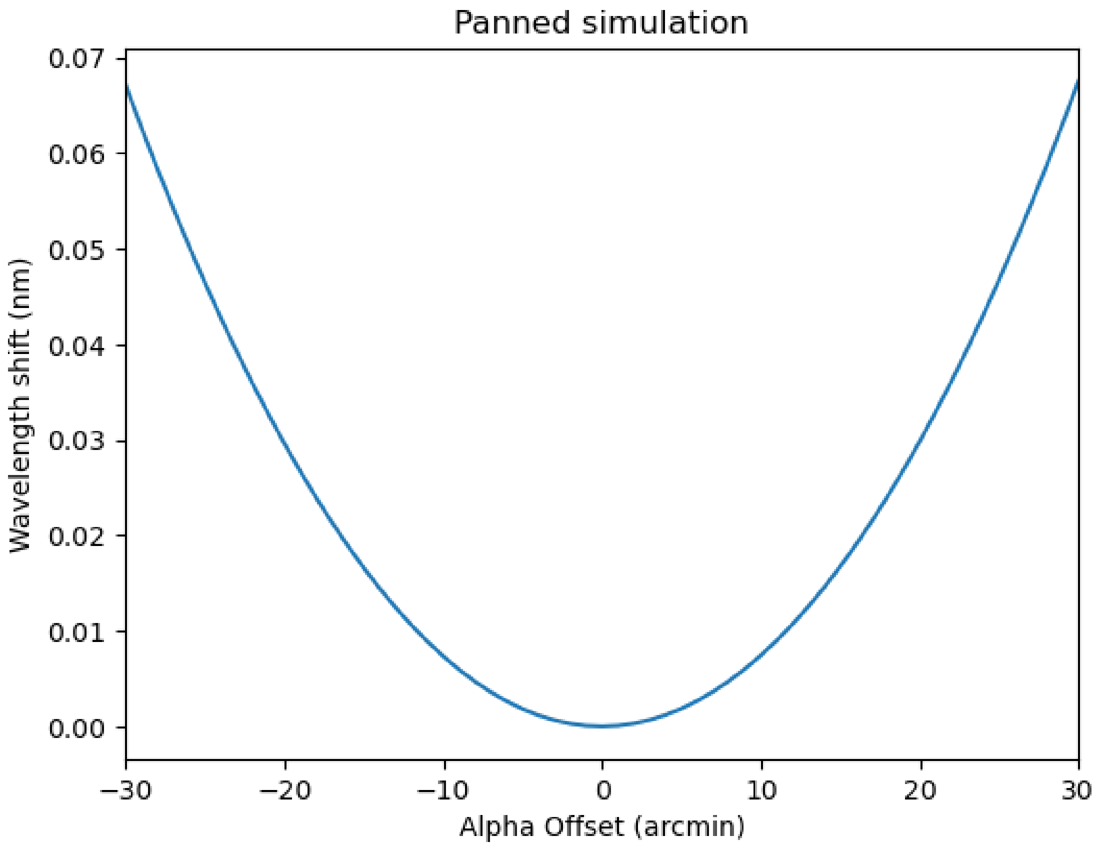
[168]:
```
# Central wavelength shift
fig, ax = plt.subplots()
ax.plot(offaxis_angle_x_min_alpha, wavelength_shift_alpha)
ax.set_xlabel("Alpha Offset (arcmin)")
ax.set_ylabel('Wavelength shift (nm)')
ax.set_xlim(-30,30)
ax.set_title("Unpanned simulation")
```

[168]: Text(0.5, 1.0, 'Unpanned simulation')



[169]:
```
# Panned central wavelength shift
# Y        He II      SDO
#    = 0   = 0
#

fig, ax = plt.subplots()
ax.plot(offaxis_angle_x_min_alpha, wavelength_shift_alpha -
        wavelength_shift_alpha[int(angle_point_num_alpha/2)]) #
ax.set_ylabel('Wavelength shift (nm)')
ax.set_xlabel("Alpha Offset (arcmin)")
ax.set_title("Panned simulation")
ax.set_xlim(-30,30)
```

[169]: (-30.0, 30.0)

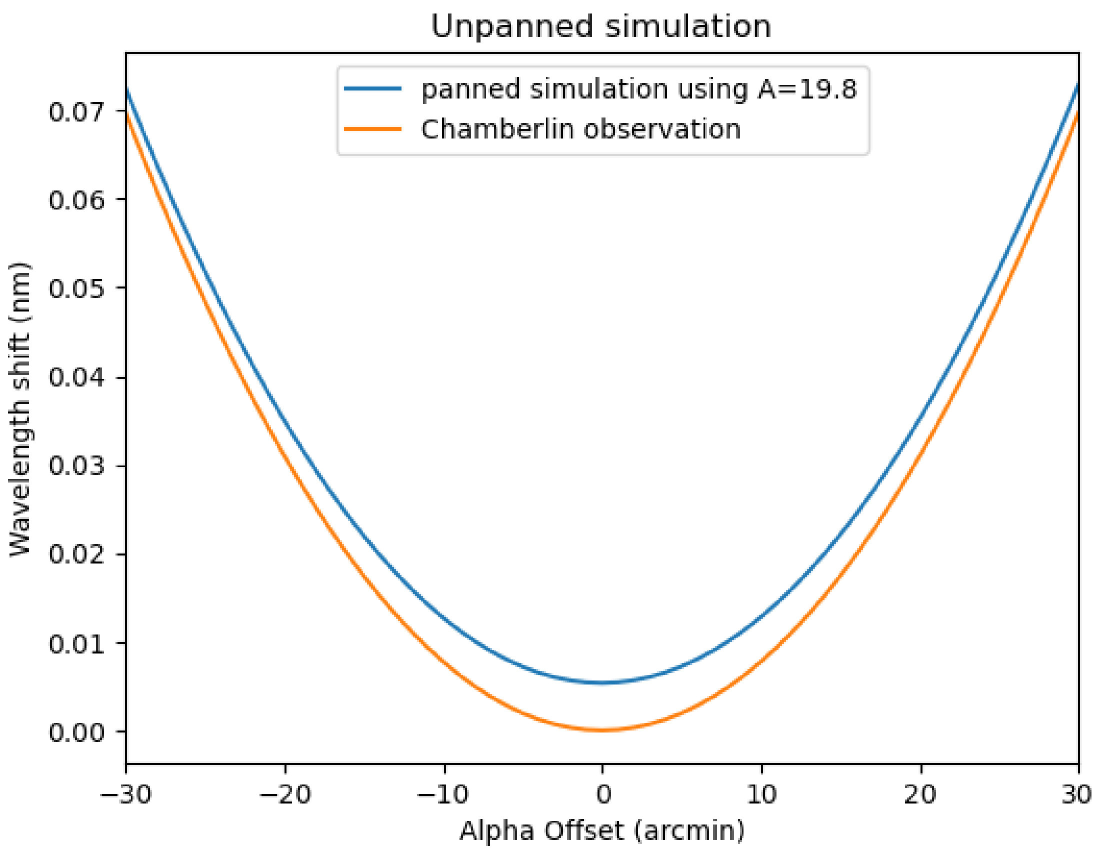


[170]:
```
wavelength_shift_alpha -wavelength_shift_alpha[int(angle_point_num_alpha/2)]
```

[170]:
```
array([6.71449916e-02, 6.26944443e-02, 5.83997358e-02, 5.42605860e-02,
       5.02766885e-02, 4.64477239e-02, 4.27733566e-02, 3.92532338e-02,
       3.58869867e-02, 3.26742322e-02, 2.96145754e-02, 2.67076128e-02,
       2.39529328e-02, 2.13501298e-02, 1.88987924e-02, 1.65985173e-02,
       1.44489099e-02, 1.24495880e-02, 1.06001853e-02, 8.90035415e-03,
       7.34976846e-03, 5.94812548e-03, 4.69514946e-03, 3.59059195e-03,
       2.63423417e-03, 1.82588801e-03, 1.16539705e-03, 6.52637229e-04,
       2.87517440e-04, 6.99798320e-05, 0.00000000e+00, 7.75870155e-05,
       3.02783331e-04, 6.75664558e-04, 1.19633911e-03, 1.86494773e-03,
       2.68166281e-03, 3.64668761e-03, 4.76025523e-03, 6.02262731e-03,
       7.43409265e-03, 8.99496544e-03, 1.07055829e-02, 1.25663033e-02,
       1.45775028e-02, 1.67395732e-02, 1.90529181e-02, 2.15179498e-02,
       2.41350877e-02, 2.69047458e-02, 2.98273432e-02, 3.29032903e-02,
       3.61329884e-02, 3.95168261e-02, 4.30551769e-02, 4.67483970e-02,
       5.05968245e-02, 5.46007809e-02, 5.87605757e-02, 6.30765175e-02,
       6.75489134e-02])
```

[171]:
```
# central wavelength shift
# my panned simulation vs. Chamberlin(2016) observation

fig, ax = plt.subplots()
ax.plot(offaxis_angle_x_min_alpha, wavelength_shift_alpha )
ax.plot(offaxis_angle_x_min_alpha,915.53*offaxis_angle_x_alpha**2) ␣
↪#   unit_conversion.py
ax.legend(["panned simulation using A=19.8","Chamberlin observation"])
ax.set_ylabel('Wavelength shift (nm)')
ax.set_xlabel("Alpha Offset (arcmin)")
ax.set_title("Unpanned simulation")
ax.set_xlim(-30,30)
```

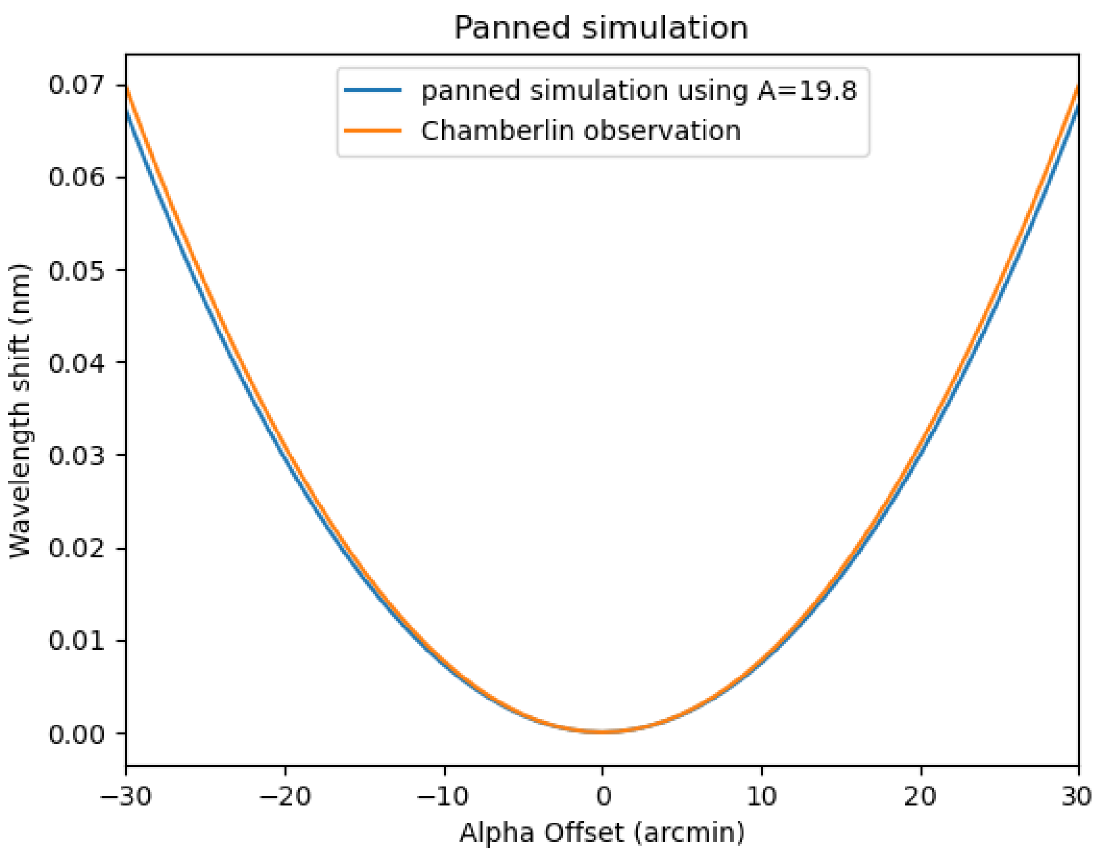
[171]: (-30.0, 30.0)



[172]:
```
# central wavelength shift
# my panned simulation vs. Chamberlin(2016) observation

fig, ax = plt.subplots()
ax.plot(offaxis_angle_x_min_alpha, wavelength_shift_alpha -
```

```
        wavelength_shift_alpha[int(angle_point_num_alpha/2)])
ax.plot(offaxis_angle_x_min_alpha,915.53*offaxis_angle_x_alpha**2) ␣
↪#   unit_conversion.py
ax.legend(["panned simulation using A=19.8","Chamberlin observation"])
ax.set_ylabel('Wavelength shift (nm)')
ax.set_xlabel("Alpha Offset (arcmin)")
ax.set_title("Panned simulation")
ax.set_xlim(-30,30)
```

[172]: (-30.0, 30.0)



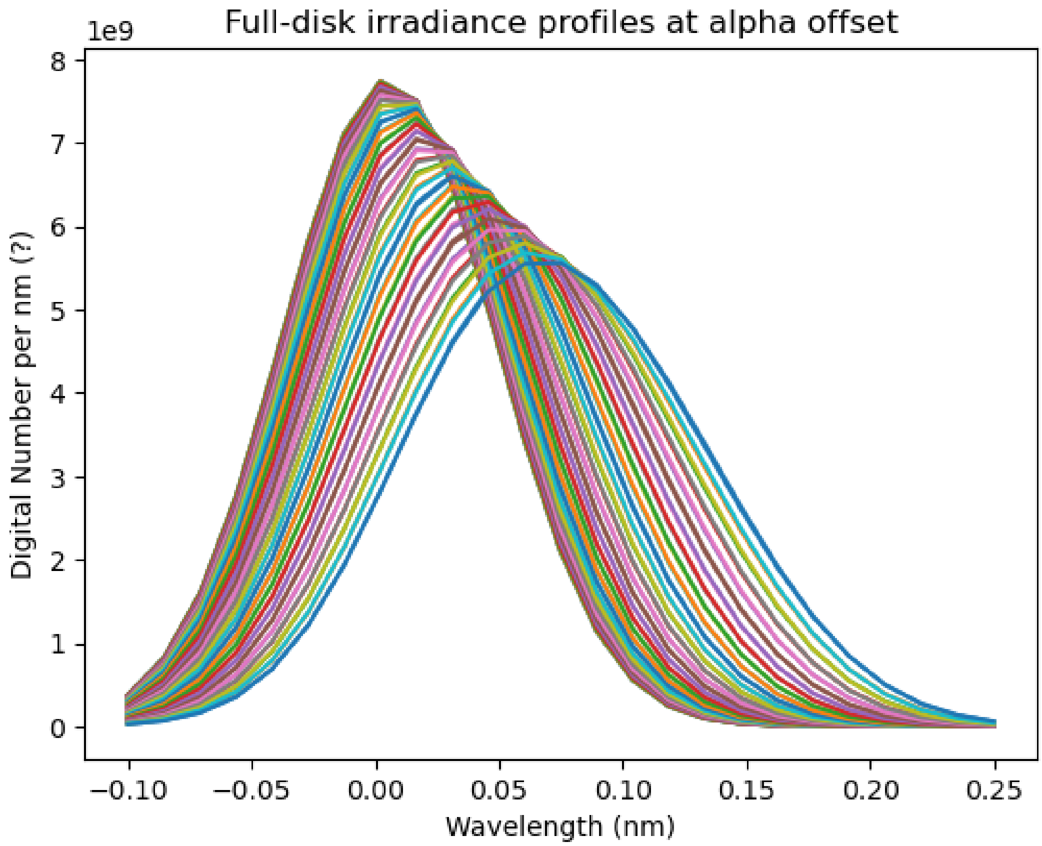
1.2 4096

[173]:
```
import numpy as np
import matplotlib.pyplot as plt
from astropy.modeling import models, fitting
import math
from calculating_DN_2048 import wavelength_point_num, wavelength_list
from calculating_DN_2048 import angle_point_num_alpha,offaxis_angle_x_alpha,␣
↪offaxis_angle_y_alpha
```

```python
from calculating_DN_2048 import angle_point_num_beta,offaxis_angle_x_beta,␣
↪offaxis_angle_y_beta

DN = np.load("output_DN/_4096/DN__4096.npz")

# Initialize
# Cruciformscan in alpha direction
offaxis_angle_x_min_alpha=offaxis_angle_x_alpha*180*60/math.pi

# Cruciformscan in beta direction
offaxis_angle_y_min_beta=offaxis_angle_y_beta*180*60/math.pi

# Fit data in DN??.npz
wavelength_shift_alpha = np.zeros(angle_point_num_alpha)
fit_alpha = []    # List of Gaussian1D
for i in range(angle_point_num_alpha):
    g_init = models.Gaussian1D(amplitude=1E9, mean=0.05, stddev=0.0424)
    # initial value for fitting
    fit_g = fitting.LevMarLSQFitter()
    g = fit_g(g_init, wavelength_list, DN['DN_alpha'][i])
    wavelength_shift_alpha[i] = g.mean.value
    fit_alpha.append(g)
```

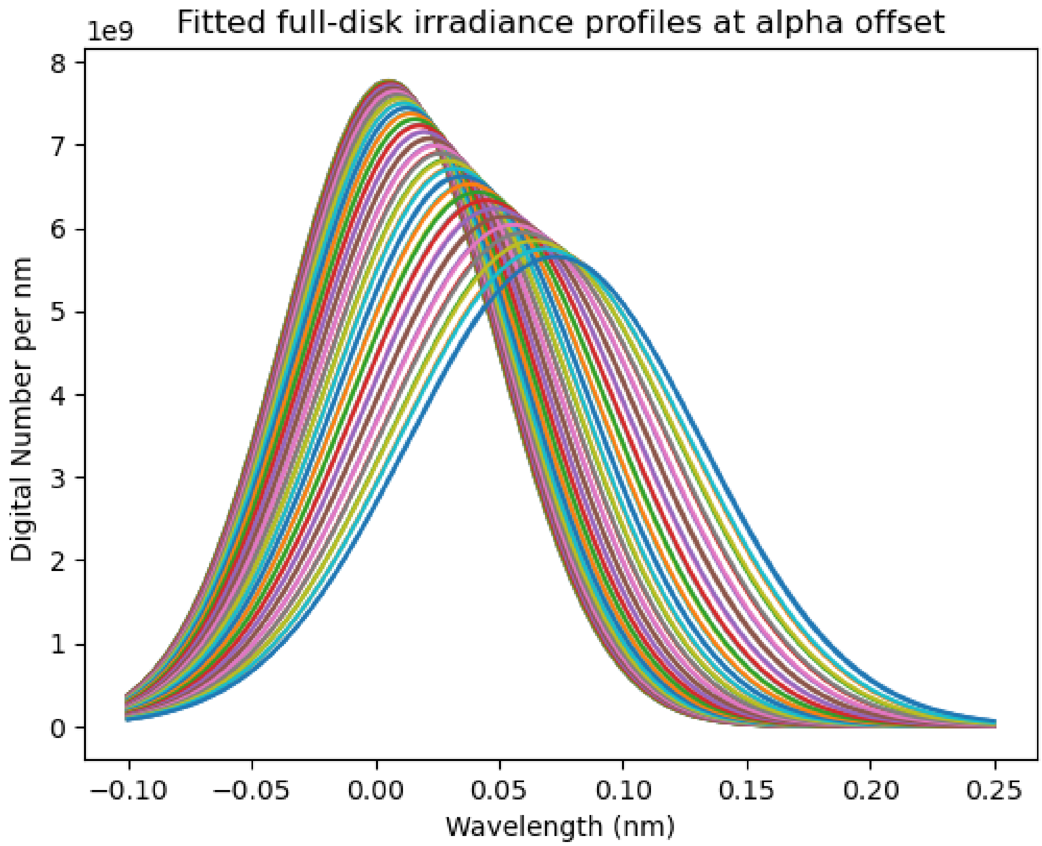
[174]:
```python
# Profiles during cruciformscan
fig, ax = plt.subplots()
# Profiles during cruciformscan in alpha direction
for i in range(angle_point_num_alpha):
    ax.plot(wavelength_list, DN['DN_alpha'][i], label='linear')
# ax.set_title("He II           ")
ax.set_title("Full-disk irradiance profiles at alpha offset")
ax.set_xlabel('Wavelength (nm)')
ax.set_ylabel("Digital Number per nm (?)")
```

[174]: Text(0, 0.5, 'Digital Number per nm (?)')



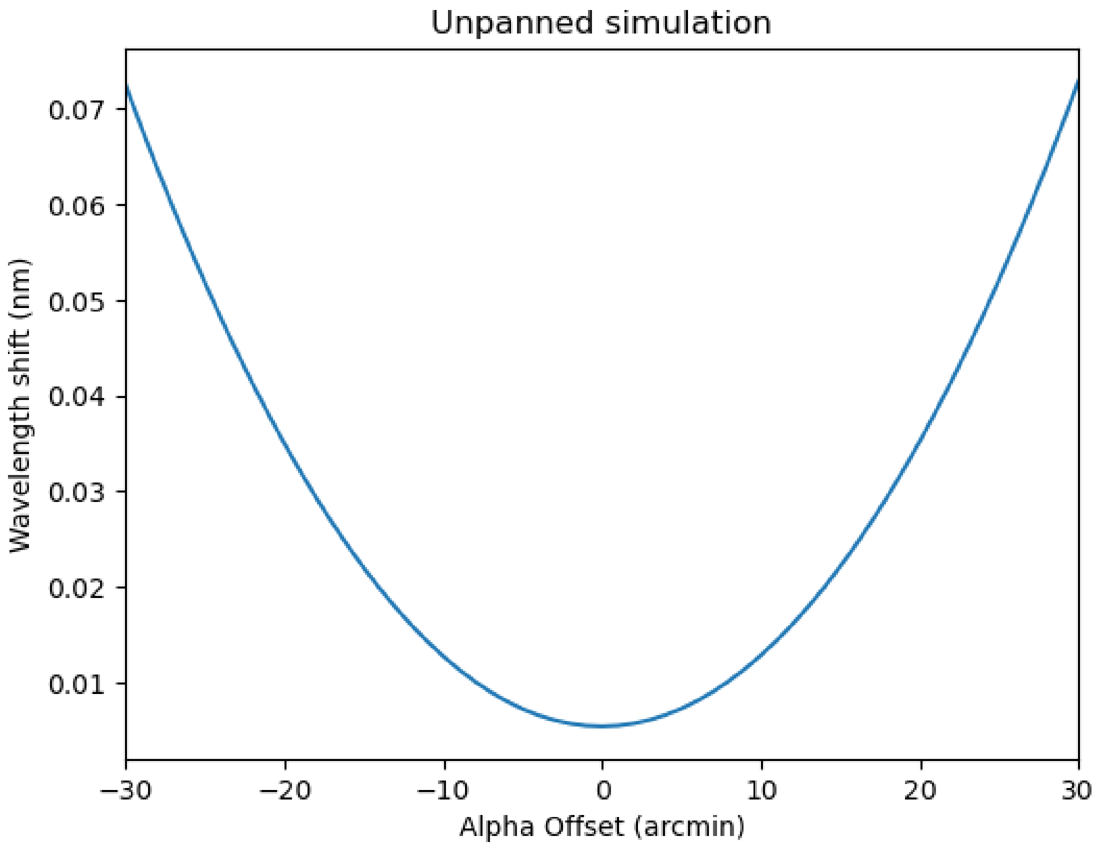
[175]:
```
# Fitted profiles during cruciformscan
fig, ax = plt.subplots()
wavelength_densed_list = np.linspace(-0.1, 0.25, 10*wavelength_point_num)
# Profiles during cruciformscan in alpha direction
for i in range(angle_point_num_alpha):
    ax.plot(wavelength_densed_list, fit_alpha[i](
        wavelength_densed_list), label='linear')

ax.set_xlabel('Wavelength (nm)')
ax.set_ylabel("Digital Number per nm ")
ax.set_title("Fitted full-disk irradiance profiles at alpha offset")
```

[175]: Text(0.5, 1.0, 'Fitted full-disk irradiance profiles at alpha offset')



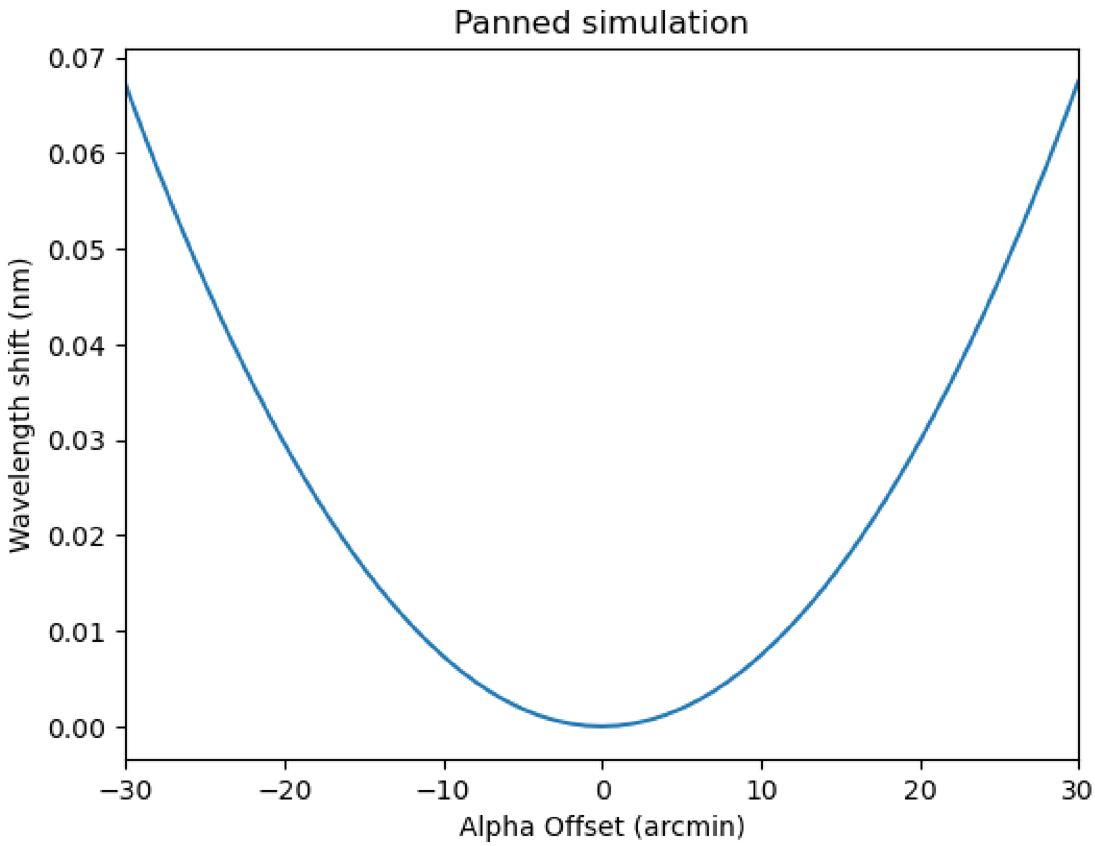
[176]:
```python
# Central wavelength shift
fig, ax = plt.subplots()
ax.plot(offaxis_angle_x_min_alpha, wavelength_shift_alpha)
ax.set_xlabel("Alpha Offset (arcmin)")
ax.set_ylabel('Wavelength shift (nm)')
ax.set_title("Unpanned simulation")
ax.set_xlim(-30,30)
```

[176]: (-30.0, 30.0)



[177]:
```
# Panned central wavelength shift
# Y        He II      SDO
#    = 0   = 0
#

fig, ax = plt.subplots()
ax.plot(offaxis_angle_x_min_alpha, wavelength_shift_alpha -
        wavelength_shift_alpha[int(angle_point_num_alpha/2)]) #
ax.set_ylabel('Wavelength shift (nm)')
ax.set_xlabel("Alpha Offset (arcmin)")
ax.set_title("Panned simulation")
ax.set_xlim(-30,30)
```

[177]: (-30.0, 30.0)



1.2.1 2048 4096

2048

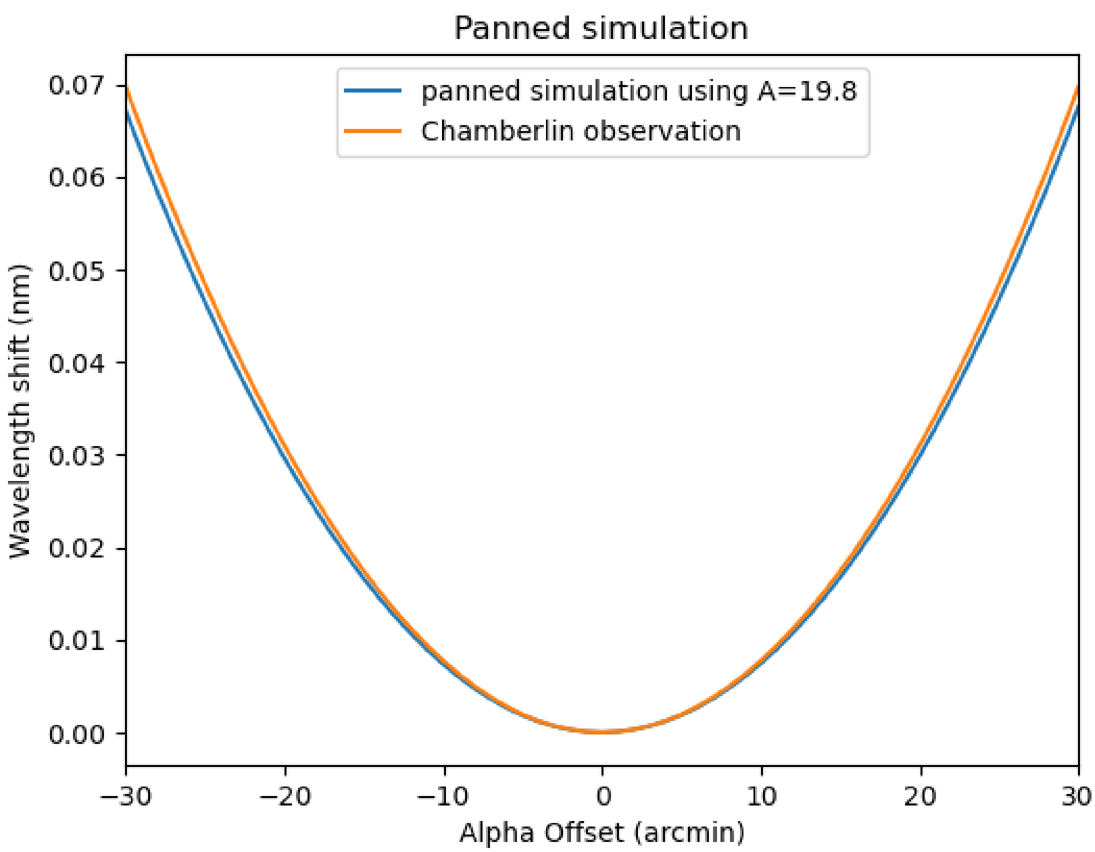
[178]:
```
#  4096
wavelength_shift_alpha -wavelength_shift_alpha[int(angle_point_num_alpha/2)]
```

[178]:
```
array([6.71446497e-02, 6.26940976e-02, 5.83993925e-02, 5.42602548e-02,
       5.02763663e-02, 4.64474030e-02, 4.27730511e-02, 3.92529384e-02,
       3.58866986e-02, 3.26739536e-02, 2.96143080e-02, 2.67073562e-02,
       2.39526897e-02, 2.13498996e-02, 1.88985763e-02, 1.65983132e-02,
       1.44487198e-02, 1.24494116e-02, 1.06000227e-02, 8.90020577e-03,
       7.34963421e-03, 5.94800568e-03, 4.69504400e-03, 3.59050061e-03,
       2.63415688e-03, 1.82582456e-03, 1.16534711e-03, 6.52600460e-04,
       2.87493416e-04, 6.99680818e-05, 0.00000000e+00, 7.75981969e-05,
       3.02805083e-04, 6.75696232e-04, 1.19638003e-03, 1.86499718e-03,
       2.68172006e-03, 3.64675193e-03, 4.76032592e-03, 6.02270355e-03,
       7.43417375e-03, 8.99505055e-03, 1.07056713e-02, 1.25663943e-02,
       1.45775959e-02, 1.67396671e-02, 1.90530132e-02, 2.15180440e-02,
       2.41351784e-02, 2.69048378e-02, 2.98274314e-02, 3.29033761e-02,
```

```
3.61330717e-02, 3.95169079e-02, 4.30552541e-02, 4.67484636e-02,
5.05968917e-02, 5.46008528e-02, 5.87606248e-02, 6.30765648e-02,
6.75489664e-02])
```

[179]:
```python
# central wavelength shift
# my panned simulation vs. Chamberlin(2016) observation

fig, ax = plt.subplots()
ax.plot(offaxis_angle_x_min_alpha, wavelength_shift_alpha -
        wavelength_shift_alpha[int(angle_point_num_alpha/2)])
ax.plot(offaxis_angle_x_min_alpha,915.53*offaxis_angle_x_alpha**2) ␣
 ↪#   unit_conversion.py
ax.legend(["panned simulation using A=19.8","Chamberlin observation"])
ax.set_ylabel('Wavelength shift (nm)')
ax.set_title("Panned simulation")
ax.set_xlabel("Alpha Offset (arcmin)")
ax.set_xlim(-30,30)
```

[179]: (-30.0, 30.0)



1.3

P46

P46 3.6 He II

P50 “ A ” :

P50 “ A ”

1. $\Delta\lambda_0 < \bar{\Delta}\lambda$
2.
3. P46

```
[ ]:
```