

RMSComparison

```
# -*- coding: utf-8 -*-
```

```
"""
```

Created on Tue Apr 2 12:50:20 2019

@author: irg16

Description: This module is used to plot the RMS comparison graph for this specific lens.

```
"""
```

```
import RayTracer as RT
import matplotlib.pyplot as plt
import numpy as np
import matplotlib.patches as mpatches
"""
```

Plotting a graph to compare RMS for the single spherical refracting surface.

This code uses all the values set for the single spherical refracting surface in BundleTest.

This is an independent set of code simply used to plot the RMS comparison graph for this specific lens.

```
"""
```

```
"""
```

Components for: single spherical refracting surface

```
"""
```

```
sphere1=RT.SphericalRefraction(100, -0.03, 1., 1.5, 10) #define the lens
plane1=RT.SphericalRefraction(101, 0, 1.5, 1.5, 10) #nonexistent plane
focal=199.9997999996499 #the focal point as calculated in BundleTest.py for sphere1
```

```
"""
```

```
"""
```

```
"""
```

Components for: plano-convex singlet lens with the plane surface facing the input

```
sphere1=RT.SphericalRefraction(45, 0.02, 1.5168, 1., 10) #define the lens
plane1=RT.SphericalRefraction(40, 0, 1., 1.5168, 10) #define the plane
focal=194.33259736520344 #the focal point as calculated in BundleTest.py for sphere1
```

```
"""
```

```
"""
```

Components for: plano-convex singlet lens with the convex surface facing the input

```
sphere1=RT.SphericalRefraction(40, -0.02, 1., 1.5168, 10) #define the lens
plane1=RT.SphericalRefraction(45, 0, 1.5168, 1, 10) #define the plane
focal=138.4527001775158 #the focal point as calculated in BundleTest.py for sphere1
```

```
"""
```

RMSComparison

```
output1=RT.OutputPlane(focal)

for i in np.arange(1, 6, 0.01):
    buni=RT.RayBundle([0,0,0],[0,0,1],i,10)
    buni.FireBundle(sphere1) #propagate the bundle through the lens
    buni.FireBundle(plane1) #propagate the bundle through the plane
    buni.FireBundle(output1) #propagate the bundle to the output plane
    RMS = buni.RMS() #calculate the RMS for the bundle
    #Calculate the diffraction scale and the effect on the RMS
    wavelen = 0.000588 #set the wavelength of the light rays - this is 588nm
    beamD = buni._br*2
    if sphere1._z0 < plane1._z0:
        focallen = focal - sphere1._z0
    elif sphere1._z0 > plane1._z0:
        focallen = focal - plane1._z0
    diffscale = wavelen*focallen/beamD
    plt.plot(i,RMS, 'bo')
    plt.plot(i,diffscale, 'ro')

plt.title('Finding the Diffraction Limit of the Beam Radius')
red_patch = mpatches.Patch(color='red', label='Diffraction Scale')
blue_patch = mpatches.Patch(color='blue', label='RMS (paraxial approximation)')
plt.legend(handles=[red_patch, blue_patch])
plt.xlabel('Ray Bundle Radius /mm')
plt.ylabel('Size /mm')
```