How We Predict the Shapes of Donuts in the WFS devices

Bo Xin

3/13/20

Notes

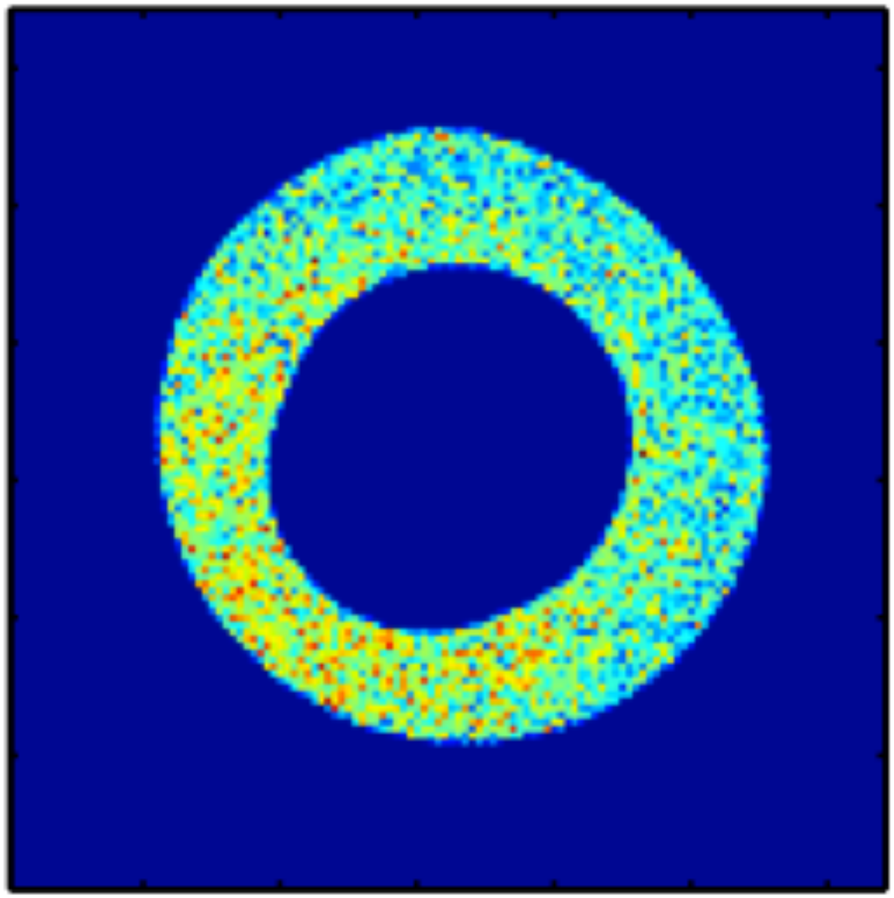
1. The boundary of the donuts depend on the aberrations that are present, for example, Zernikes with n=|m| only change the boundary of the donut without altering intensity uniformity.

2. What we can model is, for ideal optics, how the shape of the donuts vary with field position.

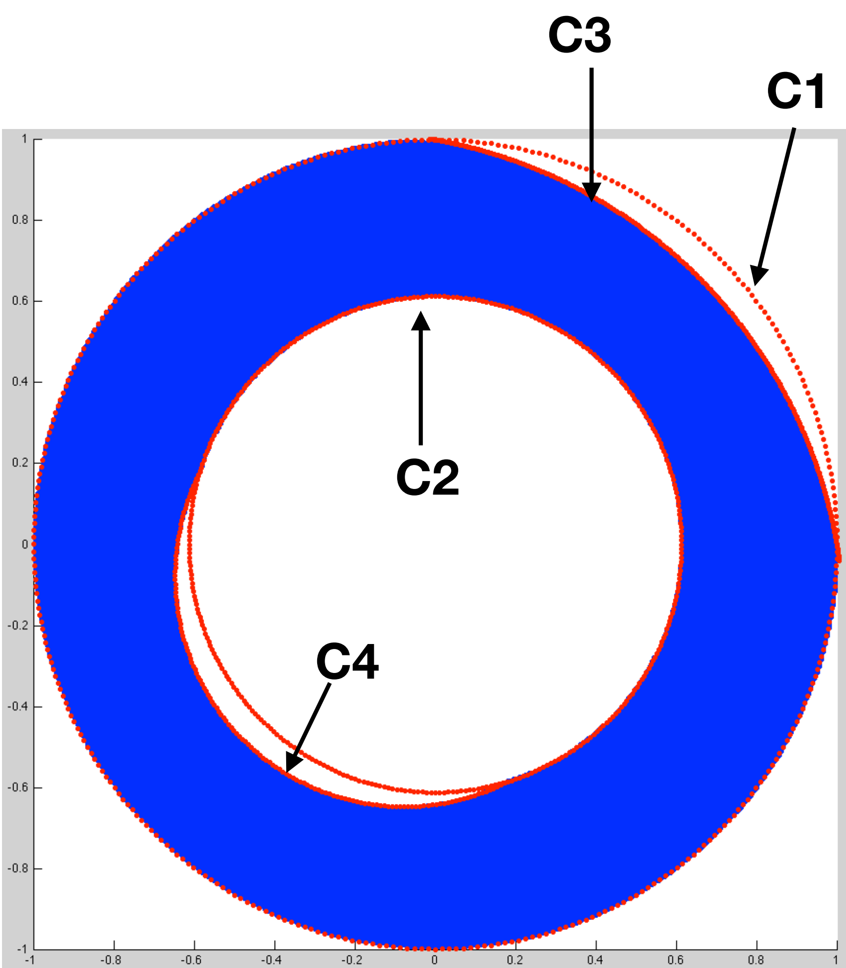
3. We model the donut shape under ideal optics at various field radius using ZEMAX LSST model (no 2D grid is needed since the optical system is axisymmetric), then interpolate to any field radius/position.

4. The pupil mask is derived as described above. The computational mask is obtained by enlarging the pupil mask by a few pixels (via eroding). Given #1 above, the computational mask is only valid when the aberration is small so that no intensity signal is thrown away. Having an iterative algorithm for CWFS solves the problem.

A typical donut on WFS



Binary Image



This shape can be carved out using 4 circles: C1, C2, C3, C4

- C1 is a unit-radius circle, centered at (0,0)
- C2 has radius = obscuration ratio, centered at (0,0)
- C3 and C4 are for characterizing the vignetting.

4 parameters are needed to model this shape:

1. Ca: center position of C3. We only need the distance from donut (0,0), because it it's always on the line connecting donut (0,0) to center of focal plane.
2. Ra: radius of C3.
3. Cb: center position of C4.
4. Rb: radius of C4

Code Detail

- In my CWFS code, this is how we create the mask
https://github.com/bxin/cwfs/blob/d4a889d52aa15c3b57349d7d5863f7bbaf850cf9/python/lsst/cwfs/image.py#L103

- This is how I create the list which is used as input to the above
https://github.com/bxin/cwfs/blob/d4a889d52aa15c3b57349d7d5863f7bbaf850cf9/python/lsst/cwfs/image.py#L87

- You can see how Ca, Cb, Ra, Rb are being used.

- Te-Wei can easily point out where to find the corresponding parts in ts_wep