The Journal of Open Source Software

AnisoCADO: a python package for analytically generating adaptive optics point spread functions for the Extremely Large Telescope

Kieran Leschinski[1] **and Eric Gendron**[2]

1 Department of Astrophysics, University of Vienna **2** Observatoire de Paris

DOI: 10.21105/joss.0XXXX

Software

- Review
- Repository
- Archive

Editor: Editor Name

Submitted: 01 January 1900
Published: 01 January 3030

Summary

AnisoCADO is a Python package for generating images of the point spread function (PSF) for the european extremely large telescope (ELT). The code allows the user to set a large range of the most important atmospheric and observational parameters that influence the shape and strehl ratio of the resulting PSF, including but not limited to: the atmospheric turbulence profile, the guide star position for a single conjugate adaptive optics (SCAO) solution, differential telescope pupil transmission, etc. Documentation can be found at https://anisocado.readthedocs.io/en/latest/

Statement of need

Adaptive optics are mandatory for the next generation of ground-based telescopes

The larger the telescope aperture, the smaller the diffraction limit of the observations. For space-based telescope this statement is always true. However the resolution of ground based telescopes is limited by the blur caused by turbulence in the atmosphere - known as atmospheric Seeing. This blurring can be (mostly) removed by measuring the deformation of the wavefront of the incoming light, and applying an equal and opposite deformation to the surface of one or more of the mirrors along a telescope's optical path. The current fleet of large (8-10m) telescopes were built to primarily operate at the edge of the natural seeing limit (FWHM~0.5 arcseconds @ 1um). Over the last two decades some have received upgrades in the form of active and adaptive mirrors in order to achieve up to 20x increase in resolution afforded by the physical diffraction limit of a 10m primary mirror (FWHM~0.03 arcseconds @ 1um). The next generation of "extremely large" telescopes will have primary mirrors on the order of 30-40m, with theoretical diffraciton limits on the order of 50x smaller than the natural Seeing limit. In order for these telescopes to resolve structures at scales of the diffraction limit, they must, by design, include adaptive optics systems.

Diffraction limited point-spread-functions are complex beasts

The point spread function (PSF) of an optical system is the description of the spatial distribution of light from an infinitely small point source after passing through an optical system (e.g. layers of the atmosphere, mirrors of a telescope). Due to the random nature of atmospheric turbulence, the PSF of a star in a Seeing-limited observation is well approximated by a ("nice") smooth Gaussian-like function. The PSF of a diffraction limited telescope system using an adaptive-optics correction is a complex ("ugly") function that depends on a veritible zoo of atmospheric, observational, and technical parameters. From an astronomers point of view, the consequences of a poor adaptive optics solution means the difference between a successful and a failed observation run. Therfore it is imperative that the consequences of such large variations in the PSF are accounted for in advance by those proposing to observe with this next generation of billion-dollar telescopes.

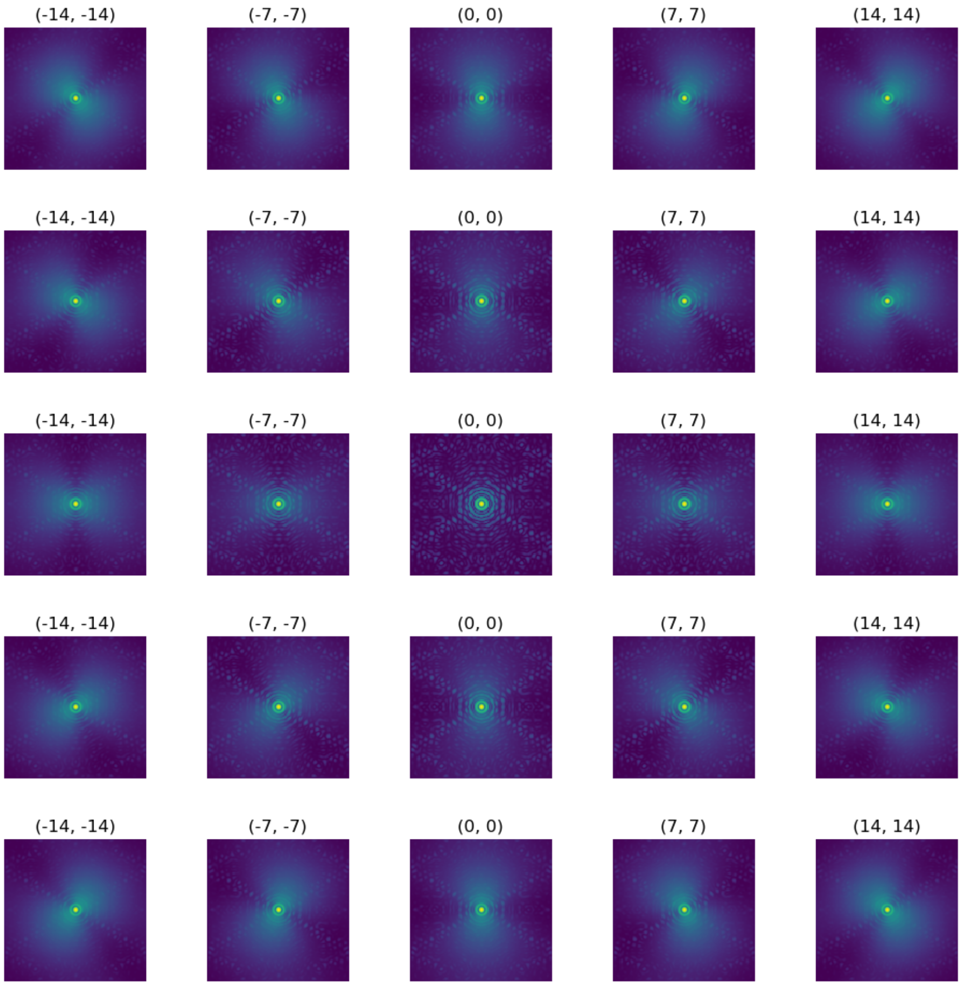

Figure 1: A grid of Ks-band (2.15um) PSFs for a range of distances from the natural guide star. The PSFs were generated using the ESO median turbulence profile.

AnisoCADO - Anisoplanatism for MICADO

AnisoCADO (Anisoplanatism for MICADO) is a package for generating images of the point spread function for a given set of atmospheric, observational, and technical conditions. It does this by combining a series of wavefront phase screens from the elements of the atmosphere and AO system that influence the final AO correction (e.g. atmospheric anisoplantism, WFS aliasing, actuator fit, etc). The final phase screen is applied to the optical transfer function for the telescope optical system. The resulting image is the expected PSF for a long exposure (>10s) on-axis observation at the given wavelength. For single-conjugate adaptive optics modes, the field PSF degrades as distance from the guide star increases. This effect is taken

into account by shifting the anisoplanatic phase screen relative to the calculated phase screen correction for the deformable mirror. Figure 1 shows how the PSF changes with distance from an on-axis guide star.
For a more detailed discussion of the mathematics behind anisoplanatism in the context of the ELT, the reader is referred to Clénet, Gendron, Gratadour, Rousset, & Vidal (2015).

Inputs

The final ELT PSF is the combination of many factors. The vast majority of these are irrelevant for the casual user. AnisoCADO therefore provides three preset option, corresponding to the standard ESO Q1, Median and Q4 turbulence profiles. All other parameters are initialised with default values. For the case of a SCAO system (for which AnisoCADO was originally conceived) PSFs can be generated for multiple guide star offsets without needing to re-make all phase screens by using the special class method `.shift_off_axis(dx, dy)`

For more detailed use cases, the following parameters are available to the user:

Atmosphere	Observation	Telescope
turbulence profile height of turbulent layers stregth of turbulent layers	natural guide star position central wavelength pupil rotation angle	pupil image 2D pupil transmissivity dead/empty mirror segments
wind speed Seeing FWHM @ 500nm Fried parameter outer scale	Zenith distance of observation	plate scale residual wavefront errors AO sampling frequency AO loop delay Interactuator distance

Outputs

AnisoCADO is easily integrated into the standard astronomers toolbox. PSF images generated by AnisoCADO can be output as either `numpy` arrays, or standard `astropy.io.fits.ImageHDU` objects. The latter can be written to file using the standard `astropy` syntax.

As AnisoCADO was written to support the development of the MICADO instrument simulator (Leschinski et al., 2016, 2019), it is also possible to generate `FieldVaryingPSF` objects using the helper functions in the `misc` module. Such files are also compatible with the generic instrument data simulator framework, ScopeSim.

Basic Example

The AnisoCADO API is described in the online documentation, which can be found at: https://anisocado.readthedocs.io/. For the purpose of illustration, the following 5 lines were used to generate the grid of PSFs in Figure 1.

```
import numpy as np
from anisocado import AnalyticalScaoPsf

psf = AnalyticalScaoPsf()
psf_grid = []
for x, y in np.mgrid[-14:15:7, -14:15:7].flatten().reshape((2, 25)).T:
    psf.shift_off_axis(x, y)
    psf_grid += [psf.kernel]
```

Acknowledgments

AnisoCADO depends on the following packages: Numpy (van der Walt, Colbert, & Varoquaux, 2011), Matplotlib (van der Walt et al., 2011), Astropy (Astropy Collaboration et al., 2018).

This development of this project was funded by the project IS538004 of the Hochschulraumstrukturmittel (HRSM) provided by the Austrian Government and administered by the University of Vienna.

References

Astropy Collaboration, Price-Whelan, A. M., Sipőcz, B. M., Günther, H. M., Lim, P. L., Crawford, S. M., Conseil, S., et al. (2018). The Astropy Project: Building an Open-science Project and Status of the v2.0 Core Package, *156*(3), 123. doi:10.3847/1538-3881/aabc4f

Clénet, Y., Gendron, E., Gratadour, D., Rousset, G., & Vidal, F. (2015). Anisoplanatism effect on the E-ELT SCAO point spread function. A preserved coherent core across the field, *583*, A102. doi:10.1051/0004-6361/201425469

Leschinski, K., Czoske, O., Köhler, R., Mach, M., Zeilinger, W., Verdoes Kleijn, G., Alves, J., et al. (2016). SimCADO: an instrument data simulator package for MICADO at the E-ELT. In Society of photo-optical instrumentation engineers (spie) conference series (Vol. 9911, p. 991124). doi:10.1117/12.2232483

Leschinski, K., Czoske, O., Köhler, R., Mach, M., Zeilinger, W., Verdoes Kleijn, G., Kausch, W., et al. (2019). SimCADO - a Python Package for Simulating Detector Output for MICADO at the E-ELT. In M. Molinaro, K. Shortridge, & F. Pasian (Eds.), *Astronomical data analysis software and systems xxvi*, Astronomical society of the pacific conference series (Vol. 521, p. 527).

van der Walt, S., Colbert, S. C., & Varoquaux, G. (2011). The numpy array: A structure for efficient numerical computation. *Computing in Science and Engineering*, *13*(2), 22–30.