

TipTop into ESO tools current status

C.F. Manara (USD), TipTop Team, O. Hainaut (E2E Operations)

T. Bierwirth, V. Forchi, M. Petr-Gotzens (ObsPrep)

J. Vinther, Y. Jung, H. Boffin (ETC2.0), M. van den Ancker (USD)

A. Kuznetsov, E. Congiu, M. Vioque, A. Concas (ESO Fellows)



TIPTOP TEAM



TipTop: toward a single tool for all ELT instrument's PSF prediction

Benoit Neichel^a, Guido Agapito^b, Arseniy Kuznetsov^{a,c,d}, Fabio Rossi^b, Cédric Plantet^b, Carlo F. Manara^d, Romain Fétick^c, Alice Concas^d, Joël Vernet^d, Olivier Hainaut^d, Anne-Laure Cheffot^b, Giulia Carlà^b, Jean-François Sauvage^{a,c}, Michele Cirasuolo^d, Paolo Padovani^d, Anne-Laure Cheffot^b, Carlos Correia^e, Cedric Taissir Hérítier^c, and Thierry Fusco^c

^aAix Marseille Univ, CNRS, CNES, LAM, Marseille, France

^bINAF - Osservatorio Astrofisico di Arcetri Largo E. Fermi 5, 50125 Firenze Italy

^cEuropean Southern Observatory, Karl-Schwarzschild-str-2, 85748 Garching, Germany

^dONERA, B.P. 72, F-92322 Chatillon, France

^eSpace ODT Lda, Av da França 492, 4050-277 Porto, Portugal

SPIE 2024 paper



TIPTOP AT ESO



MOVING TOWARDS THE ELT

WE (ESO) NEED TO *COOPERATE* WITH THE TIPTOP TEAM TOWARDS
AN OPERATION-READY MICROSERVICE &
PRACTICE THE TYPE OF **INTERFACE** WE WANT TO PUT ON TIPTOP,
& *PREPARE* FOR **INTEGRATION** WITH P1/P2 AND ETC

WE WILL USE VLT INSTRUMENTS AS **DEMONSTRATORS**
WHILE GETTING SOME **BONUS FEATURES** ALREADY NOW

PSF AO calculations at ESO

- In the **ETC**, to ensure that the S/N calculated uses a realistic PSF,
- At **Phase 1**, to demonstrate that the required resolution and (via ETC) the required S/N can be achieved, and to help selecting the best possible AO asterism for a given field.
- At **Phase 2**, to fine-tune the AO reference star selection,
- Possibly for **Quality Control**, to verify whether the observed PSF matches the expectations,
- Possibly for **data processing**: if the estimated PSF is reliable enough, it could be used to deconvolve data.

ESO-492366



EXAMPLE OF TipTop USAGE

Input: ini (or yaml) file

```

1  [[telescope]
2  TelescopeDiameter = 8.
3  ZenithAngle = 30.0
4  ObscurationRatio = 0.16
5  Resolution = 128
6  TechnicalFoV = 120
7
8  [atmosphere]
9  Wavelength = 500e-9
10 Seeing = 1.0
11 L0 = 22.0
12 Cn2Weights = [0.59, 0.02, 0.04, 0.06, 0.01, 0.05, 0.09, 0.04, 0.05, 0.05]
13 Cn2Heights = [30, 140, 281, 562, 1125, 2250, 4500, 7750, 11000, 14000]
14 WindSpeed = [6.6, 5.9, 5.1, 4.5, 5.1, 8.3, 16.3, 10.2, 14.3, 17.5]
15 WindDirection = [0., 0., 0., 0., 90., -90., -90., 90., 0., 0.]
16 r0_Value = 0
17 testWindspeed = 0
18
19 [sources_science]
20 ; set the wavelength to K-band 2.21 micron
21 Wavelength = [2.21e-06]
22 ; set source on-axes
23 Zenith = [0.0]
24 Azimuth = [0.0]
25
26 [sources_H0]
27 Wavelength = 589e-9
28 Zenith = [0.0]
29 Azimuth = [0.0]
30 Height = 90000
31
32 [sources_L0]
33 Wavelength = 750e-09
34 Zenith = [0.0]
35 Azimuth = [0.0]
36
37 [sensor_science]
38 PixelScale = 14
39 FieldOfView = 256
40
41 [sensor_H0]
42 WfsType = 'Shack-Hartmann'
43 Modulation = None
44 PixelScale = 832
45 FieldOfView = 6
46 Binning = 1
47 ; nph/frame/sub-aperture
48 NumberPhotons = [1000.0]
49

```

Telescope information

Atmospheric conditions

Targets (wl, position)

High-order GS (Laser)

Low-order (NGS)

Sensor science

Sensor High-Order

FIXED

(only zenithangle varies)

From HDR/telemetry
Or TC category (WIP)

OB info

(filter, target location)

OB info

(LGS/NGS location)

Sensor High-Order

Sensor Low-Order

Telescope information

```

38 PixelScale = 14
39 FieldOfView = 256
40
41 [sensor_H0]
42 WfsType = 'Shack-Hartmann'
43 Modulation = None
44 PixelScale = 832
45 FieldOfView = 6
46 Binning = 1
47 ; nph/frame/sub-aperture
48 NumberPhotons = [1000.0]
49 SigmaRON = 0.2
50 ExcessNoiseFactor = 2.0
51 NumberLenslets = [40]
52 NoiseVariance = [None]
53 Algorithm = 'wcog'
54
55 [sensor_L0]
56 WfsType = 'Shack-Hartmann'
57 Modulation = None
58 PixelScale = 417
59 FieldOfView = 48
60 Binning = 1
61 ; nph/frame
62 NumberPhotons = [100.0]
63 SigmaRON = 0.2
64 Dark = 0.0
65 SkyBackground = 0.0
66 ExcessNoiseFactor = 2.0
67 NumberLenslets = [40]
68 NoiseVariance = [None]
69 WindowRadiusWCoG = 6
70 ThresholdWCoG = 0.0
71 NewValueThrPix = 0.0
72
73 [DM]
74 NumberActuators = [40]
75 DmPitches = [0.2]
76 InfModel = 'gaussian'
77 InfCoupling = [0.0]
78 DmHeights = [0.0]
79 OptimizationZenith = [0]
80 OptimizationAzimuth = [0]
81 OptimizationWeight = [1]
82 OptimizationConditioning = 1.0e2
83 NumberReconstructedLayers = 3
84 AoArea = 'circle'
85
86 [RTC]
87 LoopGain_H0 = 0.3
88 SensorFrameRate_H0 = 1000.0
89 LoopDelaySteps_H0 = 3

```

EXAMPLE OF TipTop USAGE – Asterism selection

Input: ini (or yaml) file

```

1  [[telescope]
2  TelescopeDiameter = 8.
3  ZenithAngle = 30.0
4  ObscurationRatio = 0.16
5  Resolution = 128
6  TechnicalFoV = 120
7
8  [atmosphere]
9  Wavelength = 500e-9
10 Seeing = 1.0
11 L0 = 22.0
12 Cn2Weights = [0.59, 0.02, 0.04, 0.06, 0.01, 0.05,
13 Cn2Heights = [30, 140, 281, 562, 1125, 2250, 4500,
14 WindSpeed = [6.6, 5.9, 5.1, 4.5, 5.1, 8.3, 16.3, 1
15 WindDirection = [0., 0., 0., 0., 90., -90., -90.,
16 r0_Value = 0
17 testWindspeed = 0
18
19 [sources_science]
20 ; set the wavelength to K-band 2.21 micron
21 Wavelength = [2.21e-06]
22 ; set source on-axes
23 Zenith = [0.0]
24 Azimuth = [0.0]
25
26 [sources_H0]
27 Wavelength = 589e-9
28 Zenith = [0.0]
29 Azimuth = [0.0]
30 Height = 90000
31
32 [sources_L0]
33 Wavelength = 750e-09
34 Zenith = [0.0]
35 Azimuth = [0.0]
36
37 [sensor_science]
38 PixelScale = 14
39 FieldOfView = 256
40
41 [sensor_H0]
42 WfsType = 'Shack-Hartmann'
43 Modulation = None
44 PixelScale = 832
45 FieldOfView = 6
46 Binning = 1
47 ; nph/frame/sub-aperture
48 NumberPhotons = [1000.0]
49

```

An additional section of the TipTop config file (.ini or yaml) provides the data needed for the computation

- o For each asterism star we have [azimuth, zenith, flux]

```

[ASTERISM_SELECTION]
mode = 'Sets'
Zenith = [[5], [20], [5], [7], [22], [17], [20], [17] ]
Azimuth = [[0.0], [45], [90], [60.0], [145], [190], [145], [90] ]
NumberPhotons = [[900], [21000], [190], [5100], [10800], [1800], [210], [180000]]

Example 2
[ASTERISM_SELECTION]
mode = 'Sets'
Zenith = [[10, 40, 30], [15, 45, 35], [10, 40, 35]]
Azimuth = [[0.0, 45, 90], [60.0, 145, 190], [0.0, 45, 190] ]
NumberPhotons = [[1900, 2100, 1900], [1100, 1800, 1800], [1900, 2100, 1800]]

Example 3
[ASTERISM_SELECTION]
mode = 'Singles1'
Zenith = [60.0, 40, 10, 50, 30, 20, 10]
Azimuth = [0.0, 45, 95, 135, 190, 242, 177]
NumberPhotons = [1900, 1800, 700, 2000, 200, 1110, 400]
Frequencies = [250, 250, 500, 250, 500, 100, 100, 100]

```

Sensor High-Order

Sensor Low-Order

Telescope information

```

86 [RTC]
87 LoopGain_H0 = 0.3
88 SensorFrameRate_H0 = 1000.0
89 LoopDelaySteps_H0 = 3

```

ESO MICROSERVICE

The ESO microservices can invoke:

- *OverallSimulation* (1D or 2D PSF)
- *AsterismSelection*
- future (?): *GenerateHeuristic*

It runs the current stable* TipTop version on a GPU server.

It returns a FITS/JSON file or some parameters/arrays.

It can interface with ETC or P2 (retrieveing OB properties etc)

Implementation by T. Bierwirth & V. Forchi

*stable = currently **tested** in all modes in operations

Microservice Invocation

The microservice is *prototypically* installed at <https://www.eso.org/p2services/any/tiptop>

It can be invoked using a simple Python script

```
import sys
import requests
import email
import email.policy
import json

instrument = sys.argv[1]
url = 'https://www.eso.org/p2services/any/tiptop'
files = {
    'serviceDescription': ('serviceDescription.json', open('serviceDescription.json', 'rb'), 'application/json'),
    'parameterFile': (instrument + '.ini', open(instrument + '.ini', 'rb'), 'text/plain')
}

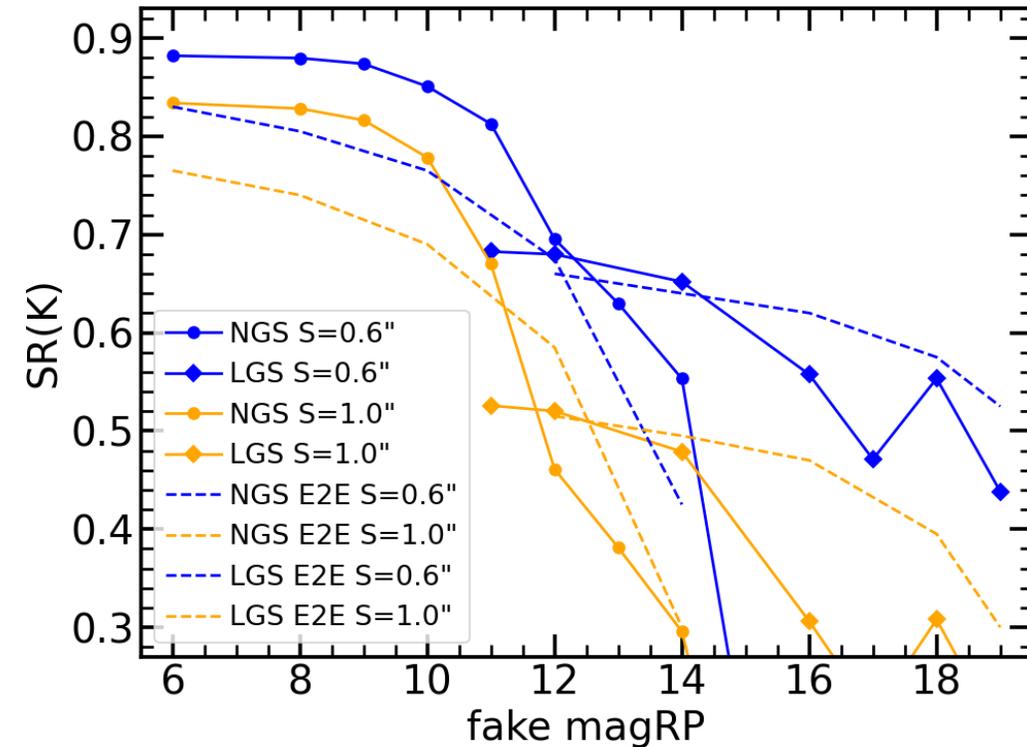
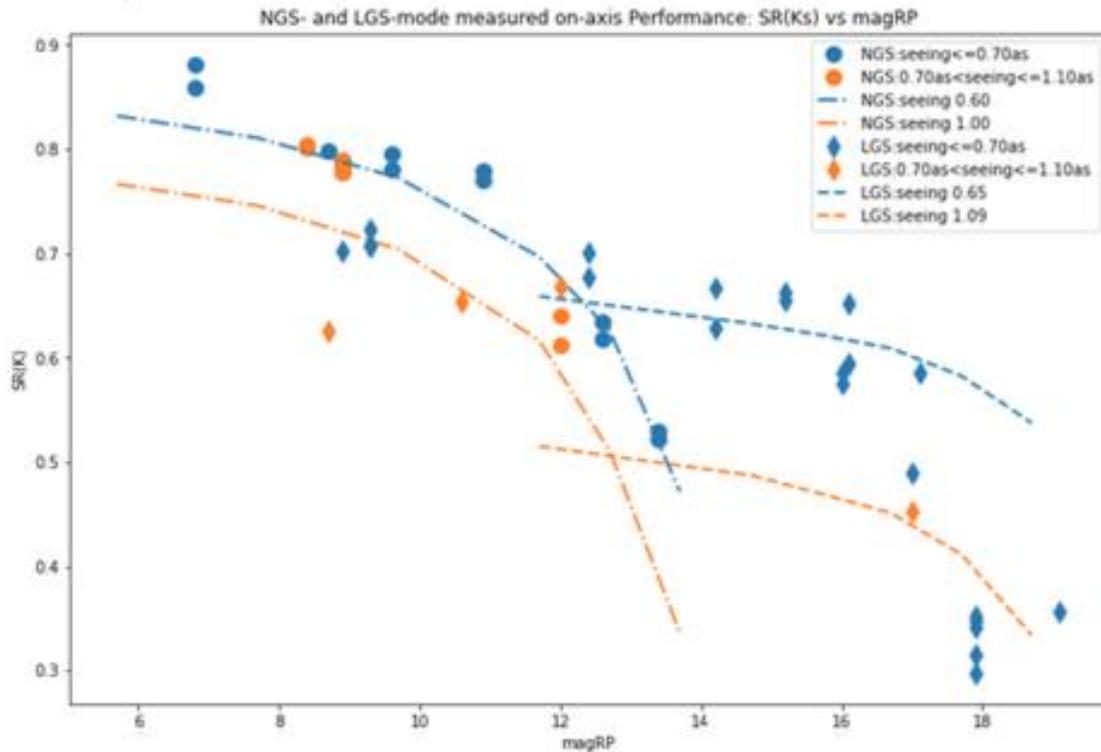
response = requests.post(url, files=files)
if response.status_code == 200:
    # prepend multipart header
    msg = ('Content-type: ' + response.headers['Content-Type'] + '\n\n').encode() + response.content
    multipart = email.message_from_bytes(msg, policy=email.policy.HTTP)
    if (multipart.is_multipart()):
        for part in multipart.walk():
            fname = part.get_filename()
            if fname and fname.endswith('.fits'):
                with open(fname, 'wb') as f:
                    f.write(part.get_content())
                    print('created %s' % fname)
            elif fname == 'tiptop.log':
                print(part.get_content())
            elif part.get_content_type() == 'application/json':
                serviceResult = json.loads(part.get_content())
                print(serviceResult['service']['message'])
    else:
        print(response.text)
```

The script invokes the microservice API and passes as inputs a `<instrument>.ini` file and an almost empty (for now) `serviceDescription.json`. It produces an `<instrument>.fits` file as output.

ERIS AO SYSTEM – TipTop version

Lots of work by A. Concas,
G. Agapito, F. Rossi,
C.F. Manara

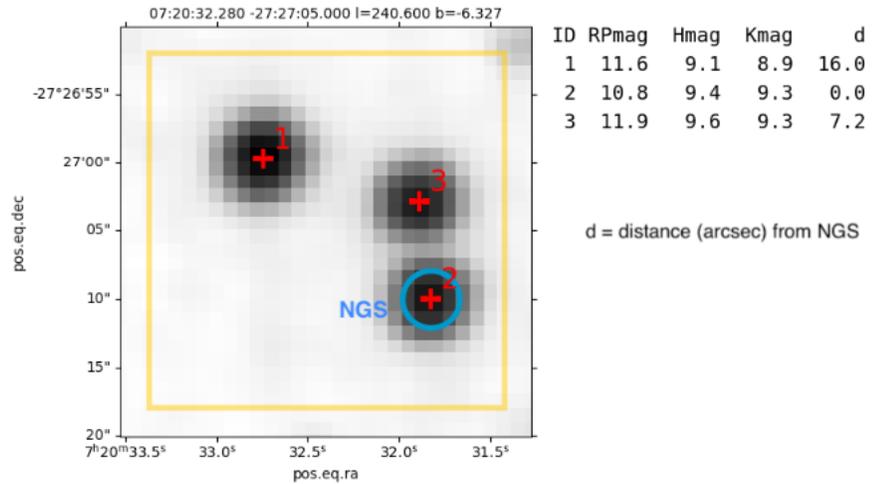
NGS mode provides better correction down to magRP~13.5



TipTop version of ERIS User Manual, Fig 17

ERIS TipTop VALIDATION: PSF SHAPE

Aim: reproducing PSF of observations taken with ERIS using TipTop



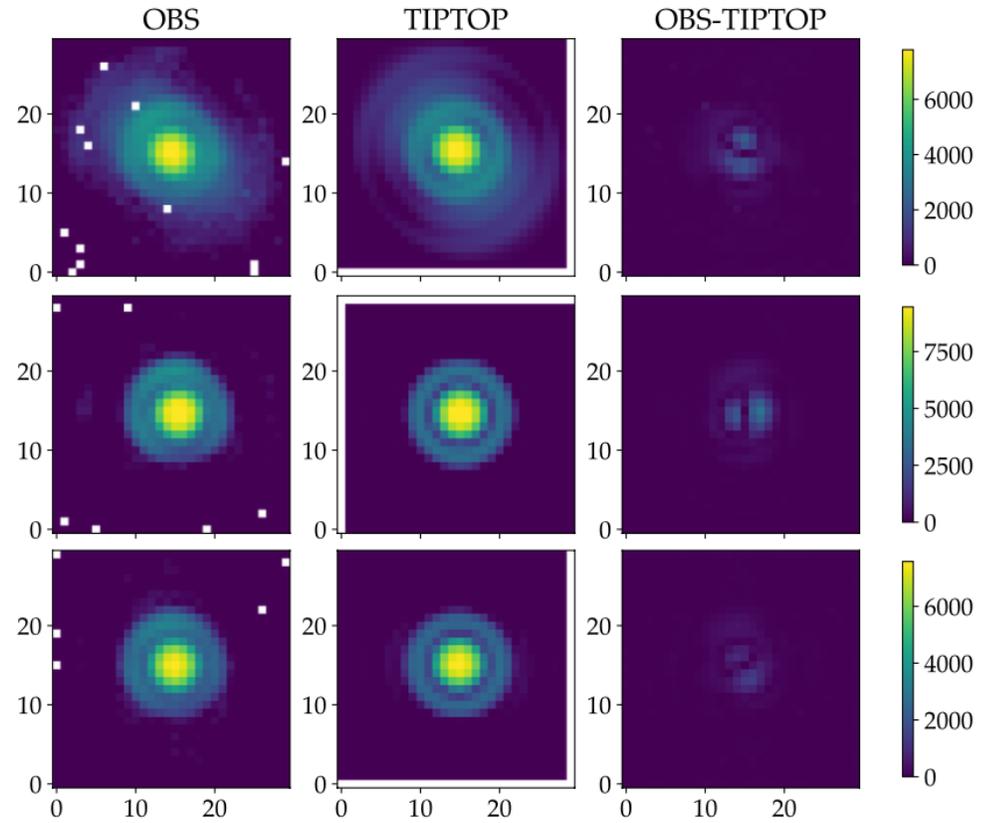
d = distance (arcsec) from NGS

After shift

STAR 1

STAR 2
NGS

STAR 3



**Using median(frames)

Work by A. Concas

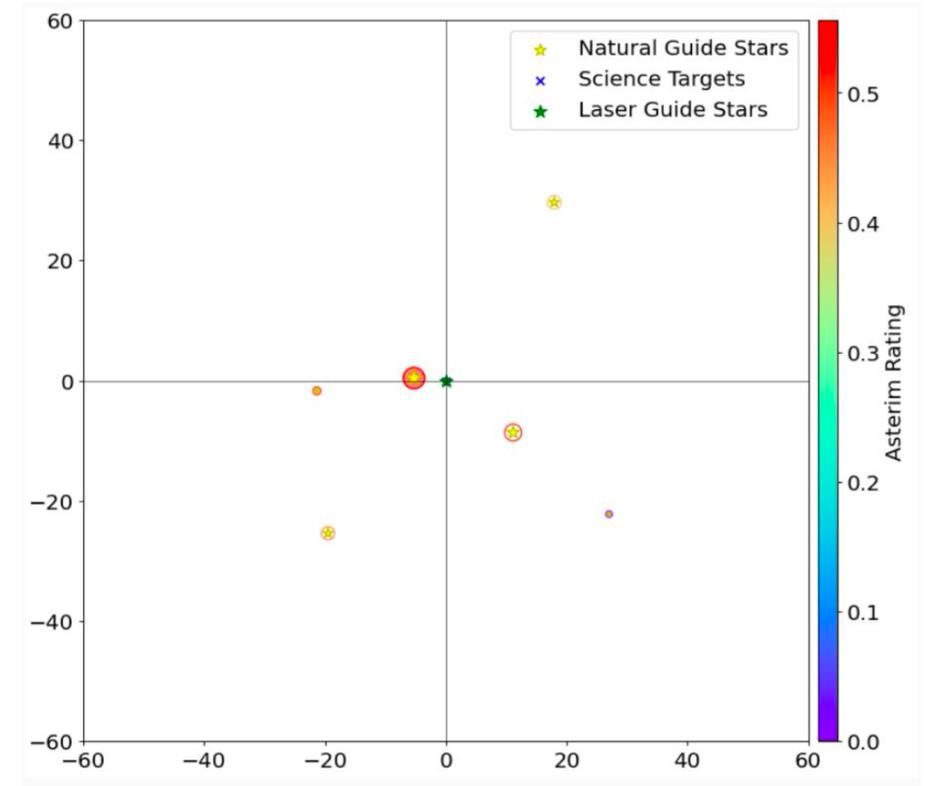
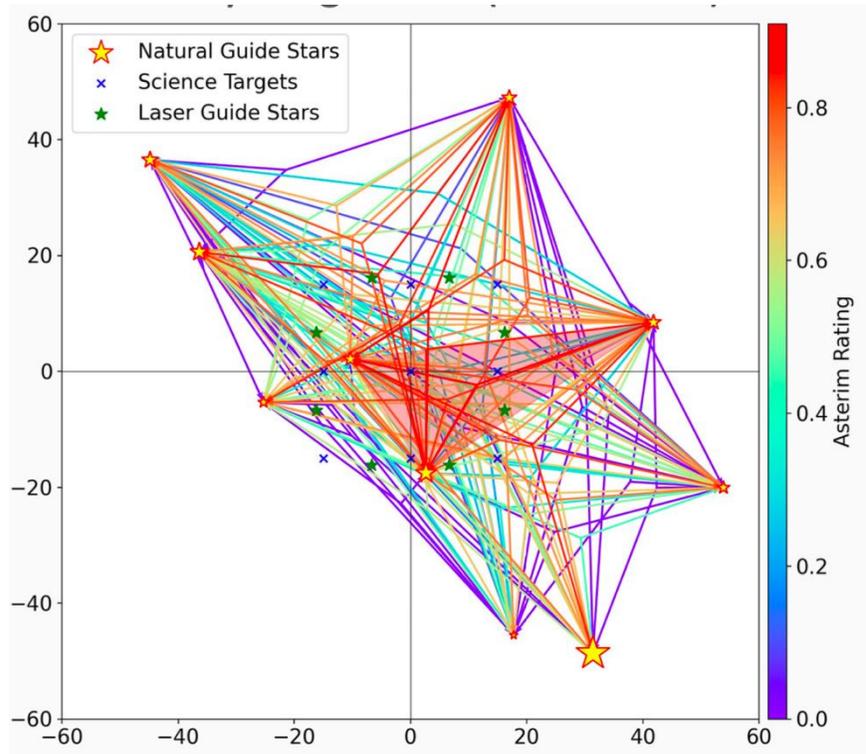
ASTERISM SELECTION IMPLEMENTATION

- For each asterism the sum of the total Tip-Tilt jitter over the science targets positions is computed
- The **lower jitter asterism** is selected as the **best one**
- **Code optimized**, avoid to repeat “per star” computations
- Performances
 - One single asterism in a field $\approx 4s$
 - 120 asterisms from 10 stars field $\approx 60s$ (0.5s/asterism)
 - faster heuristic approach ready for some AO modes (from v. 1.3.12)

ASTERISM SELECTION - examples

Selected asterism as a red triangle, NGSs are scaled by magnitude (MAVIS like).

Selected asterism as a red circle, NGSs are scaled by magnitude (ERIS like).



ERIS ObsPrep – TipTop implementation

- Python server by V. Forchi
- Script/wrapper to write ini file on the fly taking information from OB

```

47
30
31 def get_zenith(self, center: SkyCoord) -> float:
32     return self.truncate(center.separation(self.coord).arcsec)
33
34 def get_azimuth(self, center: SkyCoord) -> float:
35     return self.truncate(center.position_angle(self.coord).deg + 90)
36
37 def get_framerate_lgs(self) -> int:
38     if self.mag <= 16:
39         return 500
40     elif self.mag <= 17:
41         return 250
42     else:
43         return 100
44
45 def get_number_photons_lgs(self) -> float:
46     f0_NGS_LGS_ph = 1.51e10
47     m0_NGS_LGS = 0
48     nph_sec = f0_NGS_LGS_ph * 10**(-(self.mag-m0_NGS_LGS)/2.5)
49     return self.truncate(nph_sec / self.get_framerate_lgs())
50
51 def get_extra_error_nm_lgs(self):
52     # data from AD document VLT-TRE-ERI-14403-3001 Sect. 4.6.9
53     # assuming linear relation with magnitude
54     return 145. + (145.-180.)/(12.-19.)*(self.mag-12.)
55
56 def to_dict(self):
57     return {
58         "ra": self.ra,
59         "dec": self.dec,
60         "mag": self.mag
61     }
62
63 def get_center(target) -> SkyCoord:
64     # TODO: apply proper motion correction
65     return SkyCoord(ra=target['ra'], dec=target['dec'], units(u.hourangle, u.deg), frame='icrs')
66
67 def get_url_and_token(adminConfig):
68     url = adminConfig['url']
69     token = adminConfig['authToken']
70     if url == None:
71         raise Exception('Missing parameter admin.url')
72     if token == None:
73         raise Exception('Missing parameter admin.token')
74

```

eris_wrapper | tiptop / eris_wrapper.py

extract result from simulation
Vincenzo Forchi authored 1 week ago

eris_wrapper.py | 7.57 KIB

```

1 import sys, argparse, json, p2api, os
2 import numpy as np
3
4 from astropy.coordinates import SkyCoord
5 import astropy.units as u
6
7 from configparser import ConfigParser
8 from dataclasses import dataclass, field
9 from typing import List
10
11 orig_stdout = sys.stdout
12 sys.stdout = open('tiptop.log', 'w')
13
14 from tiptop.tiptop import *
15
16 instrument = 'eris'
17
18 @dataclass
19 class Star:
20     ra: float
21     dec: float
22     mag: float
23     coord: SkyCoord = field(init=False)
24
25 def __post_init__(self):
26     self.coord = SkyCoord(ra=self.ra * u.deg, dec=self.dec * u.deg, frame='icrs')
27
28 def truncate(self, number):
29     return int(number * 1e6) / 1e6
30
31 def get_zenith(self, center: SkyCoord) -> float:
32     return self.truncate(center.separation(self.coord).arcsec)
33
34 def get_azimuth(self, center: SkyCoord) -> float:

```



ERIS ObsPrep – TipTop implementation

Target: triple system TYC 6545-127-1

(TTR target)

Several possible TTS, not clear whether best to select the brightest in the SW of the FoV, or a slightly fainter one closer to the center of the FoV.

- How good is the Strehl ratio at the target's coordinates depending on the selected TTS?
- TIPTOP could be called to select the best TTS to maximise Strehl in a given position (0,0)?
- TIPTOP could automatically show a table of Strehl ratio for every possible TTS

60.A-9252(O) · ERIS · OB 3055089 triple_TipTop_test_2 Exp. Time: 00:00:00 · Exec. Time: 00:00:00 (Partially Defined)

Obs. Description Target Constraint Set Time Intervals Finding Charts Ephemeris Target Visibility ObsPrep

Pointing AO Stars Observing Offsets VLT Guide Star

Select 1 TTS star per template in the list below from the ca...
[Read more](#)

#1 ERIS_nixIMG_acq_LGS

RA	Dec	Gaia RP Mag
----	-----	-------------

Show Strehl

J2000 07 20 32.281 -27 27 5.02

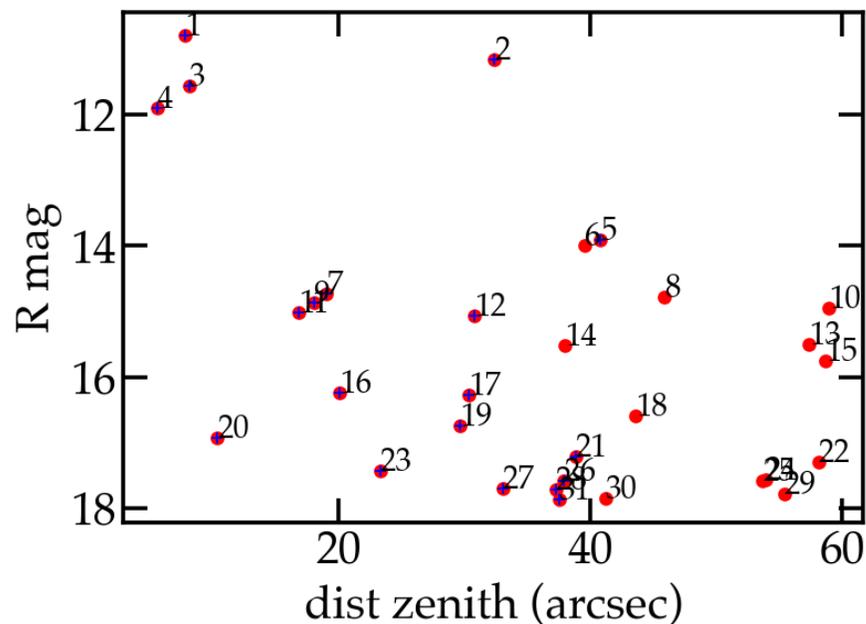


ERIS ObsPrep – TipTop implementation

Target: triple system TYC 6545-127-1

(ra='07:20:32.280',dec='-27:27:05.000')

Several possible TTS:



60.A-9252(O) · ERIS · **OB** 3055089 triple_TipTop_test_2 Exp. Time: 00:00:00 · Exec. Time: 00:00:00 (Partially Defined)

Obs. Description Target Constraint Set Time Intervals Finding Charts Ephemeris Target Visibility **★ ObsPrep**

Pointing **AO Stars** Observing Offsets VLT Guide Star

i Select 1 TTS star per template in the list below from the ca... [Read more](#)

#1 ERIS_nixIMG_acq_LGS

RA	Dec	Gaia RP Mag
----	-----	-------------

Show Strehl

The image shows a field of stars in grayscale. A yellow box highlights a central region containing several bright stars. The stars are of varying sizes and brightness, with some appearing as large, dark, out-of-focus blobs and others as smaller, sharper points. The background is filled with a dense field of fainter stars.



ERIS ObsPrep – TipTop implementation

Target: triple system TYC 6545-127-1

(ra='07:20:32.280',dec='-27:27:05.000')

Several possible TTS:

60.A-9252(O) · ERIS · **OB** 3055089 triple_TipTop_test_2 Exp. Time: 00:00:00 · Exec. Time: 00:00:00 (Partially Defined)

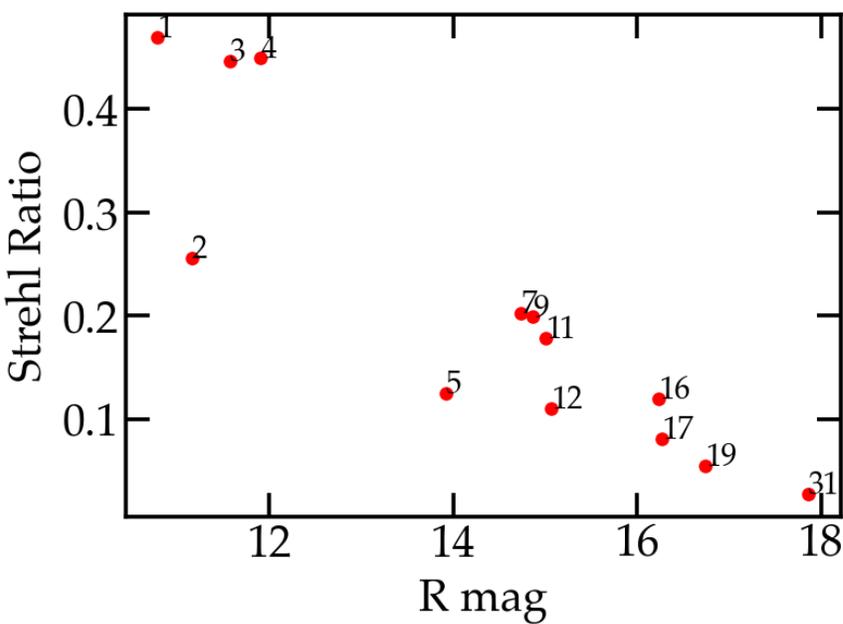
Obs. Description Target Constraint Set Time Intervals Finding Charts Ephemeris Target Visibility **ObsPrep**

Pointing AO Stars Observing Offsets VLT Guide Star

Select 1 TTS star per template in the list below from the ca... [Read more](#)

#1 ERIS_nixIMG_acq_LGS

RA	Dec	Gaia RP Mag
1. RA, DEC, Rmag, SR, FWHM		
4. RA, DEC, Rmag, SR, FWHM		
3. RA, DEC, Rmag, SR, FWHM		
2. RA, DEC, Rmag, SR, FWHM		
7. RA, DEC, Rmag, SR, FWHM		
...		



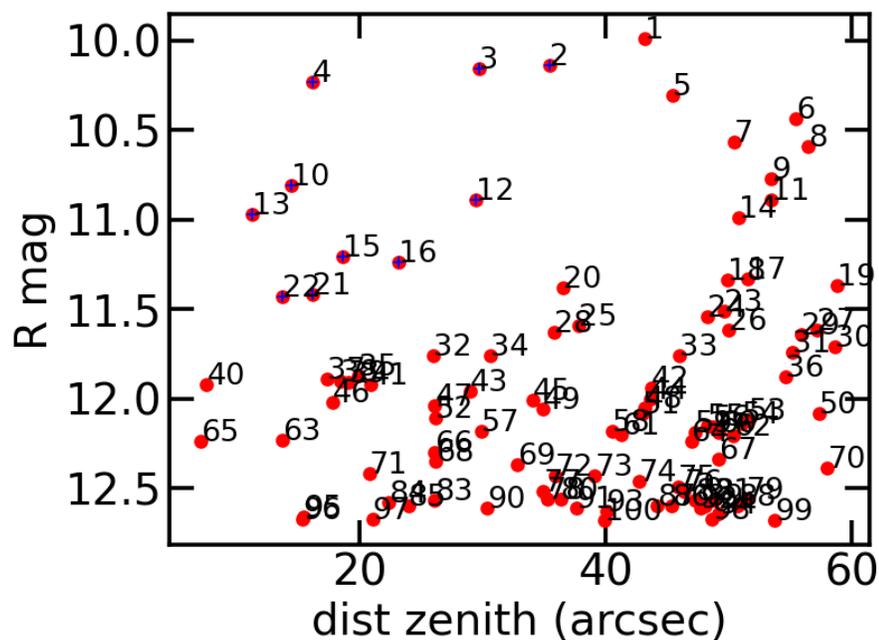


ERIS ObsPrep – TipTop implementation

Target: Omega Cen

(ra='13:26:47.280',dec='-47:28:46.100')

Many possible TTS:



60.A-9252(O) · ERIS · OB 3073257 tiptop_crowded Exp. Time: 00:00:00 · Exec. Time: 00:00:00 (P)artially Defined

Obs. Description Target Constraint Set Time Intervals Ephemeris Target Visibility **★ ObsPrep** Finding Charts

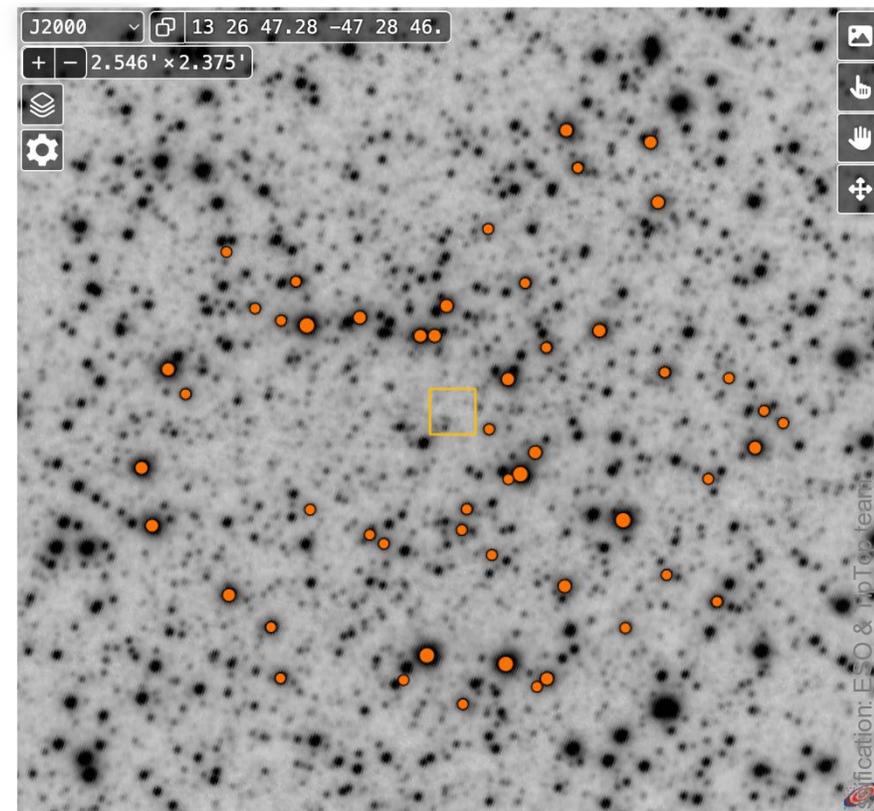
Pointing AO Stars Observing Offsets VLT Guide Star

Select 1 TTS star per template in the list below from the ca... [Read more](#)

#1 ERIS_ifs_acq_LGS

RA Dec Gaia RP Mag

Show Strehl



Data Classification: ESO & TipTop team

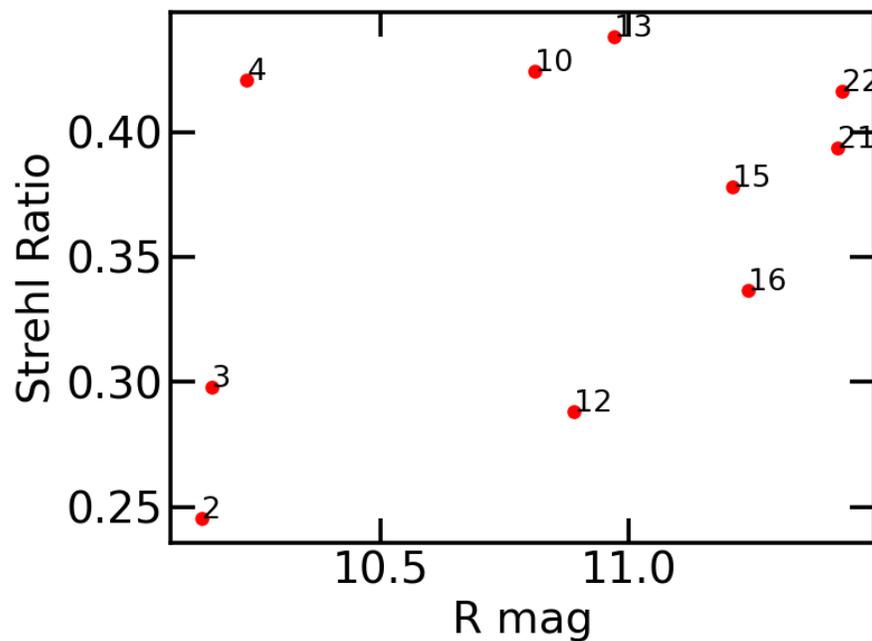


ERIS ObsPrep – TipTop implementation

Target: Omega Cen

(ra='13:26:47.280',dec='-47:28:46.100')

Many possible TTS:



60.A-9252(O) · ERIS · OB 3073257 tiptop_crowded Exp. Time: 00:00:00 · Exec. Time: 00:00:00 (P)artially Defined

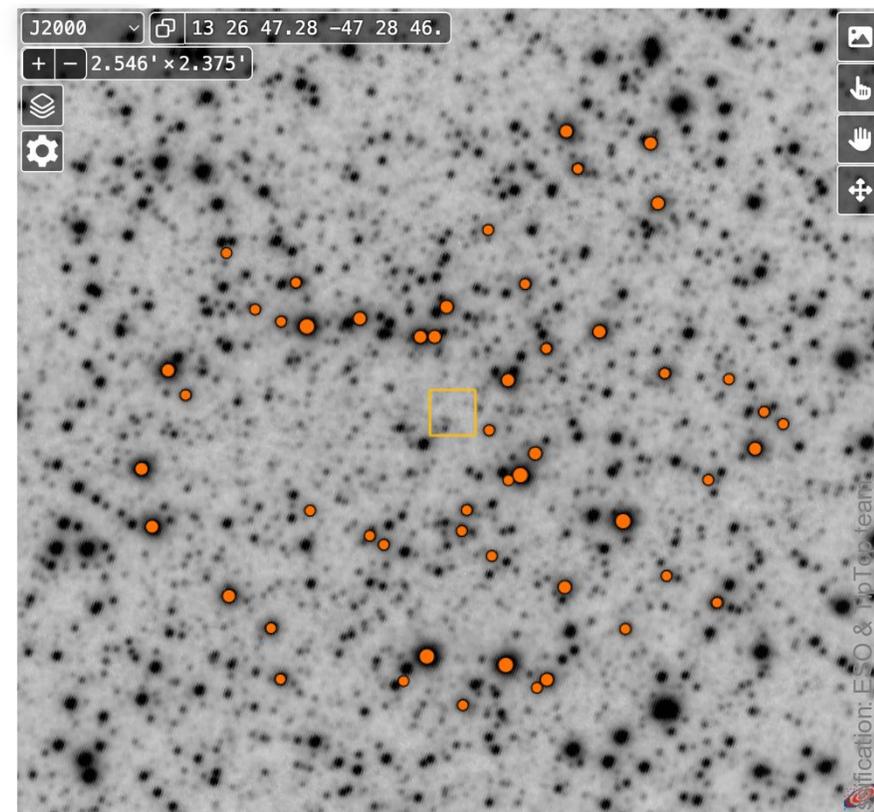
Obs. Description Target Constraint Set Time Intervals Ephemeris Target Visibility **★ ObsPrep** Finding Charts

Pointing AO Stars **Observing Offsets** VLT Guide Star

Select 1 TTS star per template in the list below from the ca... [Read more](#)

#1 ERIS_ifs_acq_LGS

RA	Dec	Gaia RP Mag
13.	RA, DEC, Rmag, SR, FWHM	
4.	RA, DEC, Rmag, SR, FWHM	
10.	RA, DEC, Rmag, SR, FWHM	
22.	RA, DEC, Rmag, SR, FWHM	
21.	RA, DEC, Rmag, SR, FWHM	
15.	RA, DEC, Rmag, SR, FWHM	
16.	RA, DEC, Rmag, SR, FWHM	
3.	RA, DEC, Rmag, SR, FWHM	
12.	RA, DEC, Rmag, SR, FWHM	
2.	RA, DEC, Rmag, SR, FWHM	
...		



Data Classification: ESO & TipTop team

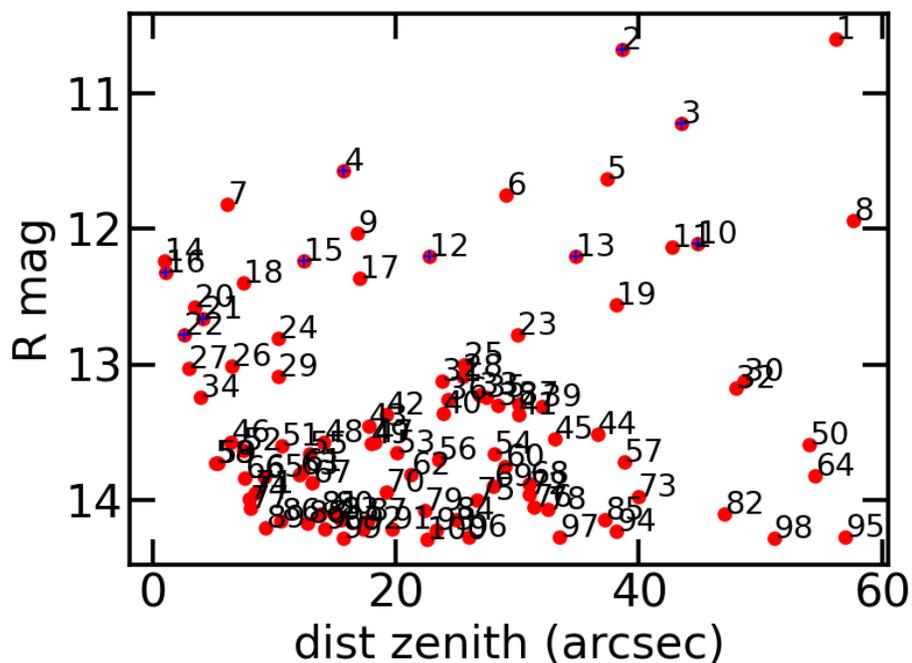


ERIS ObsPrep – TipTop implementation

Target: M30

(ra=ra='21:40:22.12',dec='-23:10:47.5')

Many possible TTS:



60.A-9252(O) · ERIS · **OB** 3073257 tiptop_crowded Exp. Time: 00:00:00 · Exec. Time: 00:00:00 (Partially Defined)

Obs. Description Target Constraint Set Time Intervals Ephemeris Target Visibility **ObsPrep** Finding Charts

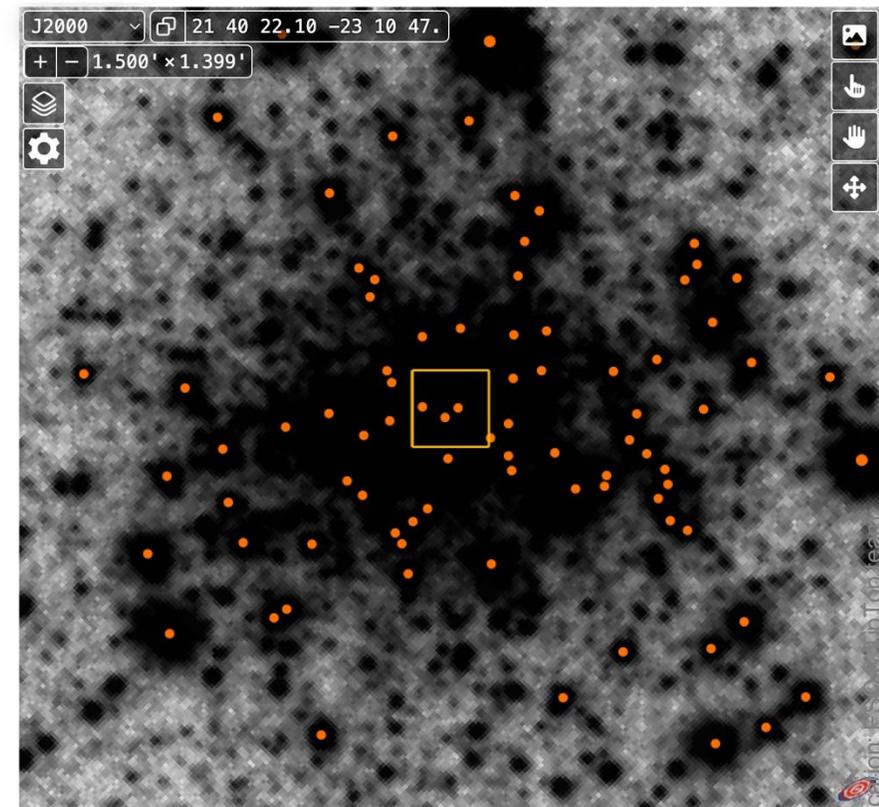
Pointing AO Stars **Observing Offsets** VLT Guide Star

Select 1 TTS star per template in the list below from the ca...
[Read more](#)

#1 ERIS_ifs_acq_LGS

RA Dec Gaia RP Mag

Show Strehl



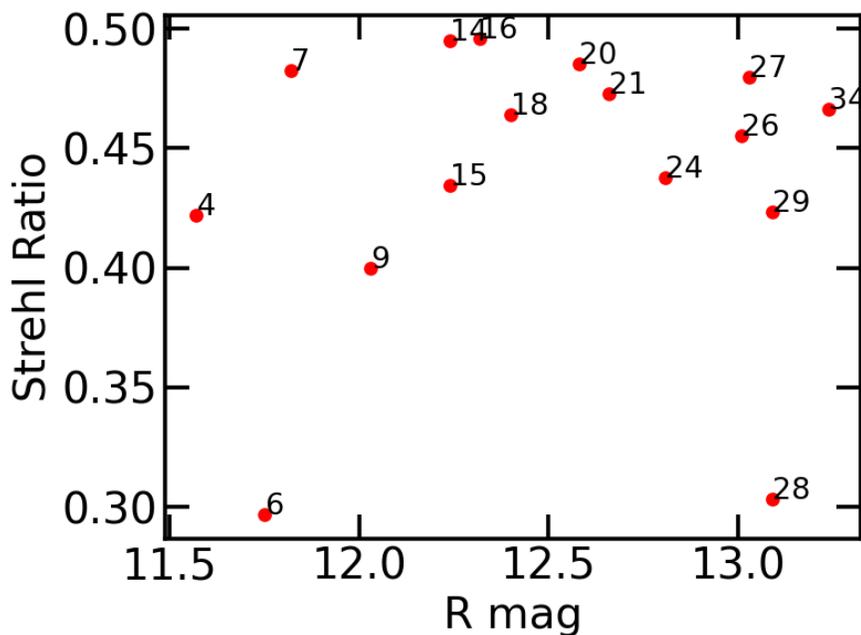


ERIS ObsPrep – TipTop implementation

Target: M30

(ra=ra='21:40:22.12',dec='-23:10:47.5')

Many possible TTS:



60.A-9252(O) · ERIS · OB 3073257 tiptop_crowded Exp. Time: 00:00:00 · Exec. Time: 00:00:00 (Partially Defined)

Obs. Description Target Constraint Set Time Intervals Ephemeris Target Visibility **ObsPrep** Finding Charts

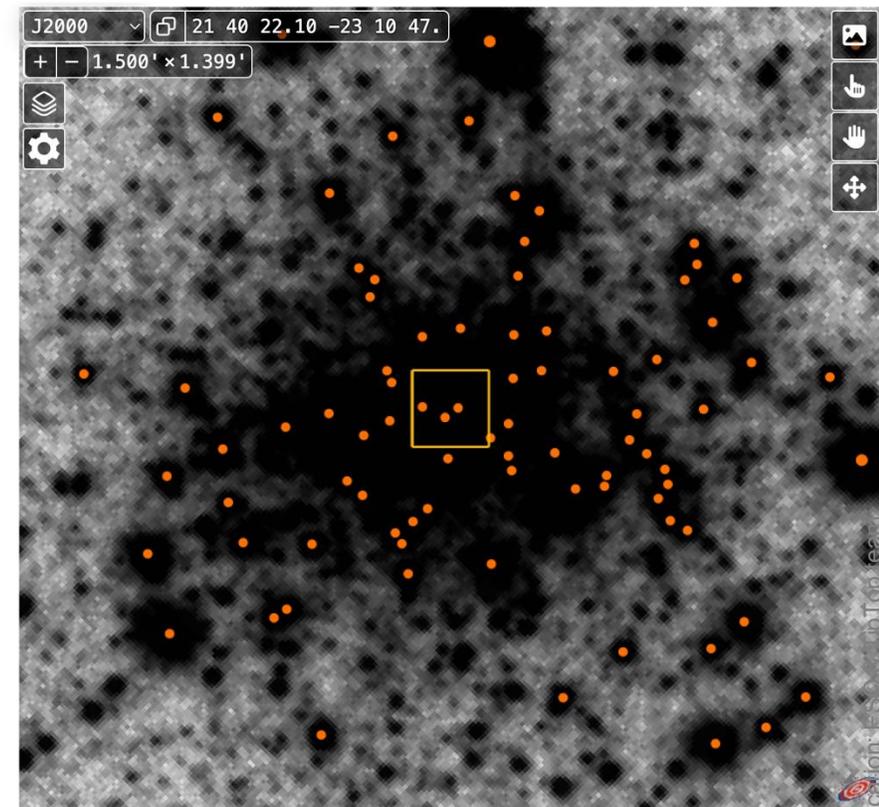
Pointing AO Stars **Observing Offsets** VLT Guide Star

Select 1 TTS star per template in the list below from the ca...
[Read more](#)

#1 ERIS_ifs_acq_LGS

RA Dec Gaia RP Mag

- 16. RA, DEC, Rmag, SR, FWHM
- 14. RA, DEC, Rmag, SR, FWHM
- 7. RA, DEC, Rmag, SR, FWHM
- 20. RA, DEC, Rmag, SR, FWHM
- 21. RA, DEC, Rmag, SR, FWHM
- 27. RA, DEC, Rmag, SR, FWHM
- ...





ERIS ObsPrep – TipTop implementation

Planning a similar implementation in Q2/Q3 2026

Using the already implemented table for AO stars selection, and the “wrapper”.

60.A-9252(O) · ERIS · OB 3073257 tiptop_crowded Exp. Time: 00:00:00 · Exec. Time: 00:13:52 (P)artially Denied

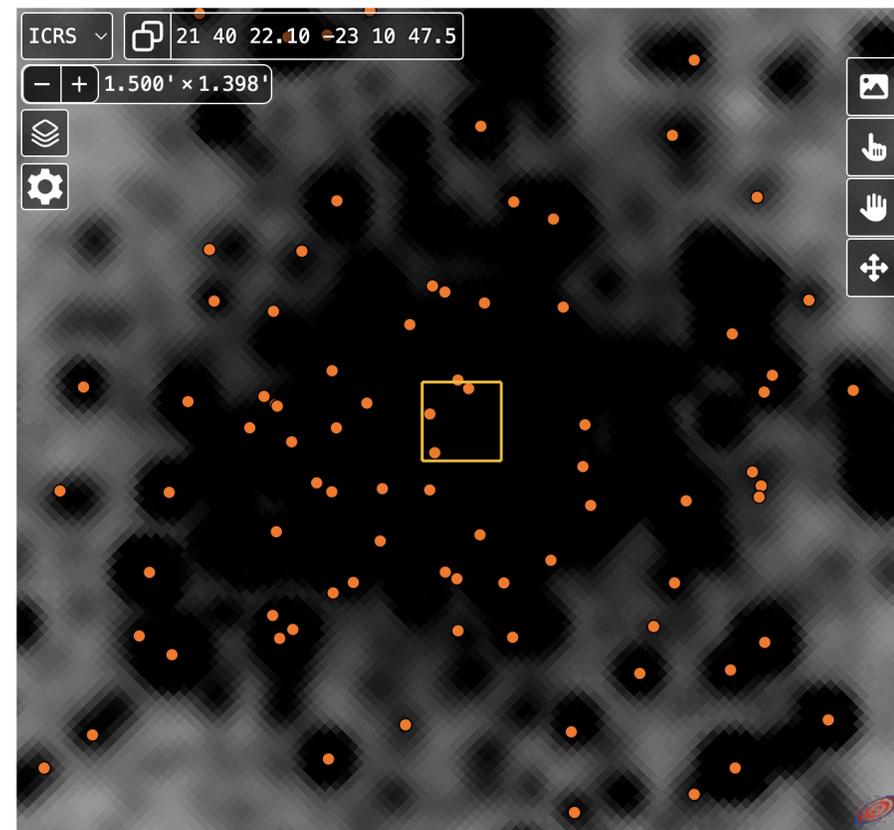
Obs. Description Target Constraint Set Time Intervals Ephemeris Target Visibility **ObsPrep** Finding Charts

Pointing AO Stars Observing Offsets VLT Guide Star

Select 1 TTS star per template in the list below from the ca... [Read more](#)

Nothing selected

Sel.	Dist. ["] ↑	RA	Dec	pmRA ["/yr]	pmDec ["/yr]	Gaia RP Mag
☆	3.3	21:40:22....	-23:10:4...	0.0000	0.0000	13.03
☆	3.4	21:40:22....	-23:10:4...	0.0000	0.0000	12.58
☆	4.2	21:40:22....	-23:10:4...	0.0000	0.0000	12.66
☆	4.2	21:40:22....	-23:10:5...	0.0000	0.0000	13.24
☆	7.7	21:40:22....	-23:10:5...	0.0000	0.0000	13.84
☆	9.8	21:40:22....	-23:10:4...	0.0000	0.0000	14.58
☆	10.5	21:40:22....	-23:10:5...	0.0000	0.0000	12.81
☆	11.1	21:40:22....	-23:10:3...	0.0000	0.0000	14.72
▲	11.7	21:40:22....	-23:10:5...	0.0000	0.0000	15.00





ETC2.0 – TipTop implementation

v. 117.0.0 CRIRES 4MOST ERIS NIRPS HARPS FORS2 UVES

Target Sky **Seeing & IQ** Instrument Exp.time & S/N Plots

AO mode <input checked="" type="radio"/> LGS AO <input type="radio"/> NGS AO <input type="radio"/> No AO	Turbulence Category (TC) 50% (seeing<=1".0) (ETC suppo) <input type="button" value="v"/> NGS R mag 12	Separation 0	Extraction Aperture 9 <input type="button" value="v"/> pixels 0.225 arcsec
--	--	------------------------	---

Compute

Reset

Dynamic S/N Formula

Ready. Compute and hover over the plots to activate the dynamic display

Input form data

Show	Download	Upload	Copy to clipboard
------	----------	--------	-------------------



ETC2.0 – TipTop implementation

PSF **CURRENTLY FROM A DATABASE!**

(Select area with mouse for zoom in) - (Shift + mouse left click for panning X axis)

PSF (normalised to integral = 1)

Angle / arcsec

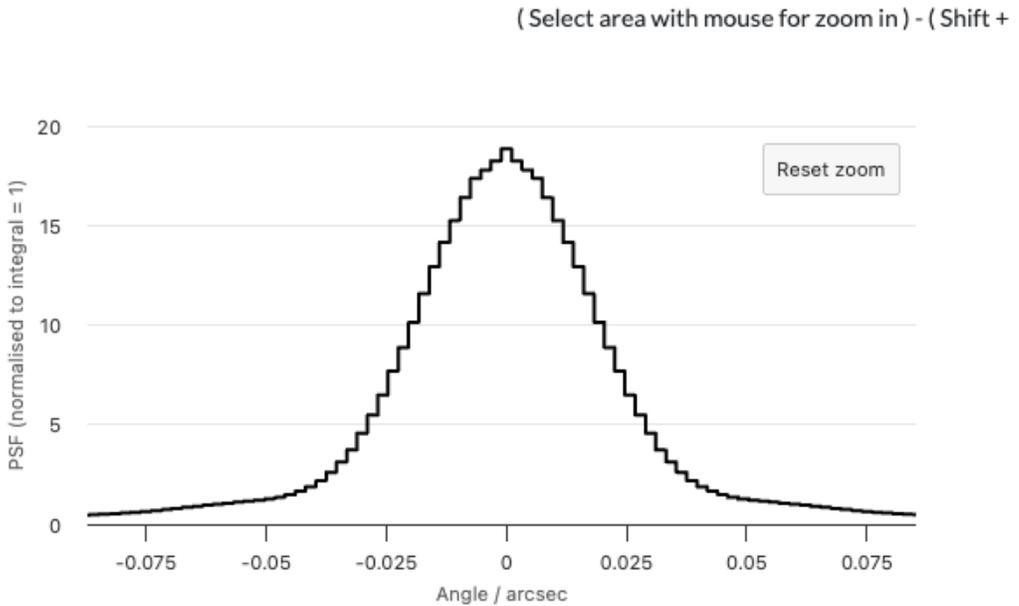
Dynamic S/N Formula

Target: $5.0 \times 10^{-3} e^{-}$	S/N = 50.9 per spectral pixel at $\lambda=1185.28$ nm	Noise ² term fractions:
Sky: $0.184 e^{-}$	$S/N = S_{\text{target}} / \sqrt{S_{\text{target}}^2 + S_{\text{sky}}^2 + \text{npixfactor}^2}$	Target: 0.52
Nspat: 64	$(N_{\text{pix}} \cdot \text{NDIT} \cdot \text{DIT} \cdot \text{dark} + N_{\text{pix}} \cdot \text{NDIT} \cdot \text{RON}^2)$	Sky: < 0.01
Nspec: 1		Dark: < 0.01
$N_{\text{pix}} = N_{\text{spat}} \cdot N_{\text{spec}} = 64$	$50.9 = 5.0 \times 10^{-3} / \sqrt{(5.0 \times 10^{-3})^2 + 0.184^2 + 0.5^2 \cdot (64 \cdot 1^2 \cdot 0.2 + 64 \cdot 1^2 \cdot 11.9^2)}$	RON: 0.48
dark: $0.2 e^{-}/\text{pix/s}$		The dominating noise term is colored blue
RON: $11.9 e^{-}/\text{pix}$	$50.9 = 5.0 \times 10^{-3} / \sqrt{(5.0 \times 10^{-3})^2 + 0.184^2 + 0.5^2 \cdot (25.6 + 9.0 \times 10^3)}$	
DIT: 2 s		
NDIT: 1		
Npix factor: 0.5		

Input form data

Show Download Upload Copy to clipboard

PSF



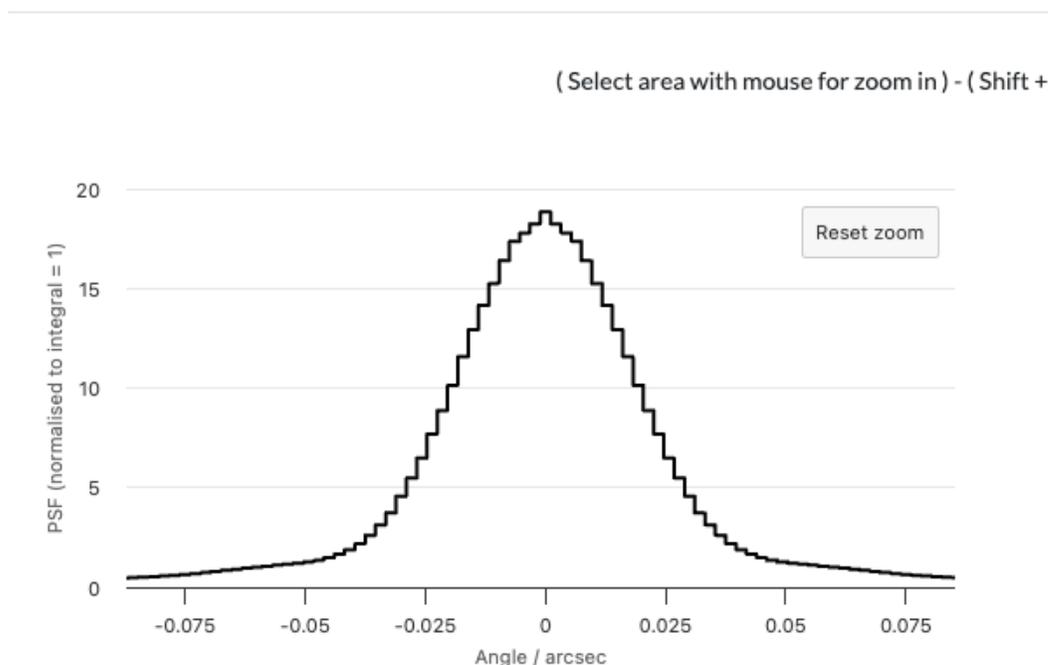


ETC2.0 – TipTop implementation

CURRENTLY UNDER DISCUSSION:

- **Sampling and spatial extent:** fine spacing in the core, coarse spacing in the wings. Ideally should be log-scale, partial solution is to have 2 calls with different spatial coverage and sampling.
- **Normalization:** the PSF is normalized to have integral = 1. The total flux must be known.
- **Wavelength coverage:** multiple wavelengths are simulated to cover the instrument wavelength coverage.
- **Memory allocation:** the calls must be compatible with the available memory on the machine in use. Currently purchasing a bigger GPU server at ESO. In production we will have tens of simultaneous users.

PSF

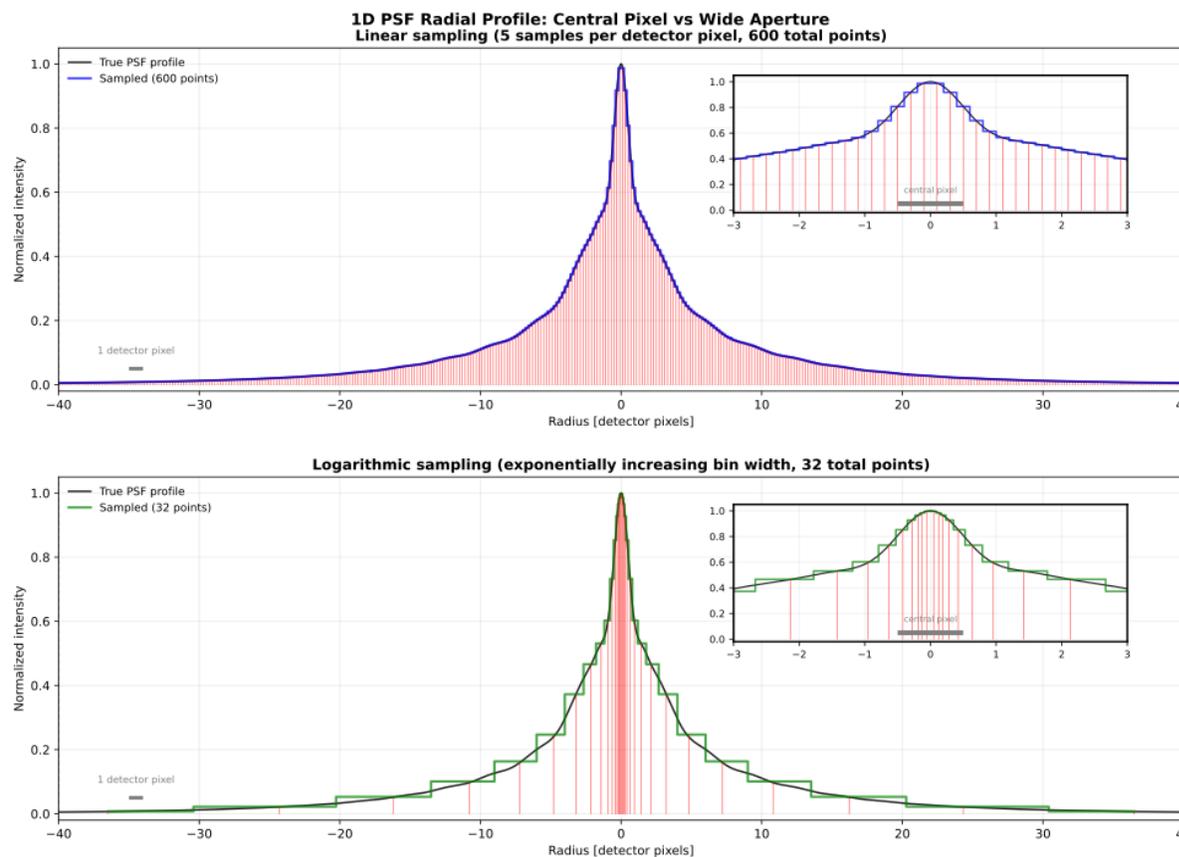




ETC2.0 – TipTop implementation

CURRENTLY UNDER DISCUSSION:

- **Sampling and spatial extent:** fine spacing in the core, coarse spacing in the wings. Ideally should be log-scale, partial solution is to have 2 calls with different spatial coverage and sampling.
- **Normalization:** the PSF is normalized to have integral = 1. The total flux must be known.
- **Wavelength coverage:** multiple wavelengths are simulated to cover the instrument wavelength coverage.
- **Memory allocation:** the calls must be compatible with the available memory on the machine in use. Currently purchasing a bigger GPU server at ESO. In production we will have tens of simultaneous users.



Courtesy of J. Vinther

ETC2.0 – TipTop implementation

- Python server by V. Forchi + code by T. Bierwirth
- Script/wrapper to write ini file on the fly taking information from user inputs

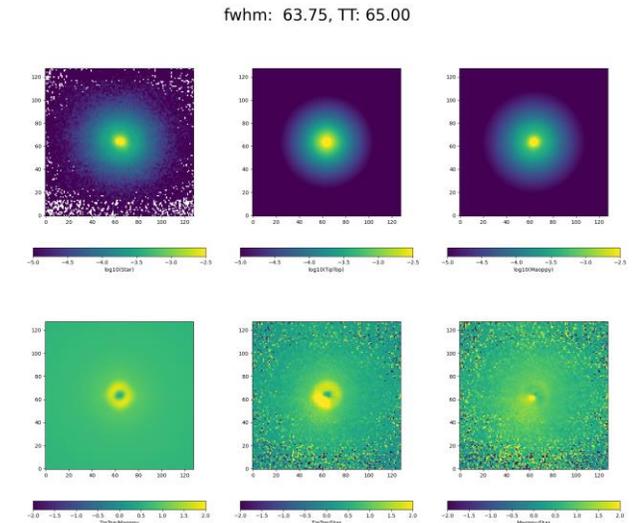
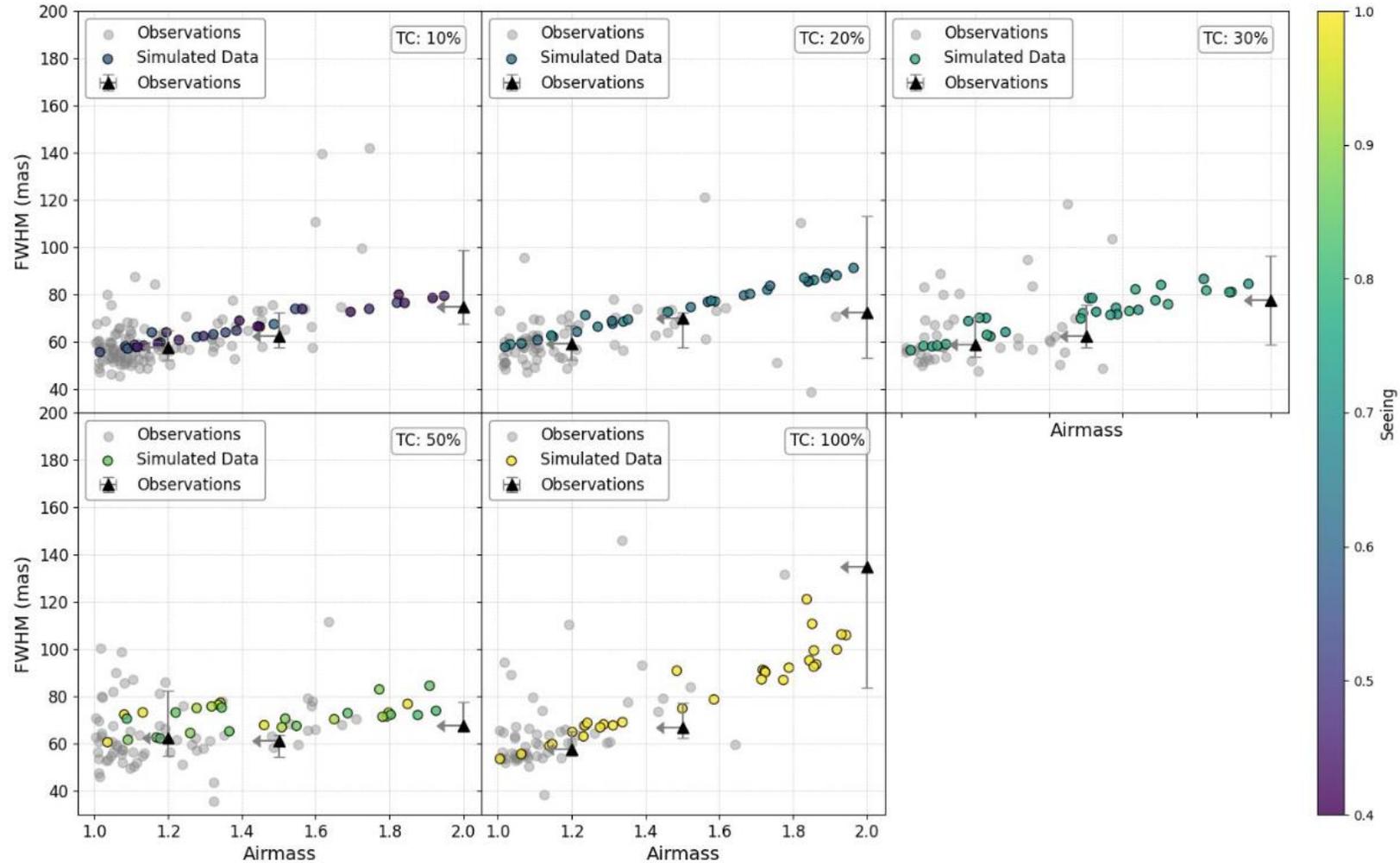
dfs / tiptop / Repository

```

18 sys.stdout = open('etc_tiptop.log', 'w')
19 app = Flask(__name__)
20
21 # based on work by E. Congiu and A. Kuznetsov (with C. Manara) on the weather conditions in the ASM database
22 # integrated with info on the TC to windspeed conditions of the ERIS ETC
23 WIND_SPEED: Dict[int, float] = { 10:10.4, 20:10.5, 30:9.9, 50:10.1, 70:12.8, 85:14.5, 100:15 }
24 WIND_DIR: Dict[int, float] = { 10: 12.5, 20: 28.8, 30: 40.2, 50: 37.2, 70: 40.0, 85: 37.0, 100: 37.0}
25 L0: Dict[int, float] = { 10:12.48, 20:17.1, 30:22.1, 50:23.5, 70:24, 85:24.8, 100:25.5 }
26 GLF: Dict[int, float] = { 10:0.971, 20:0.973, 30:0.980, 50:0.972, 70:0.976, 85:0.977, 100:0.977 }
27
28 # define common input payload
29 class RequestSchema(Schema):
30     # mode = 'LGS' | 'NGS'
31     # wavelengths = array of wavelengths in [m]
32     # airmass = [1 ... 1.96]
33     # separation = [0 ... 60] arcsecs
34     # gsMagnitude = [0 ... 19] arcsecs
35     # turbulenceCategory = 10 | 20 | 30 | 50 | 70 | 85
36     # pixelScale [arcsecs]
37     # fieldOfView [pixel/spaxel]
38     mode = fields.Str(required=True, validate=validate.OneOf(['LGS', 'NGS', 'NFM']))
39     wavelengths = fields.List(fields.Float, required=True, validate=fields.Length(min=1))
40     airmass = fields.Float(required=True, validate=[validate.Range(min=1, max=1.96, error='Airmass must be between 1.0 and 1.96.')]
41     separation = fields.Float(required=True, validate=[validate.Range(min=0, max=60, error='Separation must be between 0 and 60 arcsecs')]
42     magnitude = fields.Float(required=True, validate=[validate.Range(min=0, max=19, error='Magnitude must be between 0 and 19.')]
43     turbulenceCategory = fields.Int(required=True, validate=[validate.OneOf([10, 20, 30, 50, 70, 85], error='Turbulence category must be one of 10, 20, 30, 50, 70, 85')]
44     pixelScale = fields.Float(required=True)
45     fieldOfView = fields.Int(required=True)
46
47 def parse_and_validate_input(request_data):
48     data = RequestSchema().load(request_data)
49     return {
50         'mode': data['mode'],
51         'wavelengths': data['wavelengths'],
52         'airmass': data['airmass'],
53         'separation': data['separation'],
54         'magnitude': data['magnitude'],
55         'turbulence_category': data['turbulenceCategory'],
56         'pixel_scale': data['pixelScale'],
57         'field_of_view': data['fieldOfView']
58     }

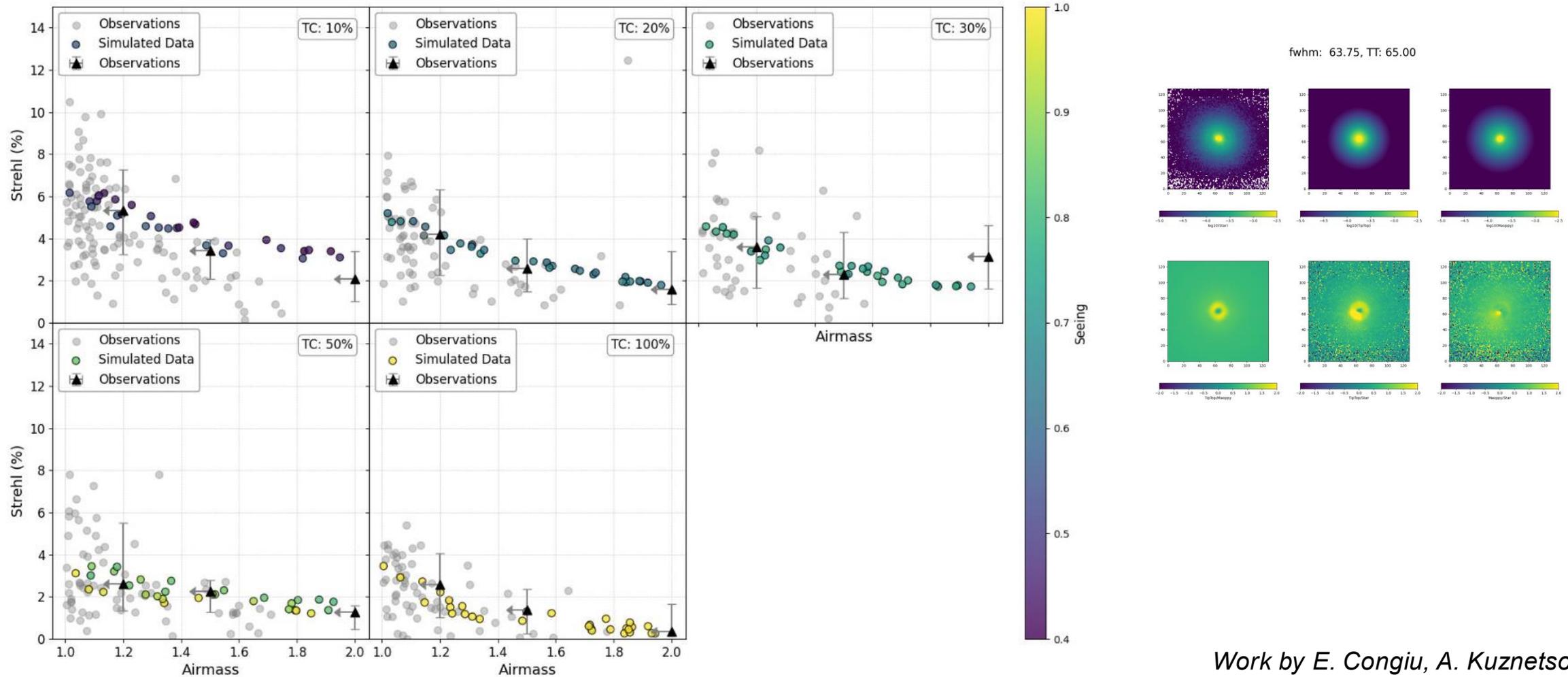
```

MUSE-NFM TipTop VALIDATION: PSF SHAPE



Work by E. Congiu, A. Kuznetsov et al.

MUSE-NFM TipTop VALIDATION: PSF SHAPE



Work by E. Congiu, A. Kuznetsov et al.



INTEGRATION OF TIPTOP INTO VLT OPERATION

1. Integration in **ETC2.0** (goal: Q1/2026, pending installation of hardware)
MUSE, ERIS, CRIRES, SPHERE are the first instruments
2. **ObsPrep**: Selecting the best NGS/TTS for ERIS (goal: Q2+/2026) and setting the requirements on the telemetry data (goal: Q2/2026)
3. Towards **ELT**: integration with the Guide Star selection
4. Continuous interaction with consortia of forthcoming instruments (e.g., MAVIS, MICADO, METIS, ANDES, HARMONI...)



Thank you!

Carlo F. Manara
USD, TipTop Project Scientist at ESO

-  @ESO Astronomy
-  @esoastronomy
-  @ESO
-  european-southern-observatory
-  @ESOobservatory

