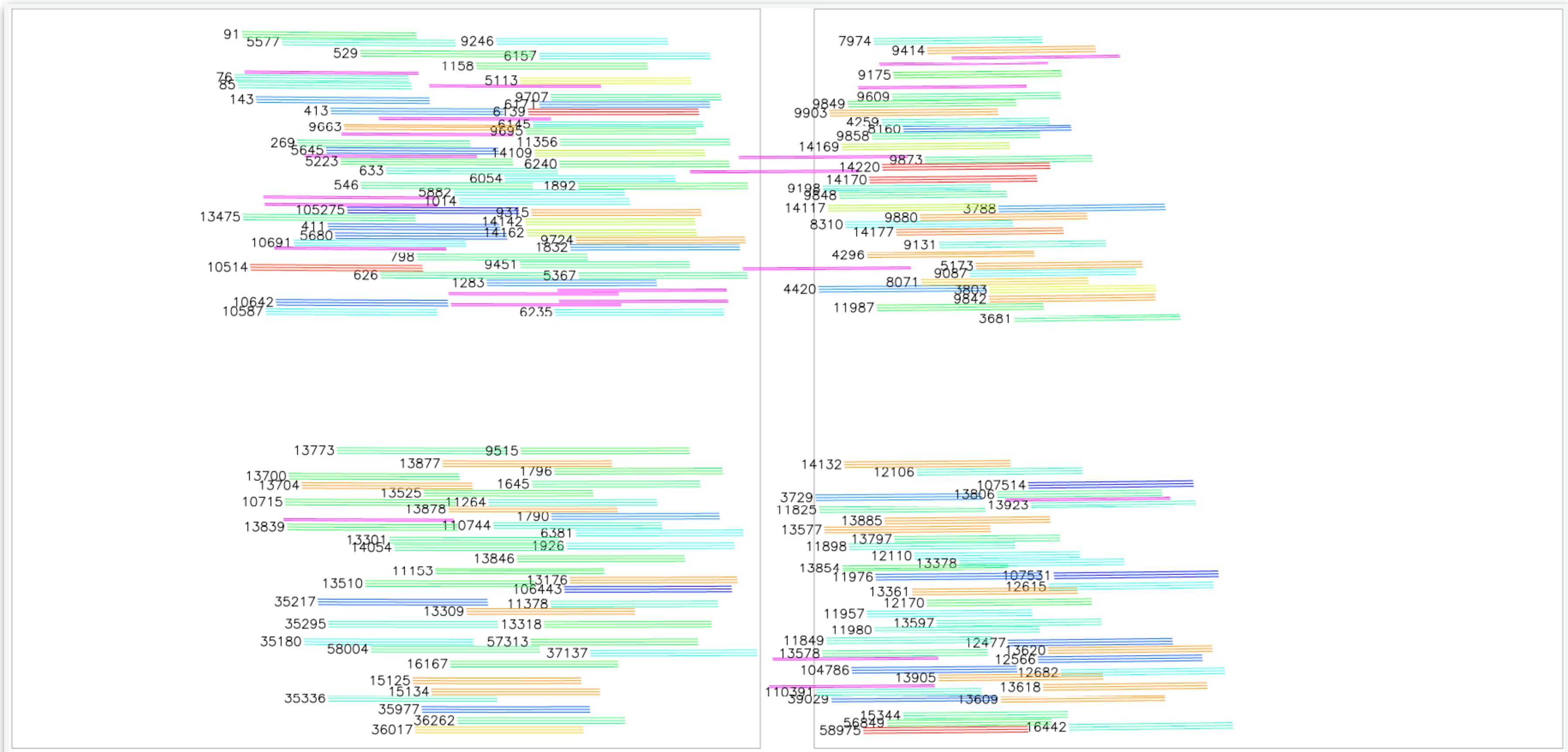
eMPT Software Tutorial



e•M•P•T

noun

alternative to the STScI APT MSA Planning Tool (MPT)

"Let's scientifically optimize our JWST/NIRSpec MSA mask with the superior eMPT software developed by the NIRSpec GTO Team."

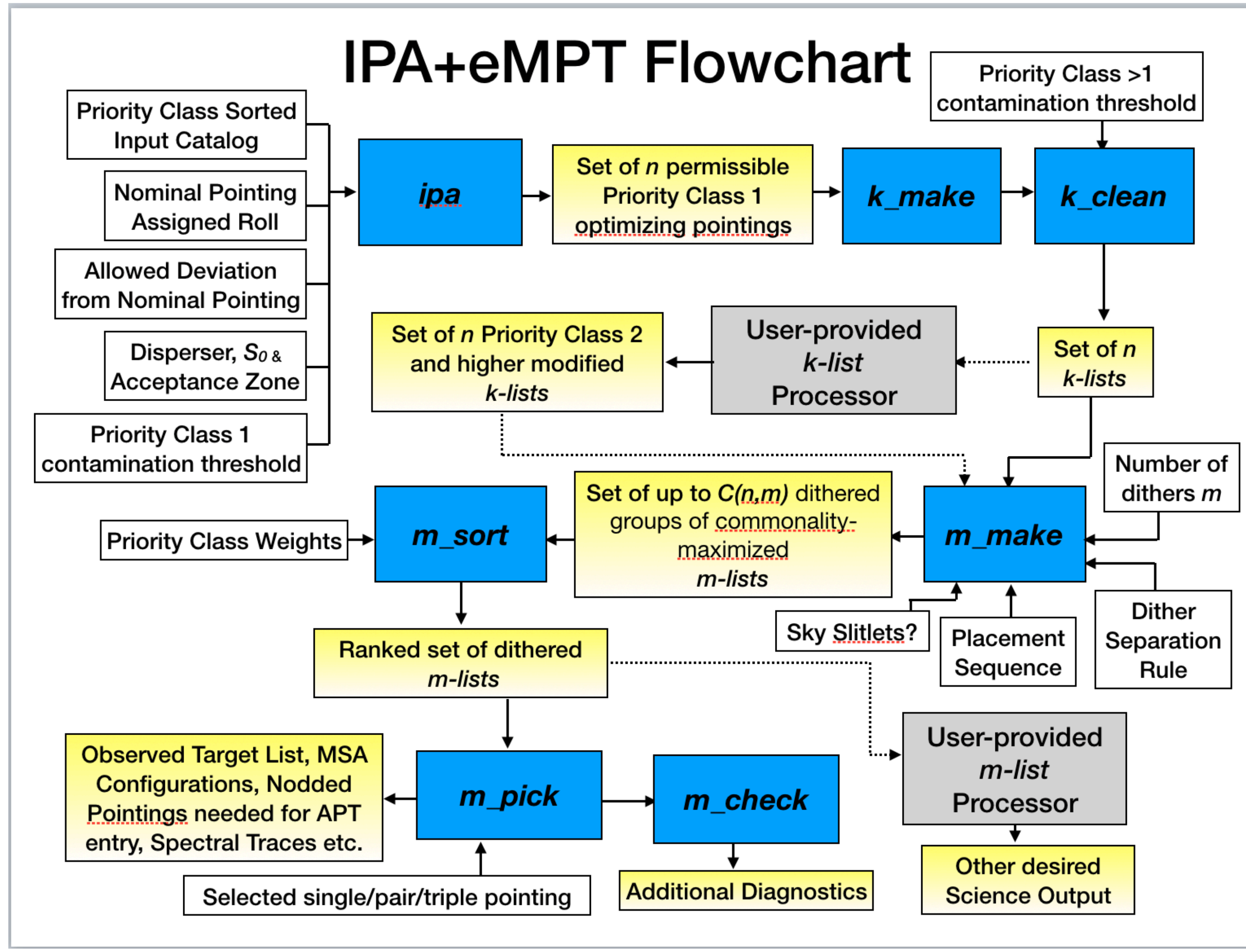
Nina Bonaventura

02 February 2023

eMPT Software Tutorial

Introduction:

The eMPT software suite* includes a set of robust, well-tested, modular *Fortran* subroutines (with a Python wrapper) developed** to produce the most accurate and scientifically optimal Microshutter Array (MSA) mask designs for JWST observations utilizing the NIRSpec MSA mode.



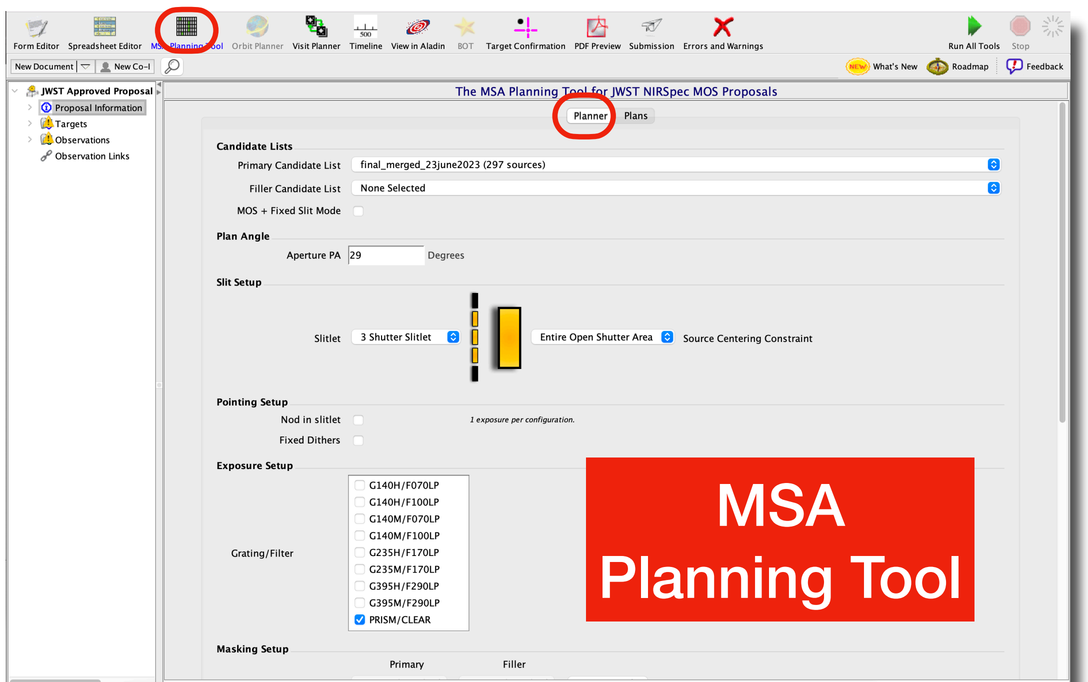
* *https://github.com/esdc-esac-esa-int/eMPT_v1*

** The eMPT was originally developed by Peter Jakobsen to address the ambitious science goals and technical requirements of the NIRSpec GTO science program. It includes algorithms and ideas contributed by multiple team members, most notably Pierre Feruit, Santiago Arribas, and Nina Bonaventura.

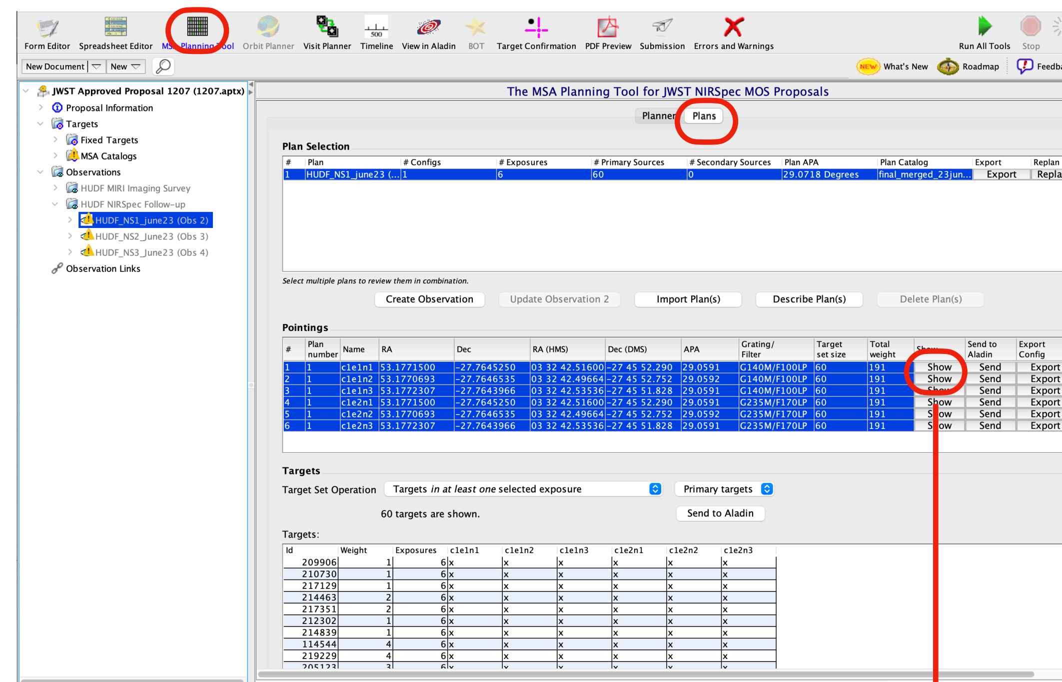
eMPT Software Tutorial

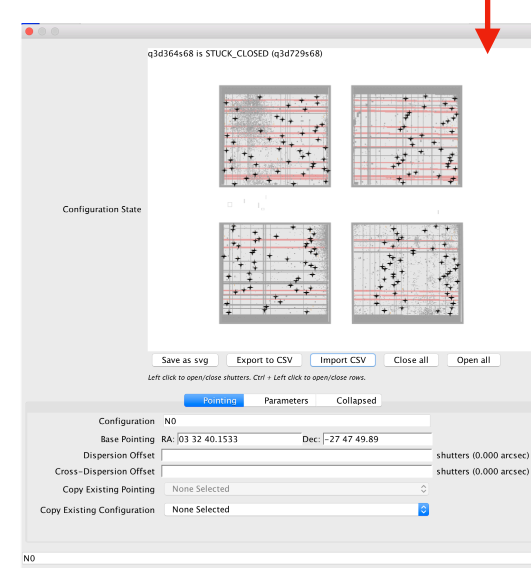
Introduction (continued):

The eMPT is intended as an alternative to the STScI APT MPT software for constructing MSA masks for NIRSpec MOS observations.



STScI Astronomer's Proposal Tool





It does not attempt to duplicate any other functions of the APT beyond the construction of highly accurate and optimal MSA configurations using novel algorithms for multiplexing and pointing optimization.

eMPT Software Tutorial

Introduction (continued):

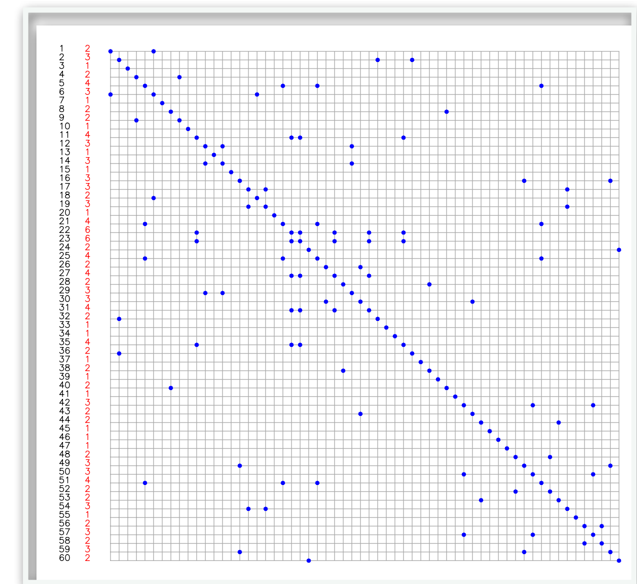
The eMPT outperforms the MPT in four major ways (as of the current date), directly as a result of the novel algorithms and ideas contributed by various team members over a number of years, most/all without a direct analog in the MPT:

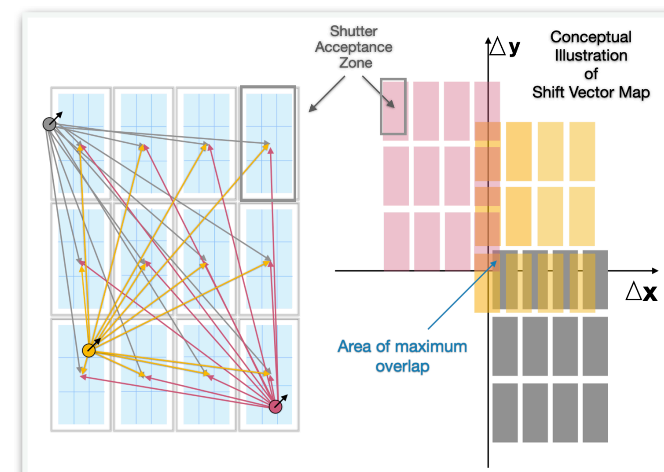
1) Achieves higher multiplexing of non-overlapping spectra with prioritized classes of targets, the **Overlap Matrix Algorithm** that handles them; and a variable, as opposed to a single, fixed, maximum horizontal spectral separation threshold.

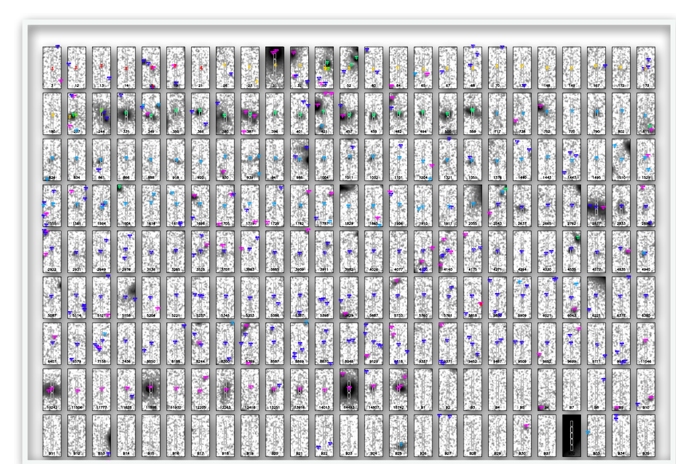
2) Utilizes a fundamentally different optimal pointing search algorithm that analytically and efficiently (i.e., not blindly) isolates the universal set of telescope pointings that maximize the number of highest priority targets simultaneously observed, via its **Initial Pointing Algorithm.**

3) Offers users multiple means of flexibility in the prioritization of individual/groups of targets, as well as the IPA-generated pointings, via a placement sequence matrix and *figure of merit* target/pointing weighting

4) Checks for and eliminates contaminating target spectra







Learn more about the eMPT algorithms and design, here:

The Near-Infrared Spectrograph (NIRSpec) on the *James Webb* Space Telescope

V. Optimal algorithms for planning multi-object spectroscopic observations★

N. Bonaventura[1,2], P. Jakobsen[1,2], P. Ferruit[3], S. Arribas[4] and G. Giardino[5]

+

Received: 8 November 2022 | **Accepted:** 7 February 2023

Abstract

We present an overview of the capabilities and key algorithms employed in the so-called eMPT software suite developed for planning scientifically optimized, multi-object spectroscopic (MOS) observations with the Micro-Shutter Array (MSA) of the Near-Infrared Spectrograph (NIRSpec) instrument on board the *James Webb* Space Telescope, the first multi-object spectrograph to operate in space. NIRSpec MOS mode is enabled by a programmable MSA, a regular grid of ~250 000 individual apertures that projects to a static, semi-regular pattern of available slits on the sky and makes the planning and optimization of an MSA observation a rather complex task. As such, the eMPT package is offered to the NIRSpec user community as a supplement to the MSA Planning Tool (MPT) included in the STScI Astronomer's Proposal Tool (APT) to assist in the planning of NIRSpec MOS proposals requiring advanced functionality to meet ambitious science goals. The eMPT produces output that can readily be imported and incorporated into the user's observing program within the APT to generate a customized MPT MOS observation. Furthermore, its novel algorithms and modular approach make it highly flexible and customizable, providing users the option to finely control the workflow and even insert their own software modules to tune their MSA slit masks to the particular scientific objectives at hand.

Key words: instrumentation: spectrographs / space vehicles: instruments / techniques: spectroscopic

★ The eMPT software package and its associated user guide are available for download from the ESA GitHub page: https://github.com/esdc-esac-esa-int

Including extensive User Guide

eMPT work flow

A sequential process of down-selection

1. ***ipa***
2. ***k_make***

Initial Pointing Algorithm

1. Finds the optimal pointings that maximize the number of user-specified top priority targets simultaneously observed, with input from the 'living' configuration file.
2. Assembles the corresponding raw *k* lists (one per pointing) of all targets in the input catalog that have 'landed' in viable slitlets, including their shutter coordinates and relative intra-shutter target locations.

3. ***k_clean***
4. ***m_make***

Overlap Matrix Algorithm

3. Filters the raw *k* target lists of contaminated targets, creating the clean *k*-lists.
4. Filters the clean *k* target lists of targets that spectrally overlap with targets of higher priority, creating the filtered *m*-lists. For dithered observations, incorporates default or user-edited *placement sequence matrix.*

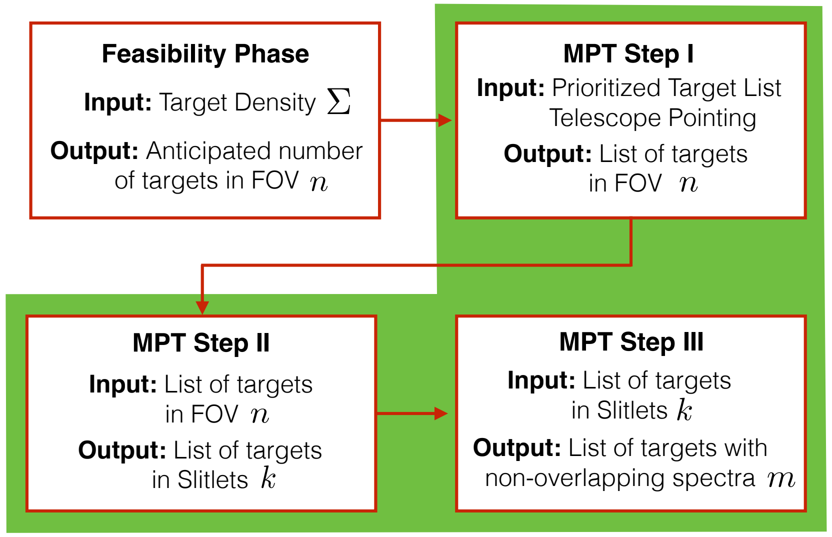
5. ***m_sort***
6. ***m_pick***

5. Calculates the Figure of Merit (FOM) value for each single *m*-list (no dithers) or double/triple set of lists (for 2 or 3 dithers). Sorts results from highest weight (best) to lowest.
6. Automatically accepts the top selected single/double/triple pointing, examines the properties of the associated MSA configuration(s), and gathers the information needed for entry into the STScI APT/MPT system.

On the nomenclature

- ***Pointing:*** the RA,Dec position to which the JWST observatory is commanded to point the NIRSpec FOV in pitch and yaw to perform a given NIRSpec MSA observation
- ***Roll angle:*** Angular orientation of the JWST spacecraft around the telescope axis, here specified as *PA_V3* or *PA_AP (= PA_V3+138.492 deg)*
- ***MSA/slit configuration/mask/design:*** Specification of which MSA shutters are to be commanded open and closed to carry out a given observation. It is understood that a given MSA configuration is always tied to a matching ***pointing*** and ***roll***
- ***Nodding:*** Repointing the telescope to move the target between the three shutters making up its slitlet
- ***Dithering:*** Observing the same target field at multiple offset pointings and MSA configurations

IPA+eMPT module naming traces back to the notation developed in ESA-JWST-SCI-NRS-TN-2016-028:



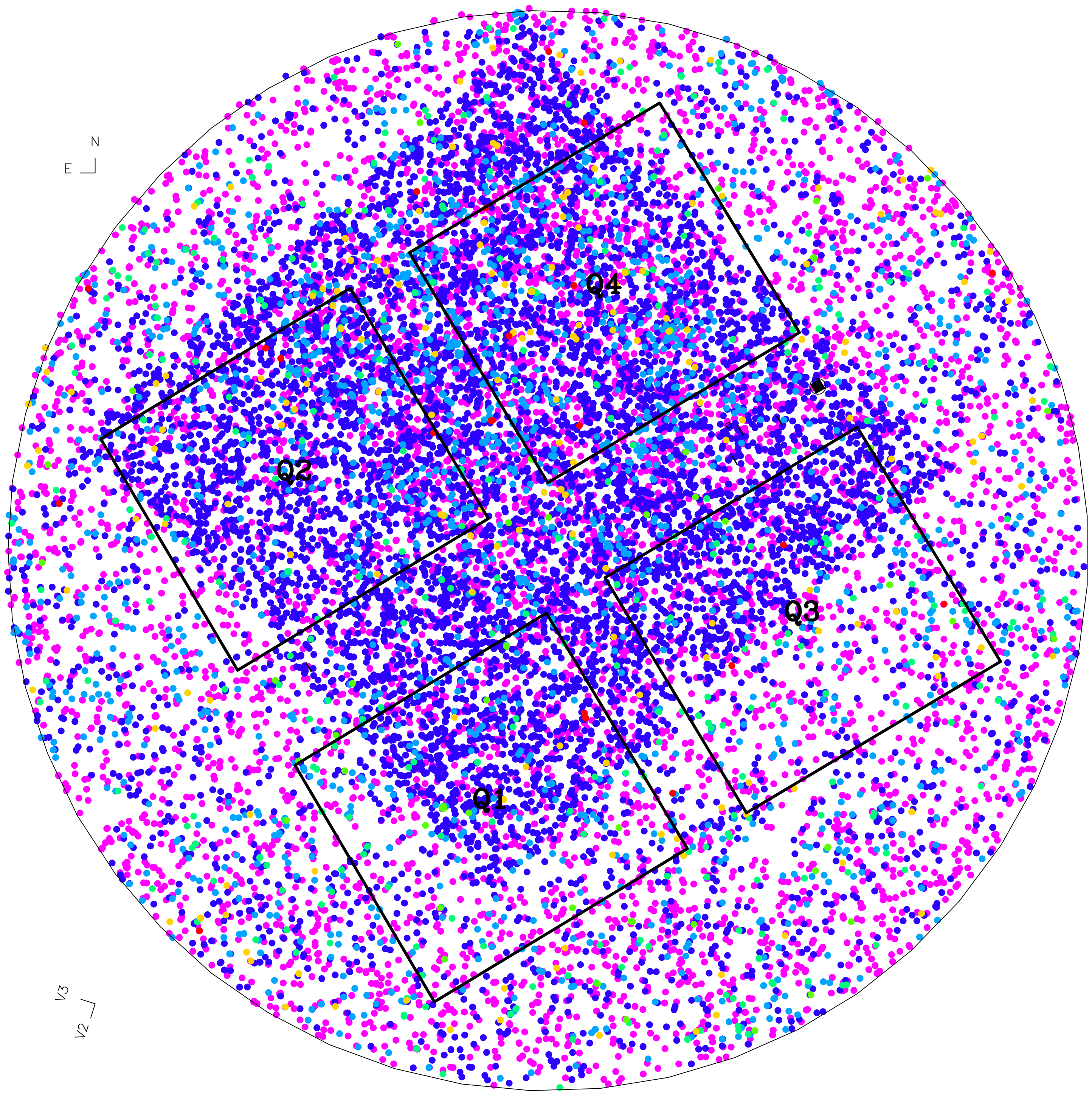
k-list: subset of input catalog targets located in the *Acceptance Zone* of *Viable Slitlets* at a given pointing and roll angle

m-list: subset of the *k-list* targets whose spectra can be accommodated on the NIRSpec FPA without incurring overlap

Credit: P. Jakobsen

eMPT work flow

A sequential process of down-selection



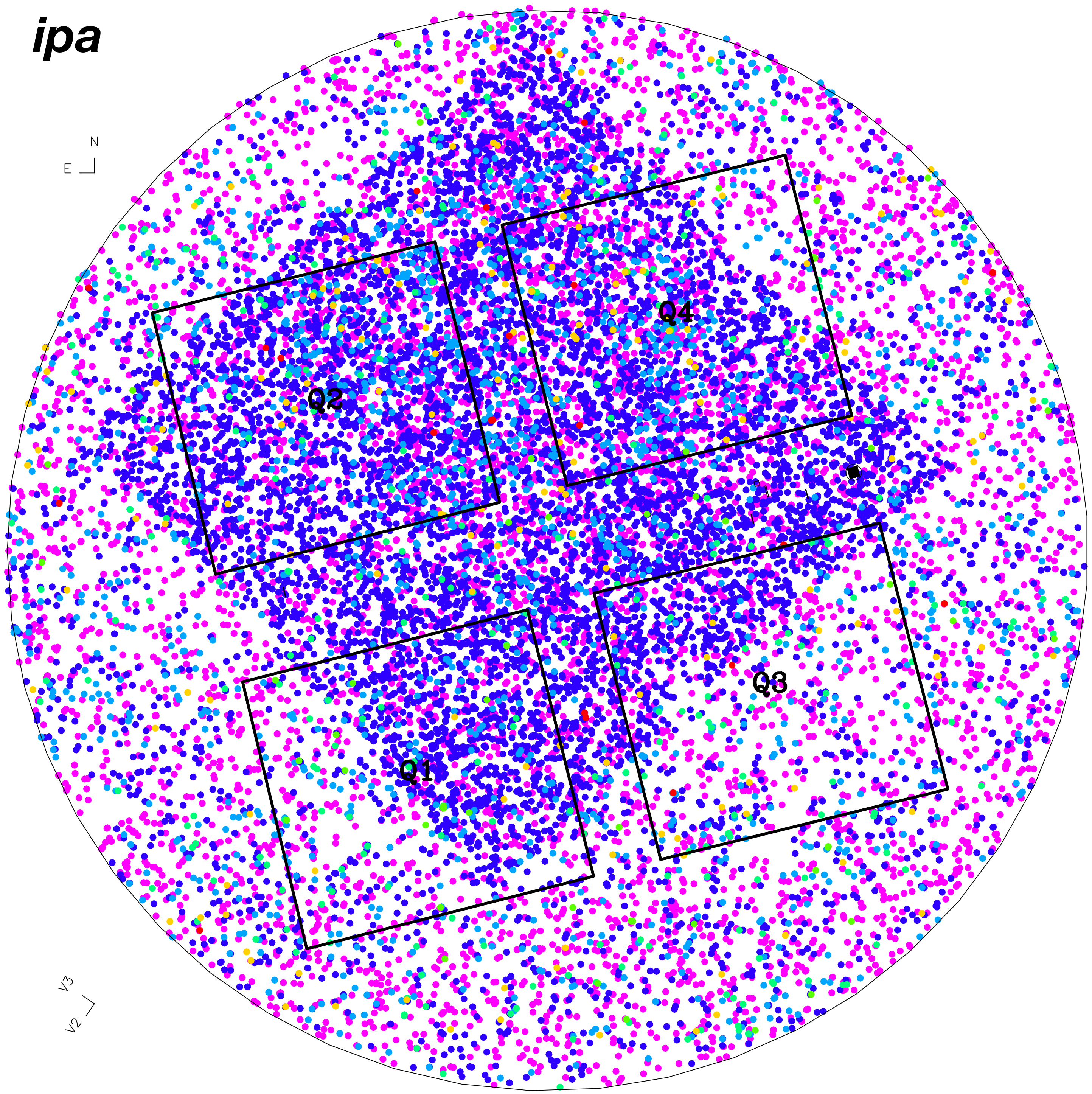
Version of 3DHST-based MEDIUM_HST GTO catalog

Oversized (3’ radius) proposal Input Catalog centered on nominal MSA pointing (RA, Dec and Roll Angle)

16,782 Candidate Targets in the Input Catalog

Credit: P. Jakobsen

eMPT work flow

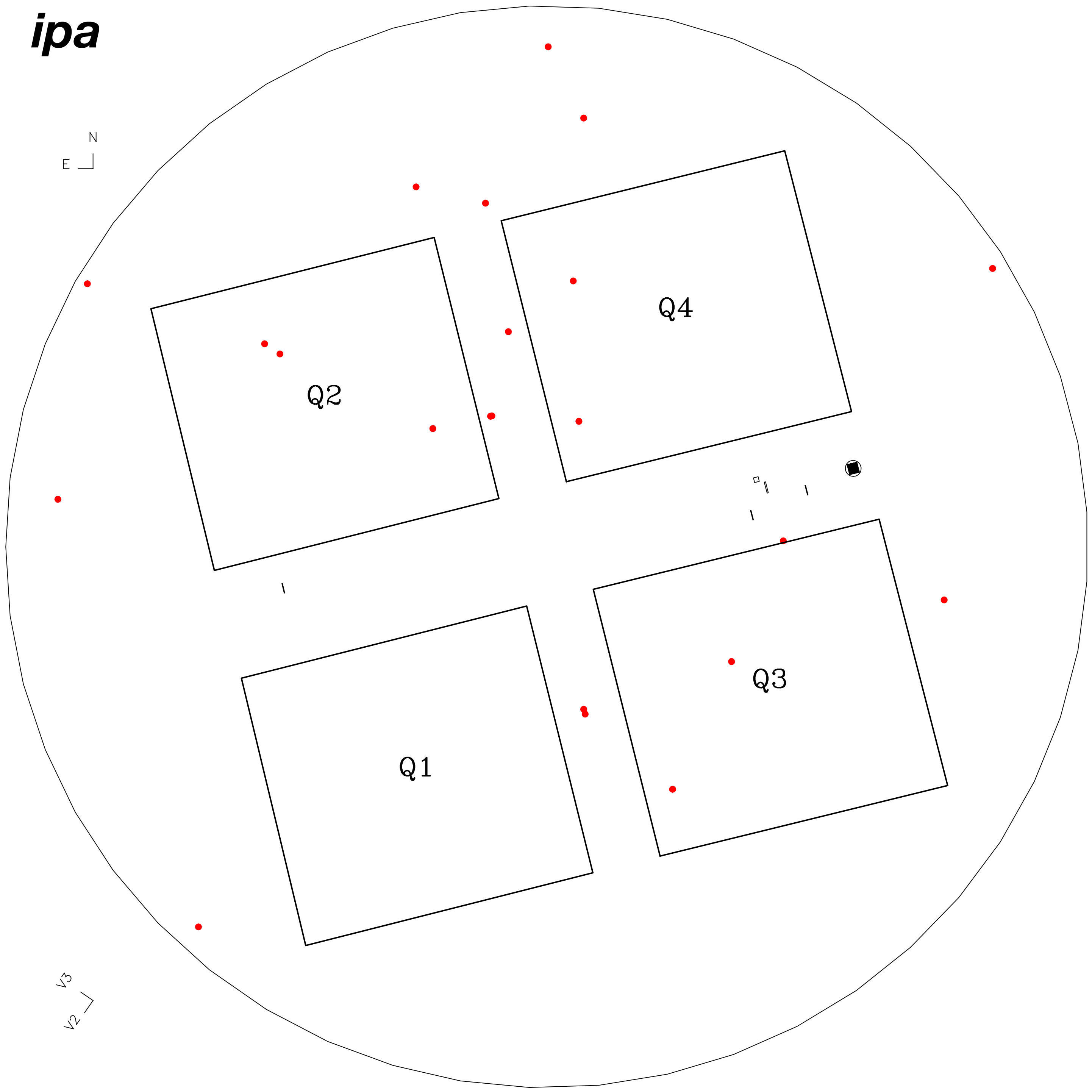


STScI assigns final Roll Angle to observation

16,782 Candidate Targets in the Input Catalog

Credit: P. Jakobsen

eMPT work flow

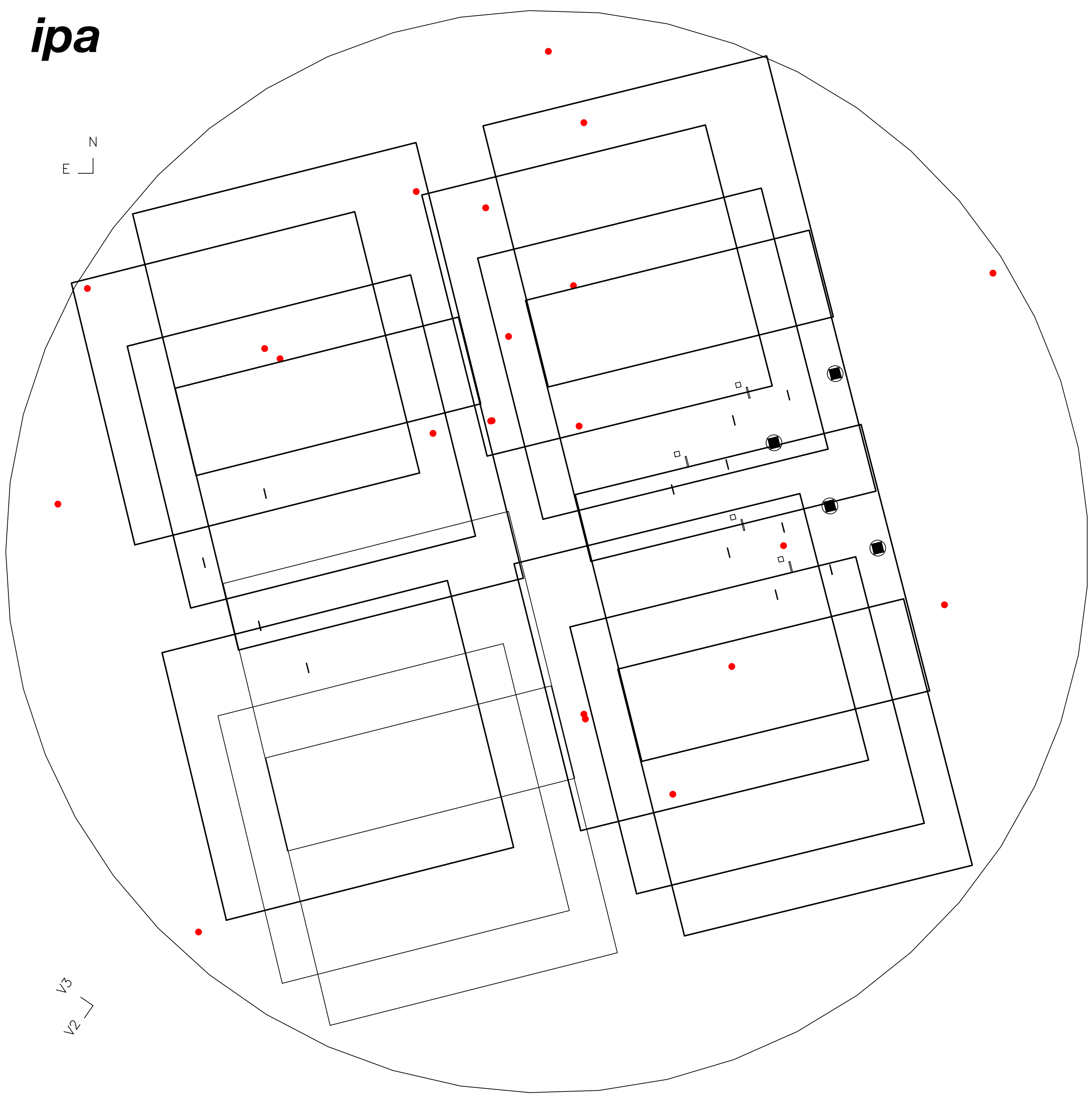


First focus on Priority Class 1 Targets only

Total of 22 Priority Class 1 Candidate Targets in the Input Catalog

Credit: P. Jakobsen

eMPT work flow



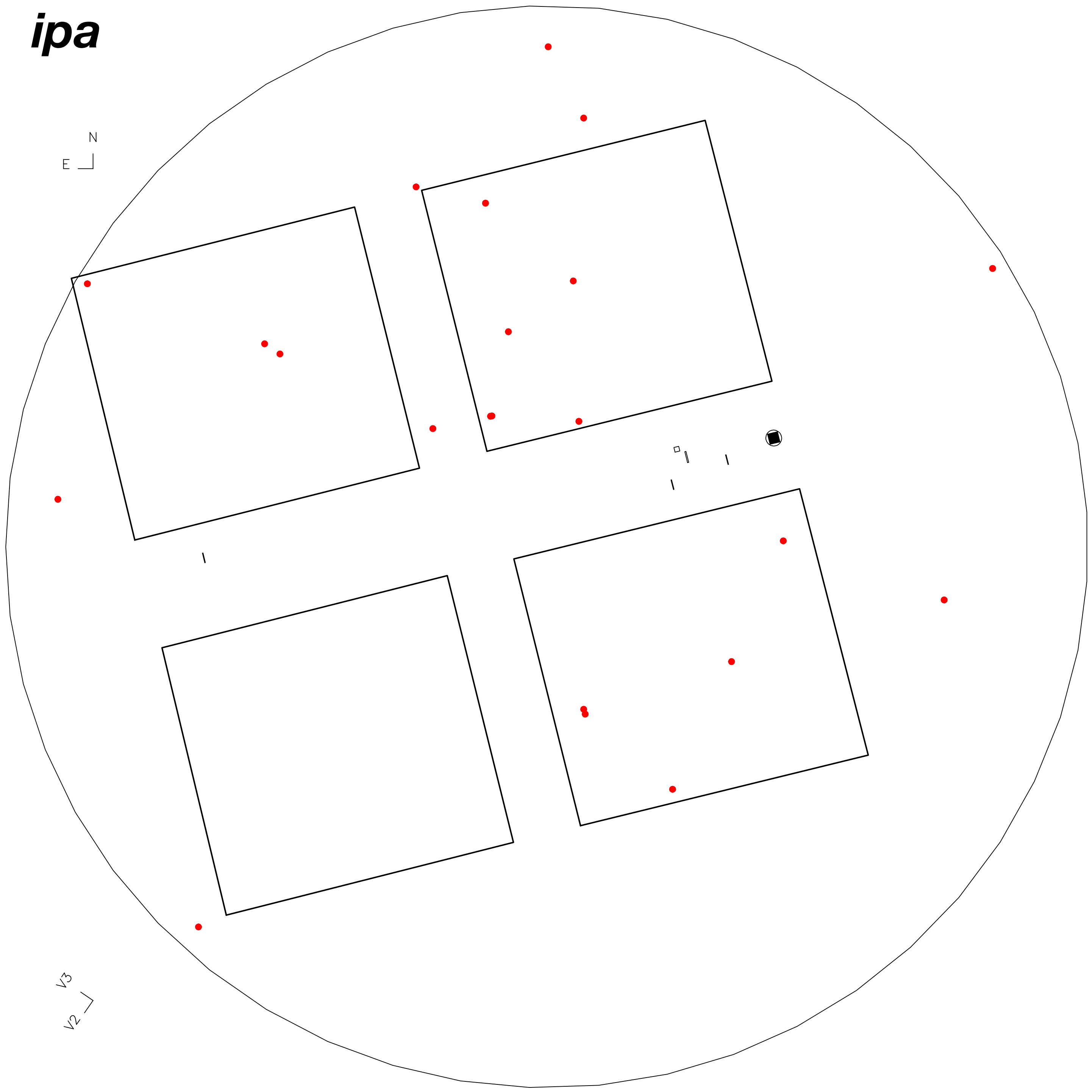
Identify the set of permissible offset pointings in RA,Dec within the catalog boundary that maximize coverage of Priority Class 1 targets at the given fixed Roll Angle

In this case, up to 7 of the 22 Priority Class 1 Candidate Targets can be observed simultaneously

Algorithm capable of finding extremely rare cases of high simultaneous target coverage over a 50” x 50'' search area centered on the nominal pointing.

Credit: P. Jakobsen

eMPT work flow

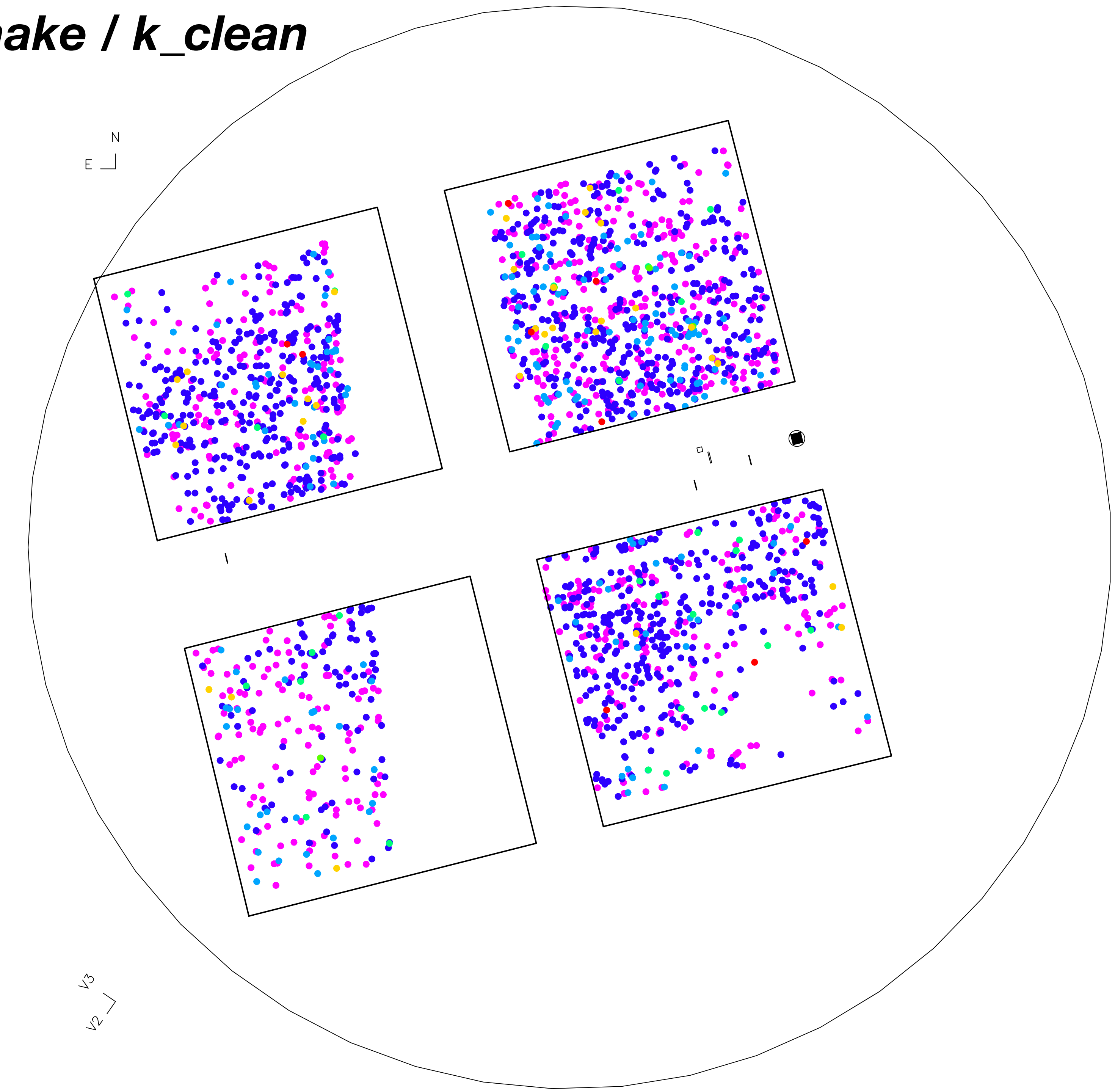


Further evaluate each pointing in the Priority Class 1 optimizing set

Credit: P. Jakobsen

eMPT work flow

k_make / k_clean



Identify the subset of targets in the Input Catalog of all Priority Classes whose positions fall within the Acceptance Zones of Viable Slitlets at the pointing in question

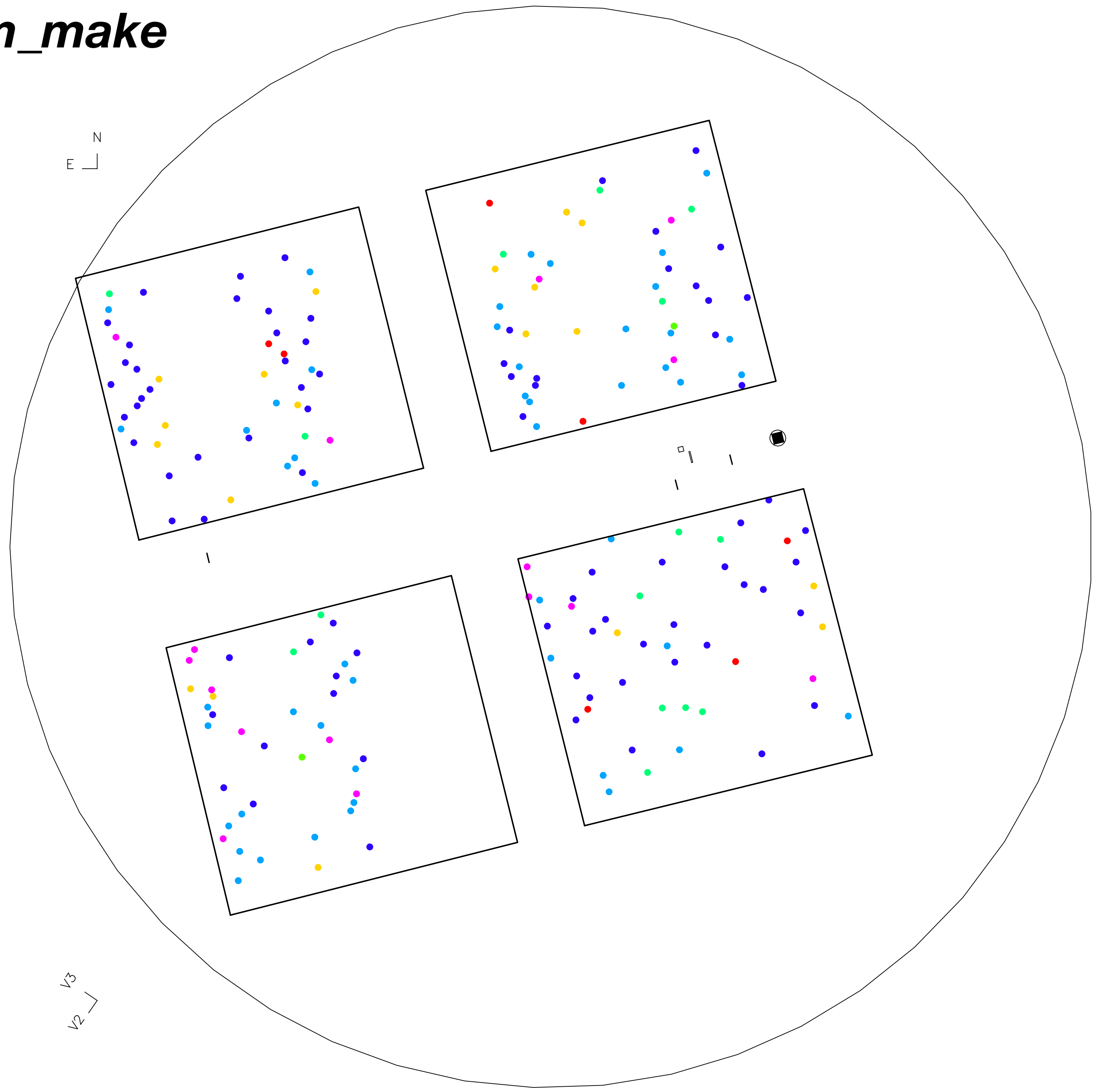
The so-called *k-list*

2,193 Candidate Targets remaining in the *k-list*

Credit: P. Jakobsen

eMPT work flow

m_make



Identify the optimal subset of targets in the *k_list* of all Priority Classes whose spectra can be placed on the NIRSpec Detector Array without incurring overlap

The so-called *m-list*

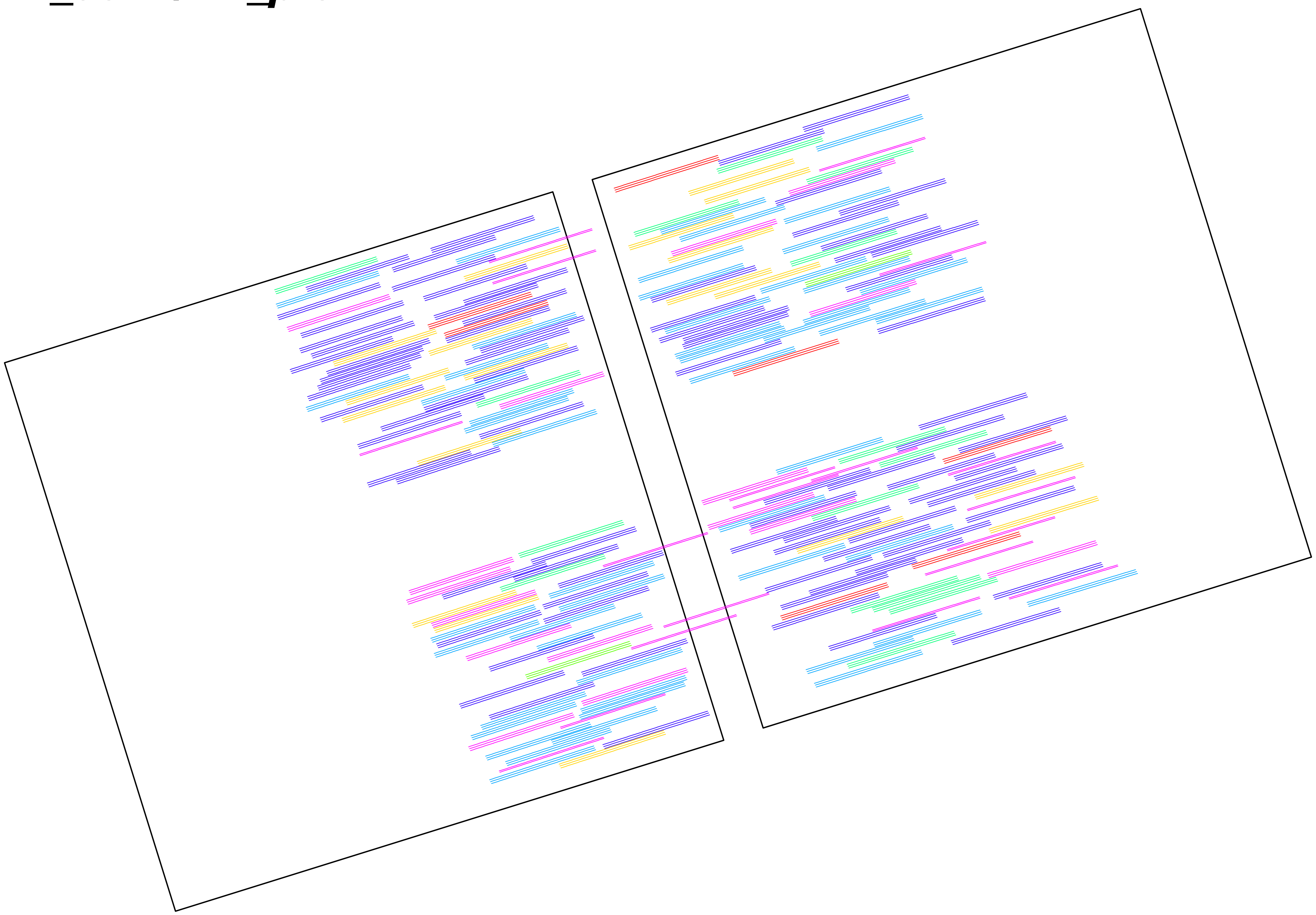
k-list to *m-list* target elimination priorities set by the user

190 Candidate Targets remaining in the *m_list*

Credit: P. Jakobsen

eMPT work flow

m_sort / m_pick



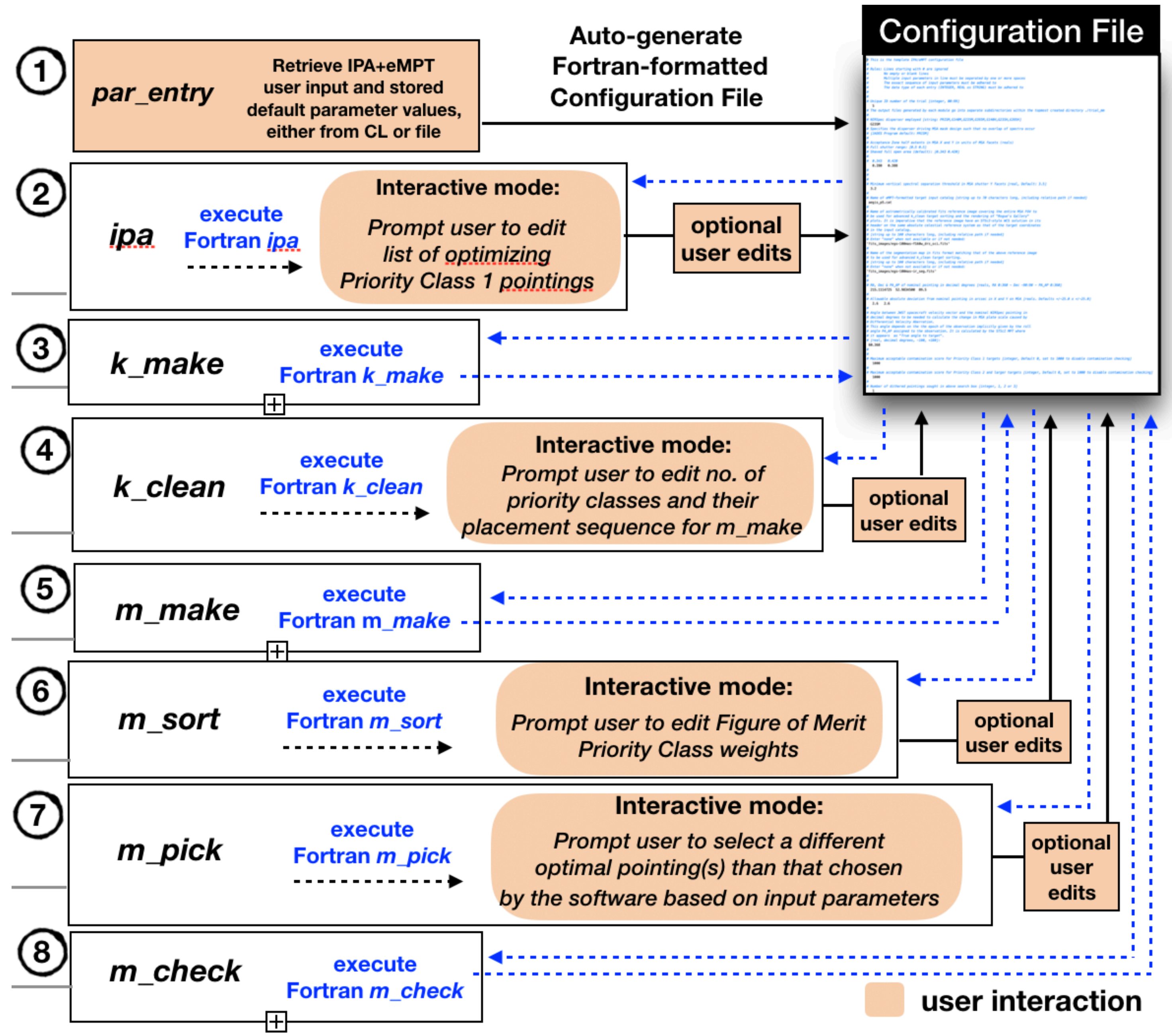
190 PRISM target spectra recorded at IPA pointing under consideration

Quantitatively rank the observed target lists of all pointings in the Priority Class 1 optimized set and select the one(s) most compatible with the scientific objectives of the program

Credit: P. Jakobsen

On the overall architecture

- The task of matching ~250,000 static slitlets to a given catalog of targets in an optimal manner is a *complicated* undertaking.
- There are several key choices that the user needs to make along the way that affect the outcome.
- The eMPT continually ‘talks’ to a controlling ***Configuration File*** throughout the module sequence, giving the user the option to update the file midstream with new parameter values, where appropriate.



- Novice users should critically examine the output of each step and assess whether the default parameter choices suggested by the software for the next step are indeed appropriate to the scientific goals of the NIRSpec program at hand.
- Experienced users familiar with all the issues involved can script the eMPT using an editable Python template script.

IPA+eMPT Modular Workflow

Command-line executables

A collection of Fortran modules executed in the sequence shown below, constitutes a full run of the IPA+eMPT software.

```
% ./ipa 01.conf
% ./k_make 01.conf
% ./k_clean 01.conf
% ./m_make 01.conf
% ./m_sort 01.conf
% ./m_pick 01.conf
% ./m_check 01.conf
```

The user specifies the name of an actively updated configuration file that serves as both a record of the session, and the place for entering required, and changing default, input parameters.

```
# This is the template IPA/eMPT configuration file
#
# Rules: Lines starting with # are ignored
#        No empty or blank lines
#        Multiple input parameters in line must be separated by one or more spaces
#        The exxact sequence of input parameters must be adhered to
#        The data type of each entry (INTEGER, REAL os STRING) must be adhered to
#
#
# Unique ID number of the trial [integer, 00:99]
  6
# The output files generated by each module go into separate subdirectories within the topmost created
  directory ./trial_mm
#
# NIRSpec disperser employed [string: PRISM,G140M,G235M,G395M,G140H,G235H,G395H]
 G235M
# Specifies the disperser driving MSA mask design such that no overlap of spectra occur
# [JADES Program default: PRISM]
#
# Acceptance Zone half extents in MSA X and Y in units of MSA facets (reals)
# Full shutter range: [0.5 0.5]
# Shaved full open area (default): [0.343 0.420]
#
#  0.343   0.420
   0.280   0.388
#
#
# Minimum vertical spectral separation threshold in MSA shutter Y facets [real, Default: 3.5]
 3.2
#
# Name of eMPT-formatted target input catalog [string up to 70 characters long, including relative path if
  needed]
 aegis_p6.cat
#
# Name of astrometrically calibrated fits reference image covering the entire MSA FOV to
# be used for advanced k_clean target sorting and the rendering of "Rogue's Gallery"
# plots. It is imperative that the reference image have an STScI-style WCS solution in its
# header on the same absolute celestial reference system as that of the target coordinates
# in the input catalog.
# [string up to 100 characters long, including relative path if needed]
# Enter "none" when not available or if not needed:
'fits_images/egs-100mas-f160w_drz_sci.fits'
#
# Name of the segmentation map in fits format matching that of the above reference image
# to be used for advanced k_clean target sorting.
# [string up to 100 characters long, including relative path if needed]
# Enter "none" when not available or if not needed:
'fits_images/egs-100mas-ir_seg.fits'
#
#
# RA, Dec & PA_AP of nominal pointing in decimal degrees [reals, RA 0:360 - Dec -90:90 - PA_AP 0:360]
  215.0390517  52.9880639  89.5
```

Fortran Configuration file

```
# This is the template IPA/eMPT configuration file
#
# Rules: Lines starting with # are ignored
#       No empty or blank lines
#       Multiple input parameters in line must be separated by one or more spaces
#       The exact sequence of input parameters must be adhered to
#       The data type of each entry (INTEGER, REAL os STRING) must be adhered to
#
#
# Unique ID number of the trial [integer, 00:99]
  6
# The output files generated by each module go into separate subdirectories within the topmost created directory ./trial_mm
#
# NIRSpec disperser employed
#  Specifies the disperser driving the MSA mask design such that no overlap of spectra occur
#   legal entries: [string:
#   PRISM/CLEAR, G140M/F070LP ,G140M/F110LP, G235M/F170LP, G395M/F290LP
#   G140H/F070LP ,G140H/F110LP, G235H/F170LP, G395H/F290LP ]
   G235M/F170LP
#
# Acceptance Zone half extents in MSA X and Y in units of MSA facets (reals)
# Full shutter range: [0.5 0.5]
# Shaved full open area (default): [0.343 0.420]
  0.280   0.388
#
# Minimum vertical spectral separation threshold in MSA shutter Y facets [real, Default: 3.5]
  3.2
#
# Name of eMPT-formatted target input catalog [string up to 70 characters long, including relative path if needed]
 aegis_p6.cat
#
#
# Name of astrometrically calibrated fits reference image covering the entire MSA FOV to
# be used for advanced k_clean target sorting and the rendering of "Rogue's Gallery"
# plots. It is imperative that the reference image have an STScI-style WCS solution in its
# header on the same absolute celestial reference system as that of the target coordinates
# in the input catalog.
# [string up to 100 characters long, including relative path if needed]
# Enter "none" when not available or if not needed:
'fits_images/egs-100mas-f160w_drz_sci.fits'
```

The user enters input parameters at the top of a provided template configuration file.

Required:

Trial ID no.
Disperser
Target catalog
Nominal RA, Dec, PA_AP
Number of dithers

Fortran Configuration file

```
# Name of the segmentation map in fits format matching that of the above reference image
# to be used for advanced k_clean target sorting.
# [string up to 100 characters long, including relative path if needed]
# Enter "none" when not available or if not needed:
'fits_images/egs-100mas-ir_seg.fits'
#
#
# RA, Dec & PA_AP of nominal pointing in decimal degrees [reals, RA 0:360 - Dec -90:90 - PA_AP 0:360]
  215.0390517  52.9880639  89.5
#
# Allowable absolute deviation from nominal pointing in arcsec in X and Y on MSA [reals. Defaults +/-25.0 x +/-25.0]
  2.6  2.6
#
#
# Angle between JWST spacecraft velocity vector and the nominal NIRSpec pointing in
# decimal degrees needed to calculate the change in MSA plate scale caused by
# Differential Velocity Aberration.
# This angle depends on the the epoch of the observation implicitly given by the roll
# angle PA_AP assigned to the observation. It is calculated by the STScI MPT where
# it appears  as "True angle to target".
# [real, decimal degrees, -180, +180]:
 60.368
#
#
# Maximum acceptable contamination score for Priority Class 1 targets [integer, Default 1 2 2, set to 100 100 100 to turn off contamination elimination]
  100 100 100
#
# Maximum acceptable contamination score for Priority Class 2 and larger targets [integer, Default 1 2 2, set to 100 100 100 to turn off contamination elimination]
  100 100 100
#
# Number of dithered pointings sought in above search box [integer, 1, 2 or 3]
  1
#
#  -- Modifying any parameter in the above section requires re-running the ipa and subsequent modules ---
#
#IP# -- Marker. Do not Delete or move this line
```

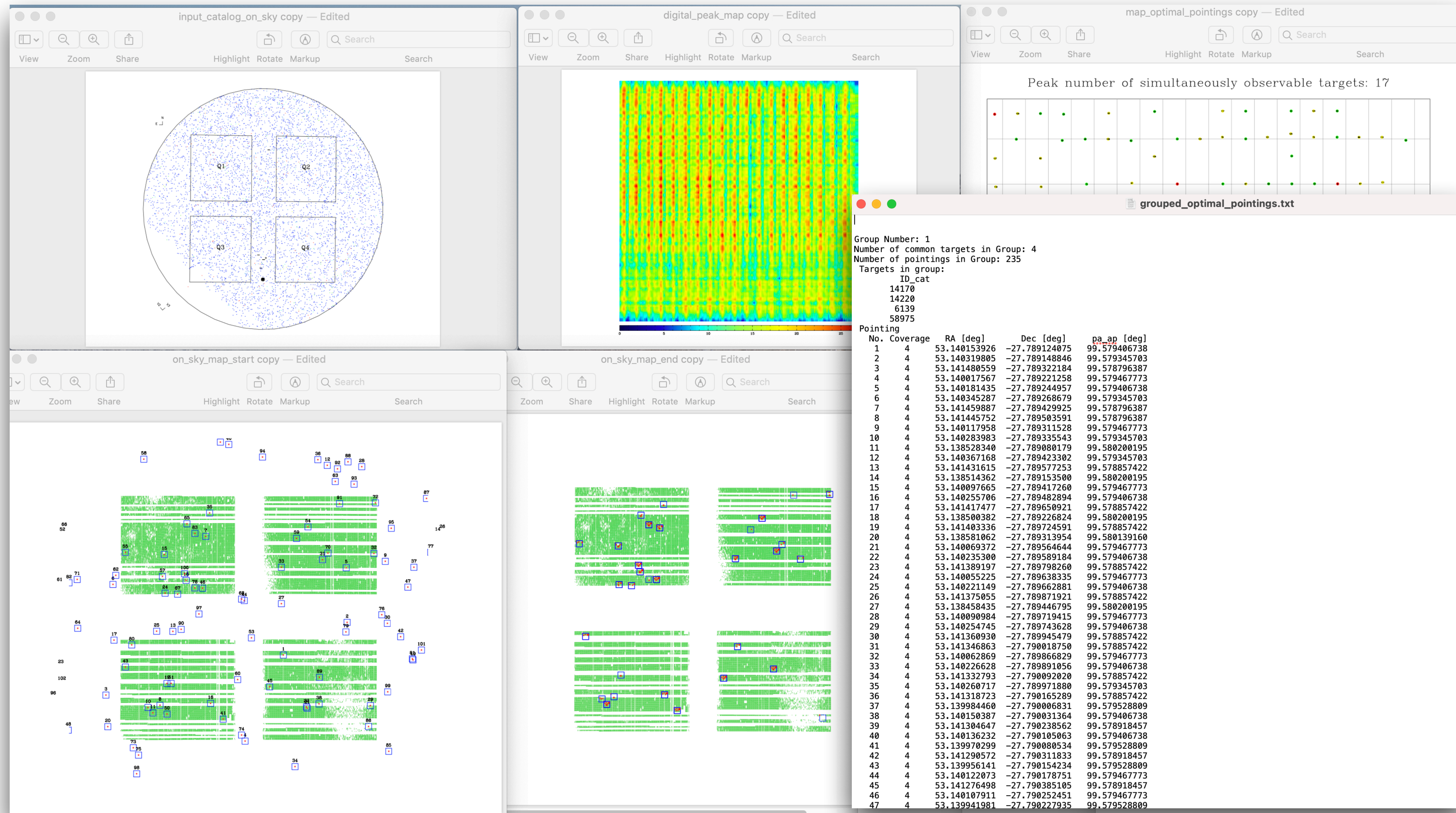
Fortran Configuration file

ipa module output continued

```
# Maximum number of simultaneously observable Priority Class 1 targets: 18
#
# Pointings with 18 simultaneously observable Priority Class 1 targets: 3
# Pointings with 17 simultaneously observable Priority Class 1 targets: 8
# Pointings with 16 simultaneously observable Priority Class 1 targets: 22
#
# Number of different 18 simultaneous target Priority Class 1 groupings: 3
# Number of different 17 simultaneous target Priority Class 1 groupings: 7
# Number of different 16 simultaneous target Priority Class 1 groupings: 20
#
# Detailed sorted lists and maps of all optimal pointings identified by ipa module can be found in the directory:
#
#   ./trial_06/ipa_output/
#
# List of prime candidate optimal pointings automatically generated by the IPA module:
#
# Pointing ID, Class 1 Target Coverage, RA, Dec and PA_AP of optimal pointings:
   1   18    215.039760785   52.987438269   89.500000000
   2   18    215.039752889   52.988030933   89.500000000
   3   18    215.039985934   52.987439552   89.500000000
#
# CAUTION: The above automatically selected candidate pointing list is not necessarily optimal depending on the task
# at hand
#
# Consult the complete IPA output in ./trial_06/ipa_output/
#
# ----- END OF SECTION APPENDED BY THE IPA MODULE ----------
#
#  -- Modifying the above list of optimal pointings requires re-running the k_make and subsequent modules ---
```

Review and optionally edit the set of optimal paintings found by the IPA.

ipa module output continued



Group Number: 1
Number of common targets in Group: 4
Number of pointings in Group: 235
Targets in group:

ID_cat
14170
14220
6139
58975

Pointing

No.	Coverage	RA [deg]	Dec [deg]	pa_ap [deg]
1	4	53.140153926	-27.789124075	99.579406738
2	4	53.140319805	-27.789148846	99.579345703
3	4	53.141480559	-27.789322184	99.578796387
4	4	53.140017567	-27.789221258	99.579467773
5	4	53.140181435	-27.789244957	99.579406738
6	4	53.140345287	-27.789268679	99.579345703
7	4	53.141459887	-27.789429925	99.578796387
8	4	53.141445752	-27.789503591	99.578796387
9	4	53.140117958	-27.789311528	99.579467773
10	4	53.140283983	-27.789335543	99.579345703
11	4	53.138528340	-27.789080179	99.580200195
12	4	53.140367168	-27.789423302	99.579345703
13	4	53.141431615	-27.789577253	99.578857422
14	4	53.138514362	-27.789153500	99.580200195
15	4	53.140097665	-27.789417260	99.579467773
16	4	53.140255706	-27.789482894	99.579406738
17	4	53.141417477	-27.789650921	99.578857422
18	4	53.138500382	-27.789226824	99.580200195
19	4	53.141403336	-27.789724591	99.578857422
20	4	53.138581062	-27.789313954	99.580139160
21	4	53.140069372	-27.789564644	99.579467773
22	4	53.140235300	-27.789589184	99.579406738
23	4	53.141389197	-27.789798260	99.578857422
24	4	53.140055225	-27.789638335	99.579467773
25	4	53.140221149	-27.789662881	99.579406738
26	4	53.141375055	-27.789871921	99.578857422
27	4	53.138458435	-27.789446795	99.580200195
28	4	53.140090984	-27.789719415	99.579467773
29	4	53.140254745	-27.789743628	99.579406738
30	4	53.141360930	-27.789945479	99.578857422
31	4	53.141346863	-27.790018750	99.578857422
32	4	53.140062869	-27.789866829	99.579467773
33	4	53.140226628	-27.789891056	99.579406738
34	4	53.141332793	-27.790092020	99.578857422
35	4	53.140260717	-27.789971880	99.579345703
36	4	53.141318723	-27.790165289	99.578857422
37	4	53.139984460	-27.790006831	99.579528809
38	4	53.140150387	-27.790031364	99.579406738
39	4	53.141304647	-27.790238562	99.578918457
40	4	53.140136232	-27.790105063	99.579406738
41	4	53.139970299	-27.790080534	99.579528809
42	4	53.141290572	-27.790311833	99.578918457
43	4	53.139956141	-27.790154234	99.579528809
44	4	53.140122073	-27.790178751	99.579467773
45	4	53.141276498	-27.790385105	99.578918457
46	4	53.140107911	-27.790252451	99.579467773
47	4	53.139941981	-27.790227935	99.579528809

Fortran Configuration file

k_make module
and
k_clean module
output

```
# ----- SECTION AUTOMATICALLY APPENDED BY THE K_MAKE MODULE ----------
#
# The generated k-list file is located in:
#
#    ./trial_06/k_make_output/k_list_raw.txt
#
# ----- END OF SECTION APPENDED BY THE K_MAKE MODULE ----------
#
#KC# -- Marker do not delete or move this line
#
# ----- SECTION AUTOMATICALLY APPENDED BY THE K_CLEAN MODULE ----------
#
# The modified k-list file with contaminated targets marked with negative Priority Class designations is located
# in:
#
#    ./trial_06/k_clean_output/k_list_mod.txt
#
# Detailed lists of the contaminated targets at each explored pointing are located in:
#
#    ./trial_06/k_clean_output/pointing_n/
#
#    Pointing_1: 12 out of 501 targets flagged as contaminated ( 2.40%) - 0 targets removed from sample
#    Pointing_2: 8 out of 500 targets flagged as contaminated ( 1.60%) - 0 targets removed from sample
#    Pointing_3: 14 out of 500 targets flagged as contaminated ( 2.80%) - 0 targets removed from sample
#
# A combined list of all contaminated targets in the k-list file is located in:
#
#    ./trial_06/k_clean_output/combined_contaminated_target_list.txt
#
# The following suggested default parameters should be reviewed and modified as needed before running the m_make
module:
#
# Number of Priority Classes present in Input Catalog:
          8
#
#
# Order in which segmented k-lists are to be fed to the Arribas algorithm
# Default sequence in strict order of Priority Class:
          1           2           3           4           5           6           7           8
#
#
#
#
# ----- END OF SECTION APPENDED BY THE K_CLEAN MODULE --
```

Optionally edit the number of priority classes to consider and their priority order as 'seen' by the Matrix algorithm

(meaningful for dithers to either maximize on exposure time coverage or total no. of sources).

Fortran Configuration file

```
# ----- SECTION AUTOMATICALLY APPENDED BY THE M_MAKE MODULE ----------
#
# The generated m-list file is located in:
#
#   ./trial_06/m_make_output/single_m_list.txt
#
# Number of pointings:        3
#
# The following default parameters should be reviewed and modified as needed before running the m_sort module:
#
# Figure of Merit Weights for each Priority Class in increasing order
   1.00000000      0.500000000      0.250000000      0.125000000       6.25000000E-02   3.12500000E-02
1.56250000E-02   7.81250000E-03
#
# ----- END OF SECTION APPENDED BY THE M_MAKE MODULE ----------
#
#  -- Modifying any parameter in the above section requires re-running the m_sort and following modules ---
#
#MS# -- Marker do not delete or move this line
#
#
#
# ----- SECTION AUTOMATICALLY APPENDED BY THE M_SORT MODULE ----------
#
# The figure-of-merit ranked single pointing list is located in:
#
#   ./trial_06/m_sort_output/single_list_fom2.txt
#
#
# The following default parameters should be reviewed and modified as needed before running the m_pick module:
#
# Top ranked (FOM2) pointings:
# Pointing   FOM2
      1     19.05
#      2     19.03
#      3     18.88
#
# Automatically fill out final MSA mask with Sky Background Slitlets?
# [Y or N [Default] ]
 N
#
#
# ----- END OF SECTION APPENDED BY THE M_SORT MODULE ----------
```

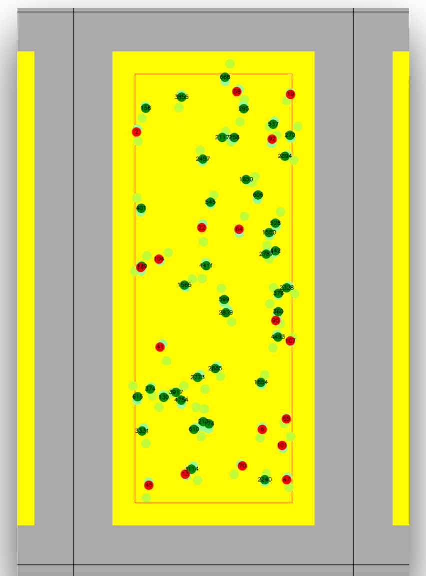
m_make module output

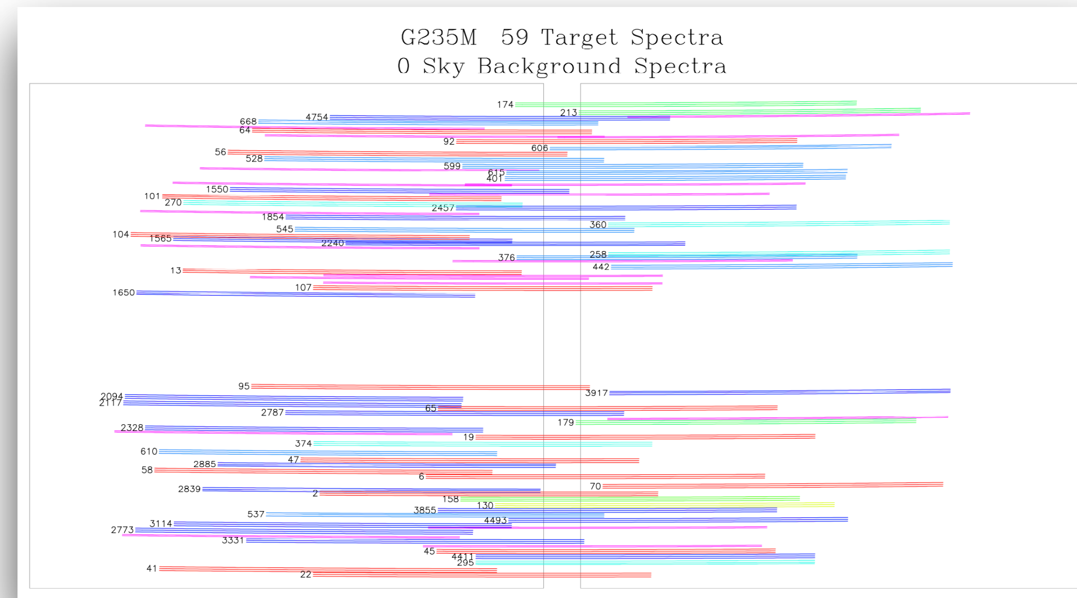
m_sort module output

Fortran Configuration file

m_pick module output

```
# ----- SECTION AUTOMATICALLY APPENDED BY THE M_PICK MODULE ----------
#
#
# Selected single optimal pointing:
#
# IPA Pointing No:          1
# Coordinates and Roll:   215.0397608    52.9874383    89.500
# Number of targets:         59
#
#
# Target breakdown by Priority Class:
#
#    Pri   Number
#     1     18
#     2      0
#     3      1
#     4      1
#     5      3
#     6      5
#     7     11
#     8     20
#
# 0 Sky Background slitlets added to the IPA Pointing 1 MSA Mask
#
#
# The full output from the m_pick module can be found in the directory:
#
#    ./trial_06/m_pick_output/pointing_1/
#
#  and in the file:
#
#    ./trial_06/m_pick_output/pointing_1/target_list.txt
#
#
#
## ----- END OF SECTION APPENDED BY THE M_PICK MODULE ———
```





------- Summary -----------

Single Pointing

IPA Pointing No: 1

Pointing information:

RA, Dec of Central Pointing:
Nod 0: 215.0397608 52.9874383

Roll angle:
PA_AP: 89.5000
PA_V3: 311.0080

RA, Dec of Nodded Pointings:
Nod 1: 215.0400048 52.9874396
Nod 2: 215.0395168 52.9874370

MSA mask optimized for: G235M

Acceptance Zone Thresholds: 0.280 0.388
Acceptance Zone Open Area Filling Factor: 0.435

Overlap Acceptance Threshold: 3.200

Total number of targets in input catalog: 5800
Number of targets in Viable Slitlets: 501
Number of accepted targets: 59

Target breakdown by Priority Class:

PrCl	In Catalog	In Slitlets	Accepted	% Accepted
1	110	21	18	85.71
2	11	0	0	NaN
3	29	1	1	100.00
4	13	1	1	100.00
5	73	7	3	42.86
6	138	7	5	71.43
7	412	41	11	26.83
8	5014	423	20	4.73

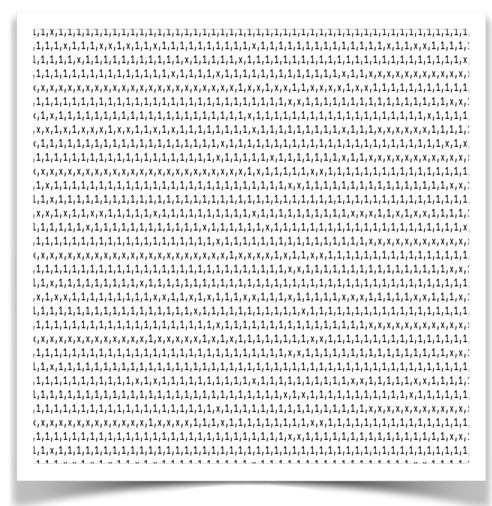
Accepted targets, assigned slitlets and intra-shutter locations:

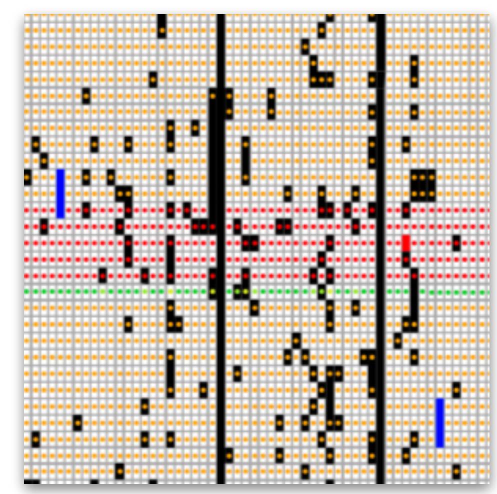
No	ID_sub	ID_cat	pri	k0	i0	j0	rx0	ry0	k1	i1	j1	rx1	ry1	k2	i2	j2	rx2	ry2
1	2	23225	1	4	37	90	-0.275	-0.283	4	37	91	-0.269	-0.266	4	37	89	-0.272	-0.288
2	6	39385	1	2	327	76	0.174	0.255	2	327	77	0.166	0.270	2	327	75	0.174	0.251
3	13	13957	1	3	261	152	-0.101	0.336	3	261	153	-0.075	0.328	3	261	151	-0.098	0.338
4	19	25075	1	2	253	44	0.275	-0.351	2	253	45	0.261	-0.340	2	253	43	0.273	-0.354
5	22	25274	1	4	43	155	-0.042	-0.110	4	43	156	-0.037	-0.084	4	43	154	-0.039	-0.116
6	41	38390	1	4	279	151	-0.191	0.106	4	279	152	-0.168	0.131	4	279	150	-0.182	0.100
7	45	40073	1	2	307	136	-0.231	0.356	2	307	137	-0.240	0.379	2	307	135	-0.231	0.350

m_pick module output
pointing_summary.txt

Selected pointing RA, Dec, & PA_AP input directly into STScI APT/MPT

Import into APT/MPT
shutter_mask.csv



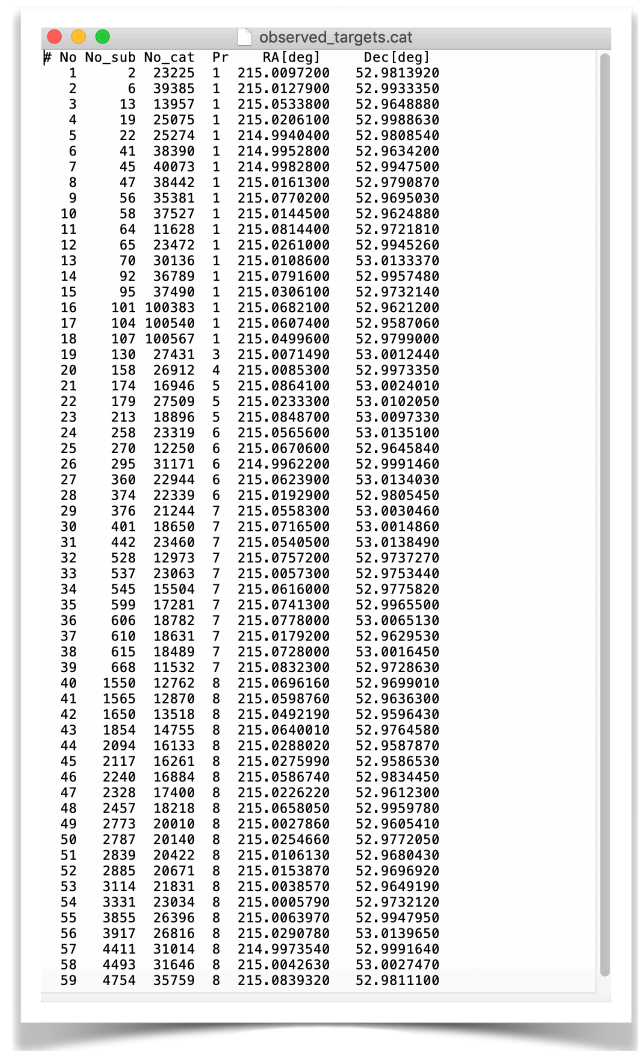


8	47	38442	1	4	68	63	0.261	0.346	4	68	64	0.269	0.359	4	68	62	0.264	0.342
9	56	35381	1	3	200	56	-0.260	-0.038	3	200	57	-0.238	-0.059	3	200	55	-0.258	-0.032
10	58	37527	1	4	292	72	0.083	-0.356	4	292	73	0.109	-0.341	4	292	71	0.090	-0.360
11	64	11628	1	3	164	38	0.091	-0.107	3	164	39	0.110	-0.130	3	164	37	0.092	-0.100
12	65	23472	1	2	312	21	0.260	0.236	2	312	22	0.252	0.244	2	312	20	0.260	0.234
13	70	30136	1	2	57	83	0.103	0.321	2	57	84	0.073	0.336	2	57	82	0.097	0.318
14	92	36789	1	1	299	46	0.209	-0.270	1	299	47	0.199	-0.293	1	299	45	0.207	-0.263
15	95	37490	1	4	148	4	0.222	0.058	4	148	5	0.238	0.064	4	148	3	0.226	0.056
16	101	100383	1	3	298	92	0.245	0.284	3	298	93	0.275	0.268	3	298	91	0.248	0.289
17	104	100540	1	3	344	123	-0.195	-0.053	3	344	124	-0.162	-0.065	3	344	122	-0.191	-0.049
18	107	100567	1	3	59	165	0.274	0.095	3	59	166	0.282	0.089	3	59	164	0.275	0.097
19	130	27431	3	2	220	99	-0.178	0.197	2	220	100	-0.195	0.215	2	220	98	-0.180	0.192
20	158	26912	4	2	273	94	-0.242	-0.326	2	273	95	-0.255	-0.308	2	273	93	-0.244	-0.330
21	174	16946	5	1	211	16	-0.016	0.245	1	211	17	-0.033	0.218	1	211	15	-0.018	0.253
22	179	27509	5	2	101	32	-0.256	-0.040	2	101	33	-0.282	-0.031	2	101	31	-0.260	-0.041
23	213	18896	5	1	113	22	-0.037	0.241	1	113	23	-0.063	0.215	1	113	21	-0.041	0.250
24	258	23319	6	1	59	137	0.074	-0.273	1	59	138	0.043	-0.284	1	59	136	0.070	-0.269
25	270	12250	6	3	265	97	0.273	-0.277	3	265	98	0.300	-0.293	3	265	96	0.276	-0.273
26	295	31171	6	2	247	145	0.108	-0.325	2	247	146	0.094	-0.301	2	247	144	0.106	-0.331
27	360	22944	6	1	61	113	0.231	0.042	1	61	114	0.201	0.028	1	61	112	0.227	0.047
28	374	22339	6	4	49	50	-0.226	0.182	4	49	51	-0.219	0.194	4	49	49	-0.223	0.178
29	376	21244	7	1	199	140	0.232	0.009	1	199	141	0.214	-0.001	1	199	139	0.230	0.012
30	401	18650	7	1	222	76	-0.258	-0.145	1	222	77	-0.274	-0.164	1	222	75	-0.260	-0.139
31	442	23460	7	1	54	147	0.221	-0.068	1	54	148	0.190	-0.077	1	54	146	0.217	-0.064
32	528	12973	7	3	143	61	0.221	-0.118	3	143	62	0.239	-0.139	3	143	60	0.223	-0.112
33	537	23063	7	4	118	107	0.213	-0.297	4	118	108	0.225	-0.277	4	118	106	0.218	-0.302
34	545	15504	7	3	91	118	-0.011	-0.156	3	91	119	0.001	-0.169	3	91	117	-0.010	-0.152
35	599	17281	7	1	288	66	0.038	0.020	1	288	67	0.028	-0.000	1	288	65	0.037	0.026
36	606	18782	7	1	155	51	0.159	-0.169	1	155	52	0.137	-0.191	1	155	50	0.157	-0.161
37	610	18631	7	4	286	57	-0.070	0.255	4	286	58	-0.044	0.268	4	286	56	-0.063	0.252
38	615	18489	7	1	220	71	-0.271	0.196	1	220	72	-0.287	0.176	1	220	70	-0.273	0.202
39	668	11532	7	3	155	31	0.041	-0.382	3	155	32	0.059	-0.406	3	155	30	0.042	-0.375
40	1550	12762	8	3	194	86	0.198	-0.101	3	194	87	0.220	-0.118	3	194	85	0.200	-0.096
41	1565	12870	8	3	278	126	-0.104	-0.006	3	278	127	-0.076	-0.017	3	278	125	-0.100	-0.002
42	1650	13518	8	3	331	170	0.116	-0.197	3	331	171	0.148	-0.202	3	331	169	0.121	-0.195
43	1854	14755	8	3	106	108	0.169	0.170	3	106	109	0.183	0.156	3	106	107	0.171	0.174
44	2094	16133	8	4	342	13	0.255	-0.239	4	342	14	0.286	-0.232	4	342	12	0.262	-0.241
45	2117	16261	8	4	344	18	0.031	-0.273	4	344	19	0.062	-0.265	4	344	17	0.038	-0.275
46	2240	16884	8	3	12	129	0.183	0.346	3	12	130	0.188	0.336	3	12	128	0.183	0.350
47	2328	17400	8	4	309	38	0.261	-0.001	4	309	39	0.290	0.009	4	309	37	0.268	-0.004
48	2457	18218	8	1	295	100	-0.038	-0.234	1	295	101	-0.048	-0.250	1	295	99	-0.039	-0.229
49	2773	20010	8	4	318	120	-0.057	0.161	4	318	121	-0.030	0.182	4	318	119	-0.049	0.156
50	2787	20140	8	4	94	25	0.188	-0.062	4	94	26	0.198	-0.053	4	94	24	0.191	-0.065
51	2839	20422	8	4	217	87	0.044	0.044	4	217	88	0.064	0.061	4	217	86	0.050	0.039
52	2885	20671	8	4	195	67	0.006	0.145	4	195	68	0.024	0.159	4	195	66	0.011	0.141
53	3114	21831	8	4	259	115	-0.079	0.327	4	259	116	-0.057	0.347	4	259	114	-0.072	0.321
54	3331	23034	8	4	147	128	-0.255	0.258	4	147	129	-0.241	0.280	4	147	127	-0.250	0.252
55	3855	26396	8	2	307	103	-0.114	-0.346	2	307	104	-0.124	-0.327	2	307	102	-0.115	-0.351
56	3917	26816	8	2	51	8	-0.134	0.188	2	51	9	-0.164	0.194	2	51	7	-0.139	0.188
57	4411	31014	8	2	247	140	-0.026	-0.041	2	247	141	-0.040	-0.018	2	247	139	-0.028	-0.046
58	4493	31646	8	2	199	111	0.230	0.088	2	199	112	0.212	0.107	2	199	110	0.227	0.083
59	4754	35759	8	3	45	27	-0.115	0.202	3	45	28	-0.106	0.177	3	45	26	-0.115	0.209

0 Sky Background Slitlets added to the MSA Mask:

m_pick module output pointing summary

Import into APT/MPT
observed_targets.cat

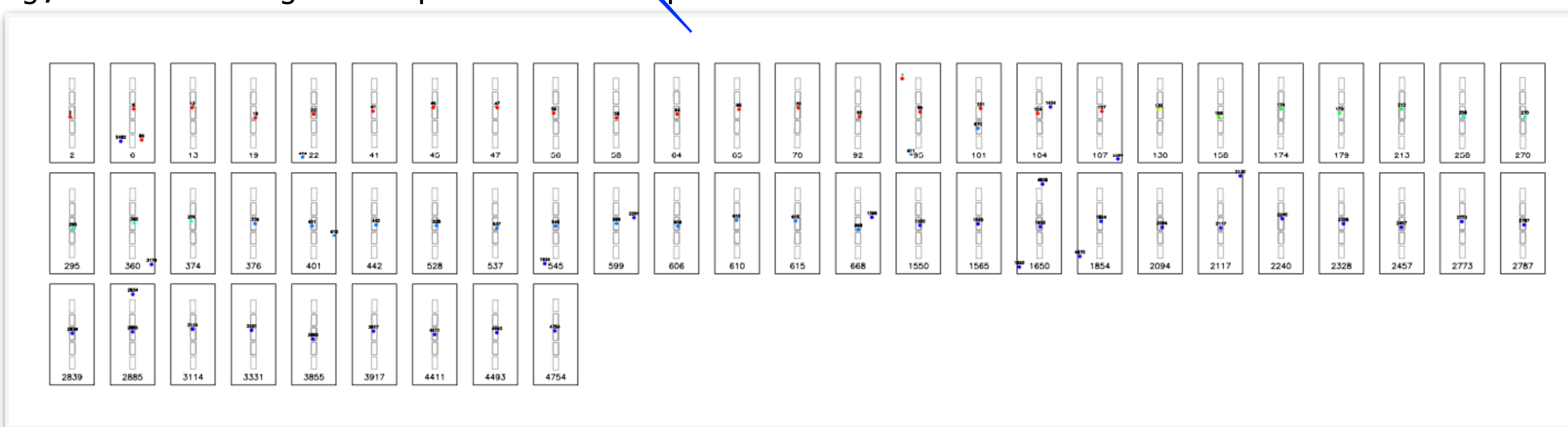

observed_targets.cat

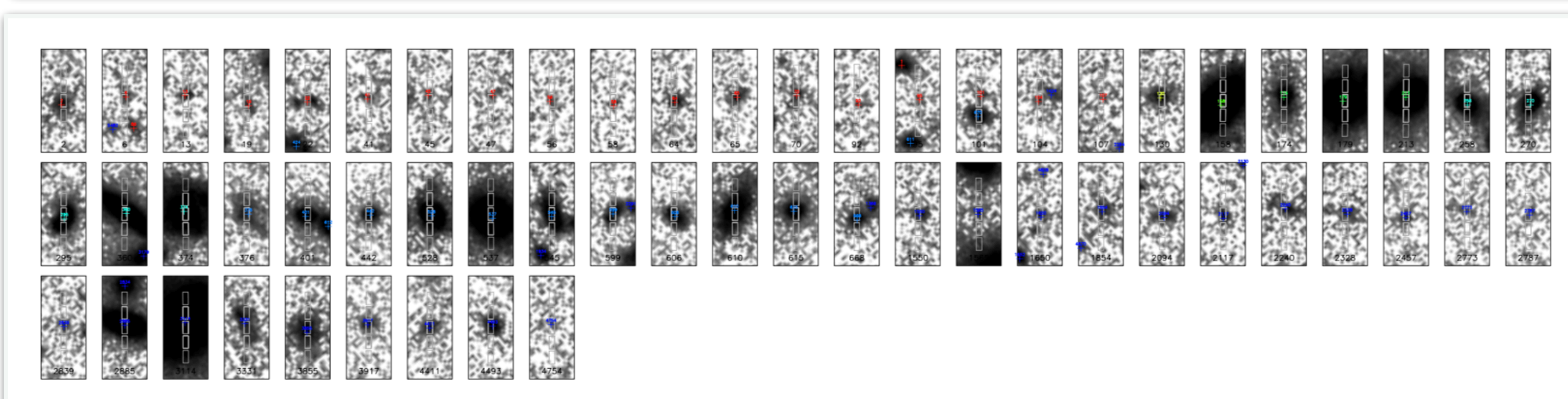
# No	No_sub	No_cat	Pr	RA[deg]	Dec[deg]
1	2	23225	1	215.0097200	52.9813920
2	6	39385	1	215.0127900	52.9933350
3	13	13957	1	215.0533800	52.9648880
4	19	25075	1	215.0206100	52.9988630
5	22	25274	1	214.9940400	52.9808540
6	41	38390	1	214.9952800	52.9634200
7	45	40073	1	214.9982800	52.9947500
8	47	38442	1	215.0161300	52.9790870
9	56	35381	1	215.0770200	52.9695030
10	58	37527	1	215.0144500	52.9624880
11	64	11628	1	215.0814400	52.9721810
12	65	23472	1	215.0261000	52.9945260
13	70	30136	1	215.0108600	53.0133370
14	92	36789	1	215.0791600	52.9957480
15	95	37490	1	215.0306100	52.9732140
16	101	100383	1	215.0682100	52.9621200
17	104	100540	1	215.0607400	52.9587060
18	107	100567	1	215.0499600	52.9799000
19	130	27431	3	215.0071490	53.0012440
20	158	26912	4	215.0085300	52.9973350
21	174	16946	5	215.0864100	53.0024010
22	179	27509	5	215.0233300	53.0102050
23	213	18896	5	215.0848700	53.0097330
24	258	23319	6	215.0565600	53.0135100
25	270	12250	6	215.0670600	52.9645840
26	295	31171	6	214.9962200	52.9991460
27	360	22944	6	215.0623900	53.0134030
28	374	22339	6	215.0192900	52.9805450
29	376	21244	7	215.0558300	53.0030460
30	401	18650	7	215.0716500	53.0014860
31	442	23460	7	215.0540500	53.0138490
32	528	12973	7	215.0757200	52.9737270
33	537	23063	7	215.0057300	52.9753440
34	545	15504	7	215.0616000	52.9775820
35	599	17281	7	215.0741300	52.9965500
36	606	18782	7	215.0778000	53.0065130
37	610	18631	7	215.0179200	52.9629530
38	615	18489	7	215.0728000	53.0016450
39	668	11532	7	215.0832300	52.9728630
40	1550	12762	8	215.0696160	52.9699010
41	1565	12870	8	215.0598760	52.9636300
42	1650	13518	8	215.0492190	52.9596430
43	1854	14755	8	215.0640010	52.9764580
44	2094	16133	8	215.0288020	52.9587870
45	2117	16261	8	215.0275990	52.9586530
46	2240	16884	8	215.0586740	52.9834450
47	2328	17400	8	215.0226220	52.9612300
48	2457	18218	8	215.0658050	52.9959780
49	2773	20010	8	215.0027860	52.9605410
50	2787	20140	8	215.0254660	52.9772050
51	2839	20422	8	215.0106130	52.9680430
52	2885	20671	8	215.0153870	52.9696920
53	3114	21831	8	215.0038570	52.9649190
54	3331	23034	8	215.0005790	52.9732120
55	3855	26396	8	215.0063970	52.9947950
56	3917	26816	8	215.0290780	53.0139650
57	4411	31014	8	214.9973540	52.9991640
58	4493	31646	8	215.0042630	53.0027470
59	4754	35759	8	215.0839320	52.9811100

Fortran Configuration file

m_check module output

```
# ----- SECTION AUTOMATICALLY APPENDED BY THE M_CHECK MODULE ----------
#
# Warning - The following included targets were flagged as contaminated
# in one or more other pointings:
#
# Potentially Contaminated Targets in Pointing_1:
#   ID_sub ID_cat
#     101   100383
#
# Refer to the close-up slitlet proximity maps in:
#
#  ./trial_06/m_check_output/pointing_1/slitlet_panel_plot.ps
#
# to gauge the contamination levels of each target at each final pointing.
#
# Targets deemed to be excessively contaminated may be excluded from observation by
# changing the sign of their Priority Class assignments to be negative in the
# input catalog, and running the ipa and subsequent modules anew.
```





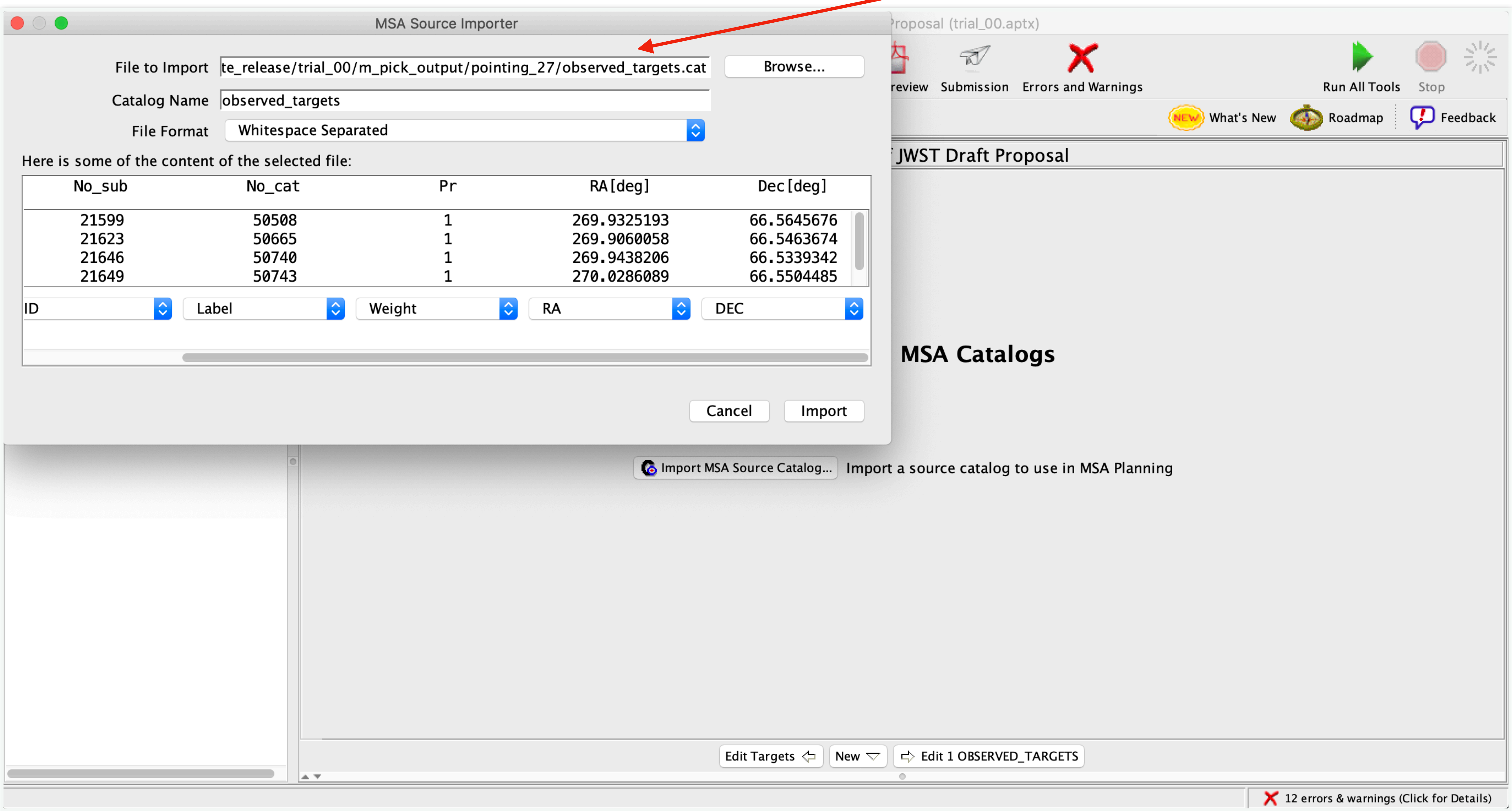
Template Python script for running eMPT in batch or interactive mode

```
# --------------------------------------------------------------------------------------------------------------------------------------------
#
# Example 'for' loop that explores two different nominal (i.e. starting) (Ra,Dec) locations around which to search for optimal pointings, at 5
# different roll angles. Edit the parameter names accordingly, add or subtract a nested loop, etc., to loop through your eMPT parameters in
# which you are interested in a similar fashion, to explore the eMPT parameter space and produce the most optimal MSA masks for your science program.
#
# --------------------------------------------------------------------------------------------------------------------------------------------

user_pars = {}
trial_map={}


test_nom_pos = [(53.14187071428572,-27.809516857142857), (53.141582,-27.81504)]
test_pa_v3 = range(25,225,5)
test_pa_ap = np.array([float(p) for p in test_pa_v3]) + 138.492


ntrial=1        #00:99 allowed
ntrial_str="01"


for pos in test_nom_pos:

  cra,cdec= pos[0],pos[1]

  for i,pa in enumerate(test_pa_ap):

    if ntrial<10:
      ntrial_str = "0"+str(ntrial)

    else:
      ntrial_str=str(ntrial)

    trial_map[ntrial] = (cra,cdec,pa)

    user_pars['cra']    = str(cra)
    user_pars['cdec']   = str(cdec)
    user_pars['cpa_ap']  = str(pa)
    user_pars['n_trial'] = ntrial_str

    updated_config = edit_config_file(confname,ntrial_str,parse_config_file(confname)[0], parse_config_file(confname)[1], parse_config_file(confname)[2],     user_pars)

    print("\nStarting trial "+ntrial_str+" with RA, Dec, PA_AP: "+str(trial_map[ntrial]))

    for mod in  ["ipa","k_make","k_clean","m_make","m_sort","m_pick","m_check","m_check_regions"]:
      execute(mod, confname=updated_config)

    if ntrial<99:
      Warning("Batch runs may not exceed 100 trials. Quitting after next trial finishes.")

    else:
       raise ValueError("Quitting...batch runs may not exceed 100 trials.")

    ntrial+=1


for k in trial_map:
   print("trial "+str(k)+": "+str(trial_map[k])+" PA_AP,RA,Dec")
```

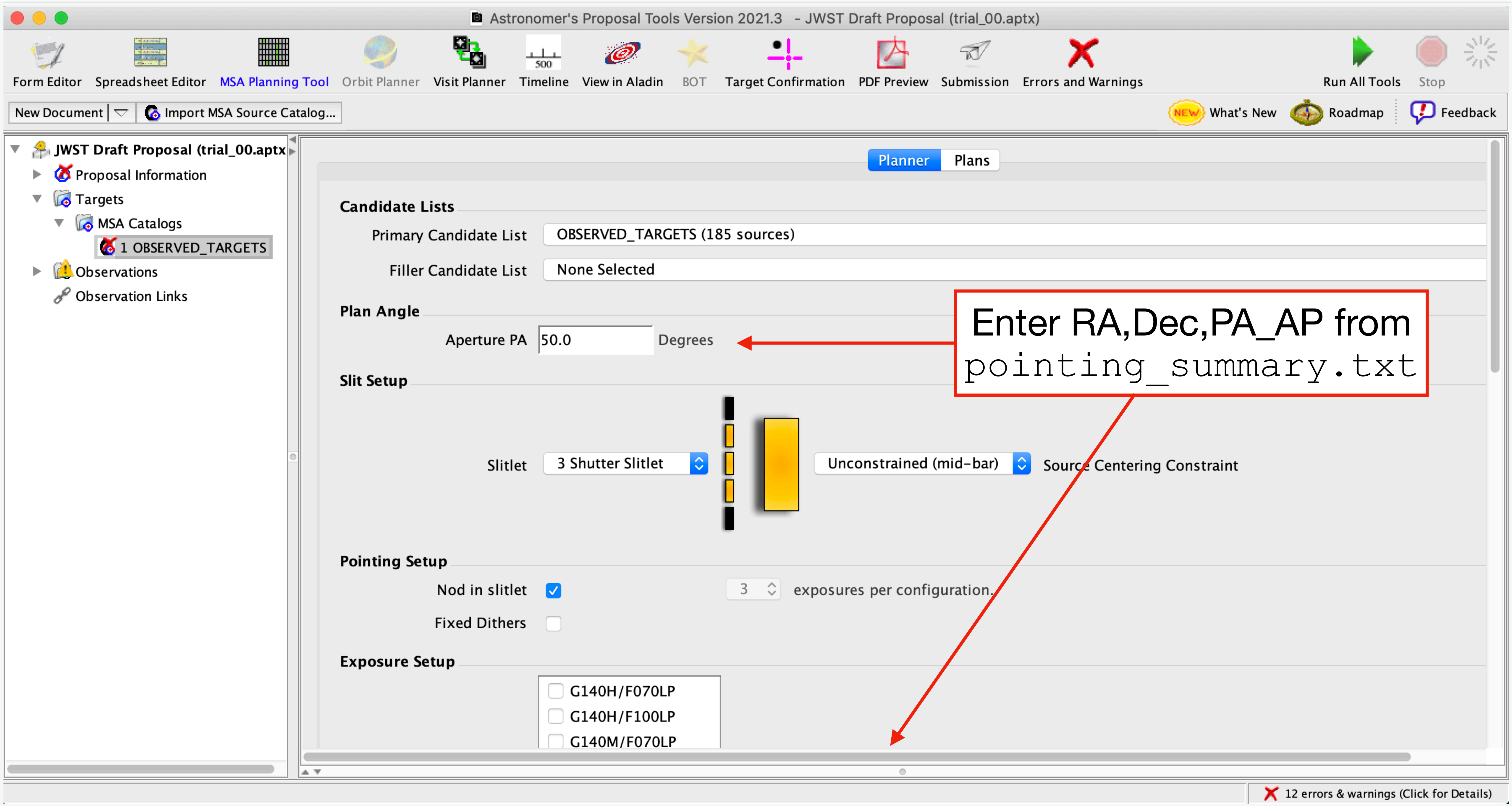
Interfacing with the STScI APT/MPT

Targets -> Import MSA Source Catalog

`observed_targets.cat`

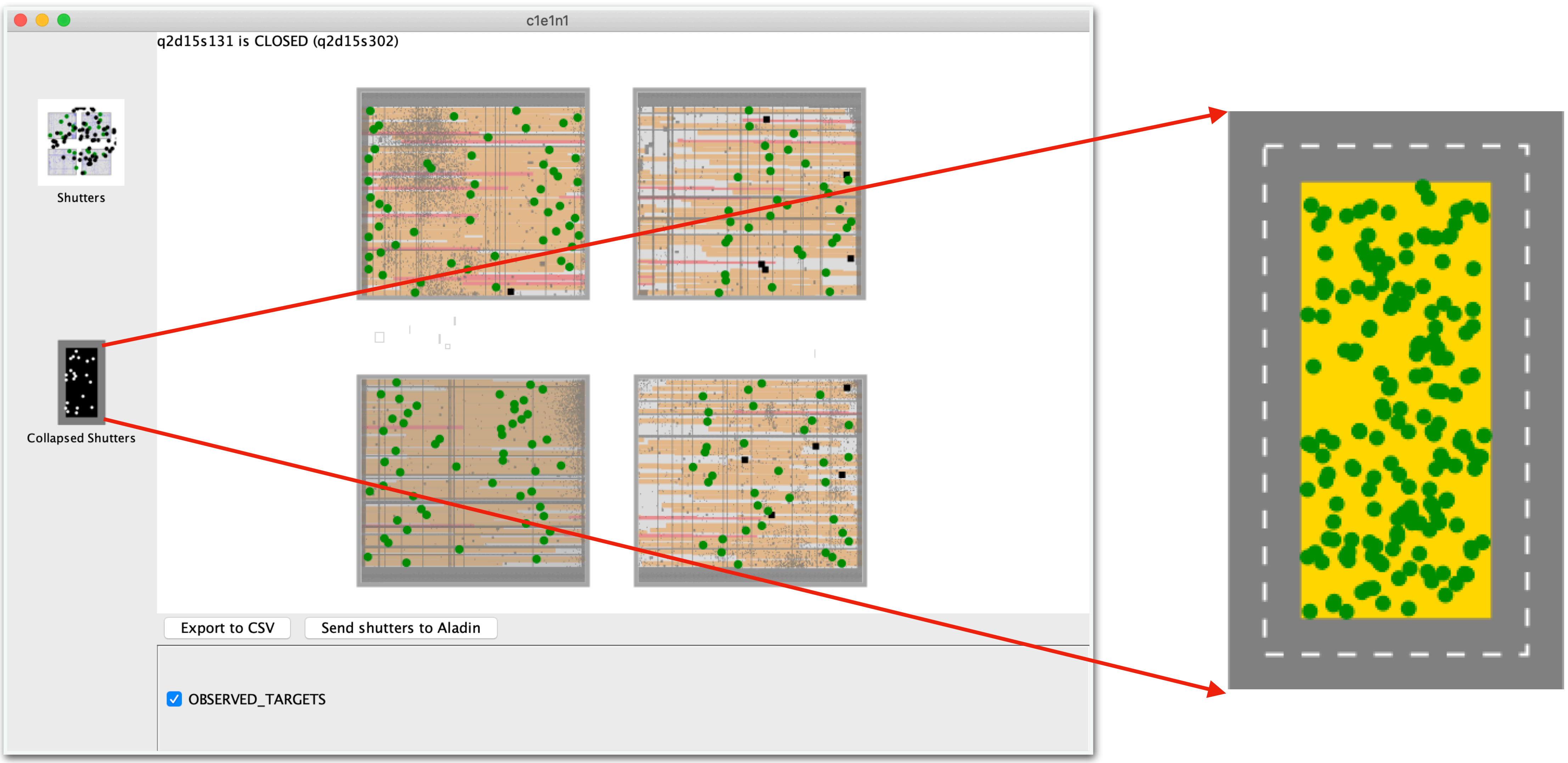


Interfacing with the STScI APT/MPT



Interfacing with the STScI APT/MPT

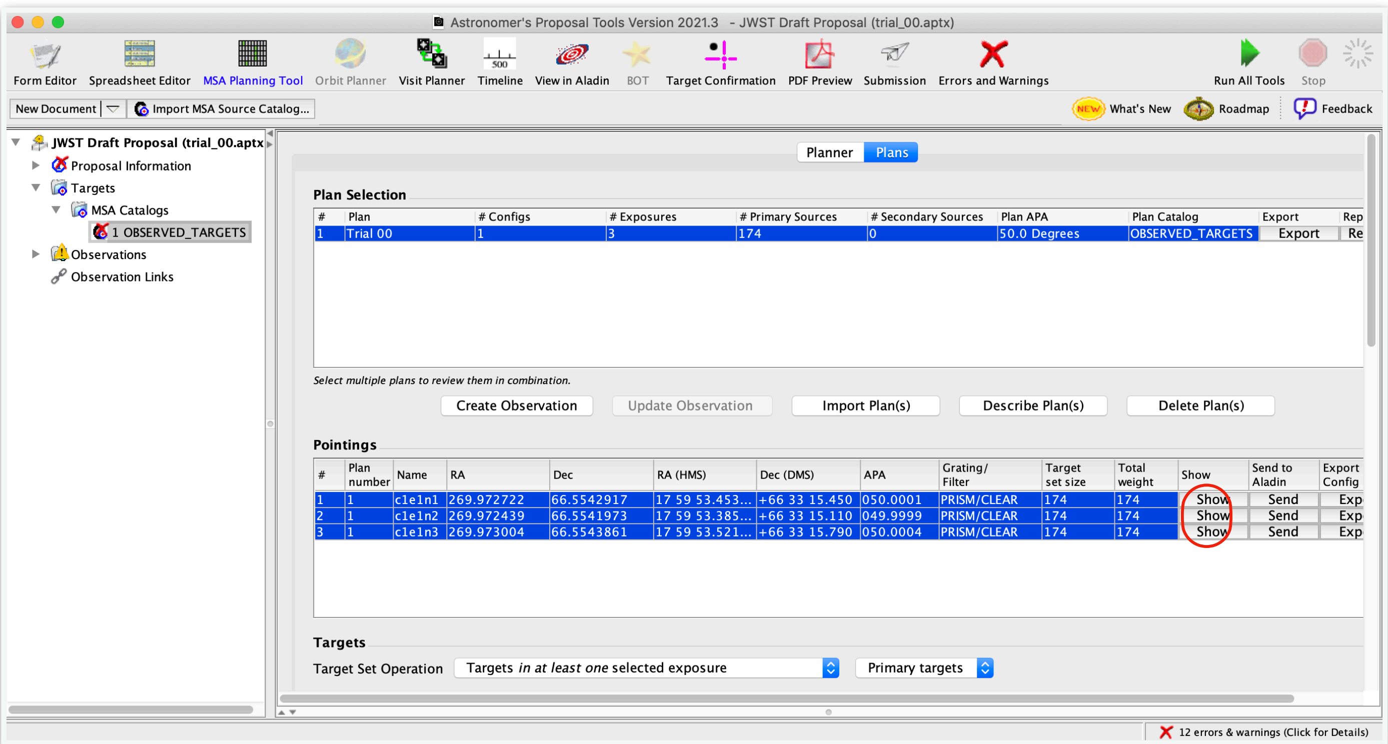
The resulting ‘MPT view’ of the eMPT-optimized pointing configuration.

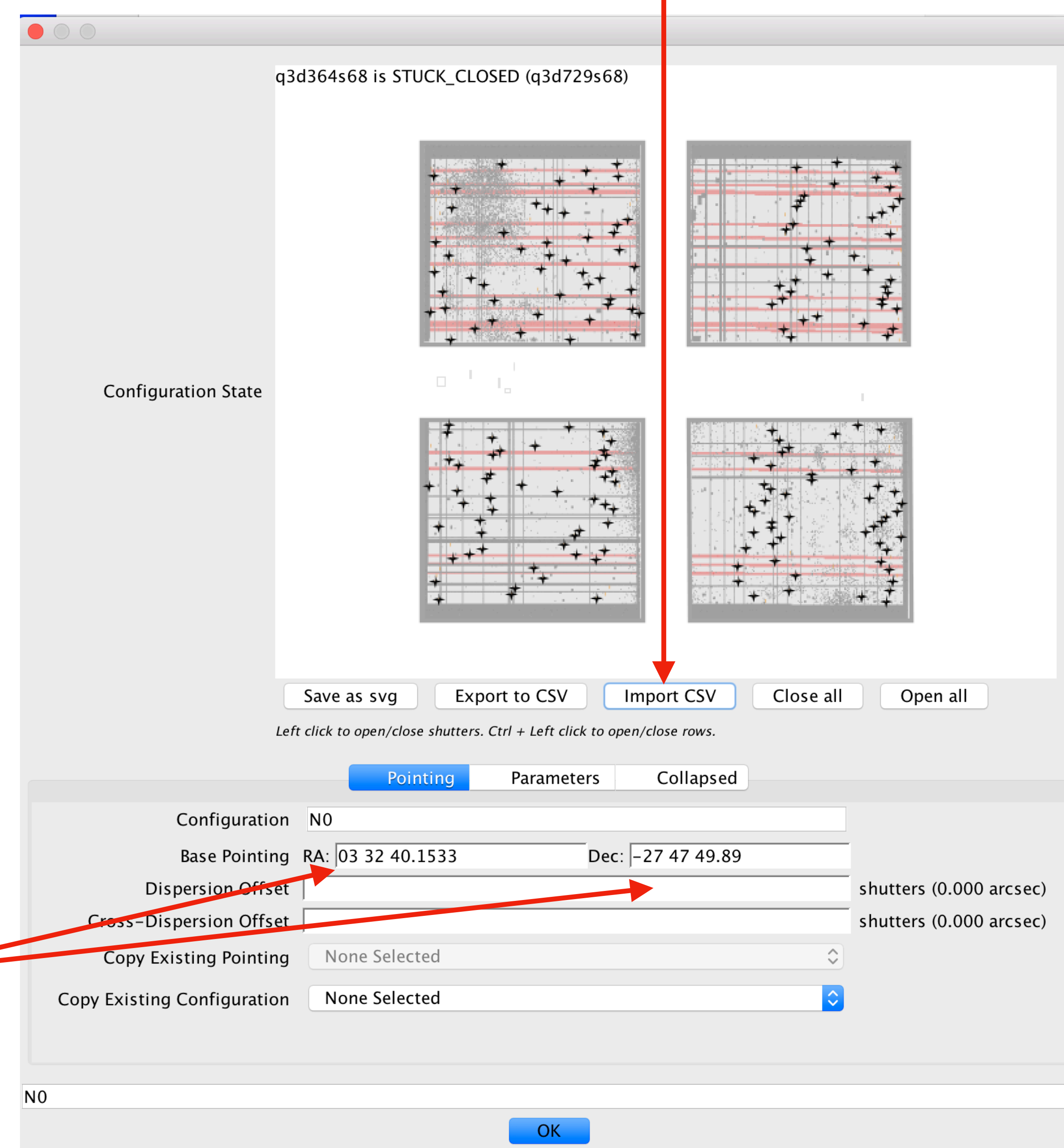


To utilize the *exact* slit mask generated by the eMPT and override the MPT version, do this →

Interfacing with the STScI APT/MPT

Import
`shutter_mask.csv`





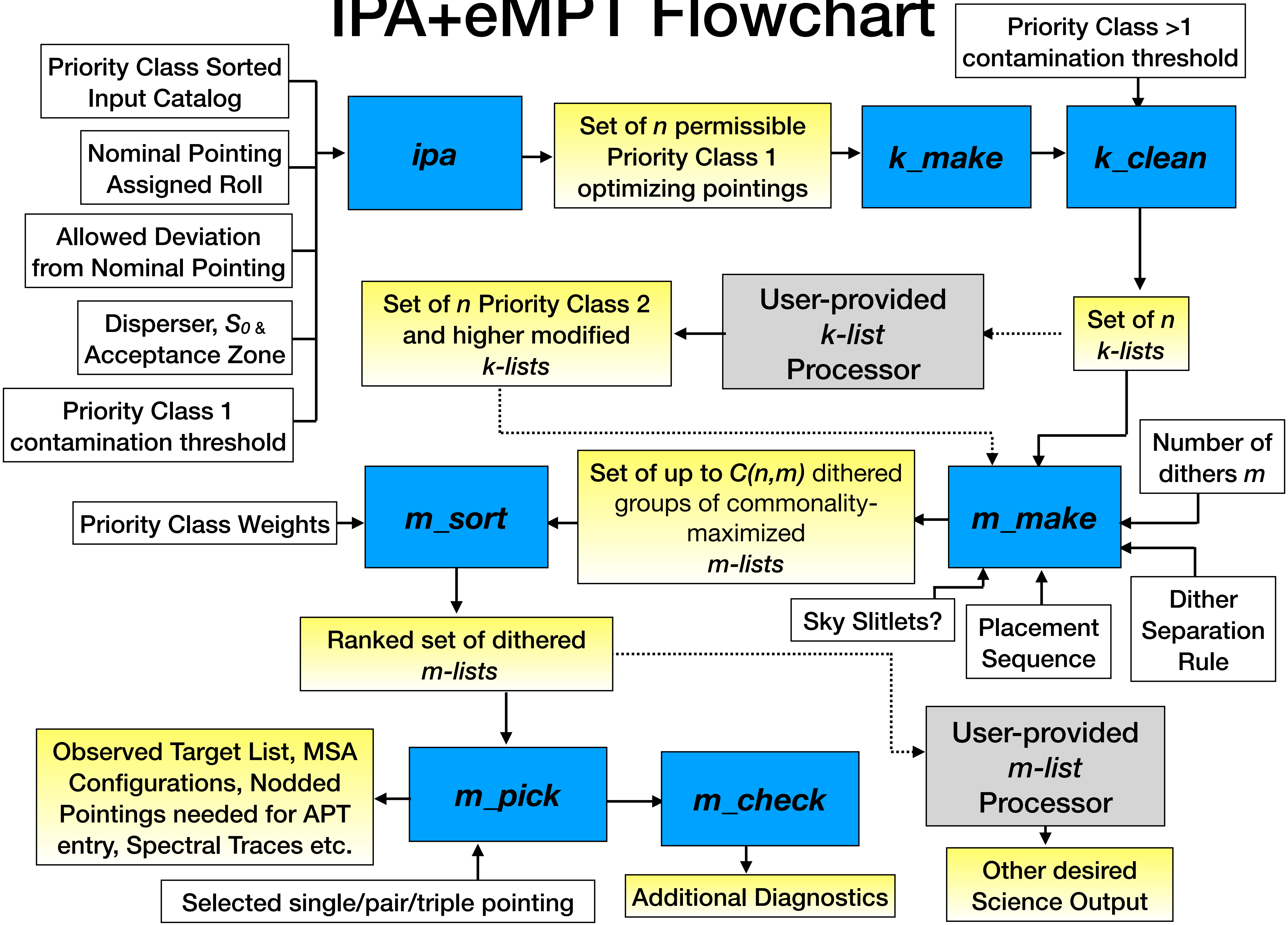
Enter RA,Dec from
`pointing_summary.txt`

Interfacing with the STScI APT/MPT

The imported and ‘forced’ eMPT slit configuration recovers those of its targets that got lost in the MPT view of the same configuration.

User Inputs

IPA+eMPT Flowchart



Credit: P. Jakobsen

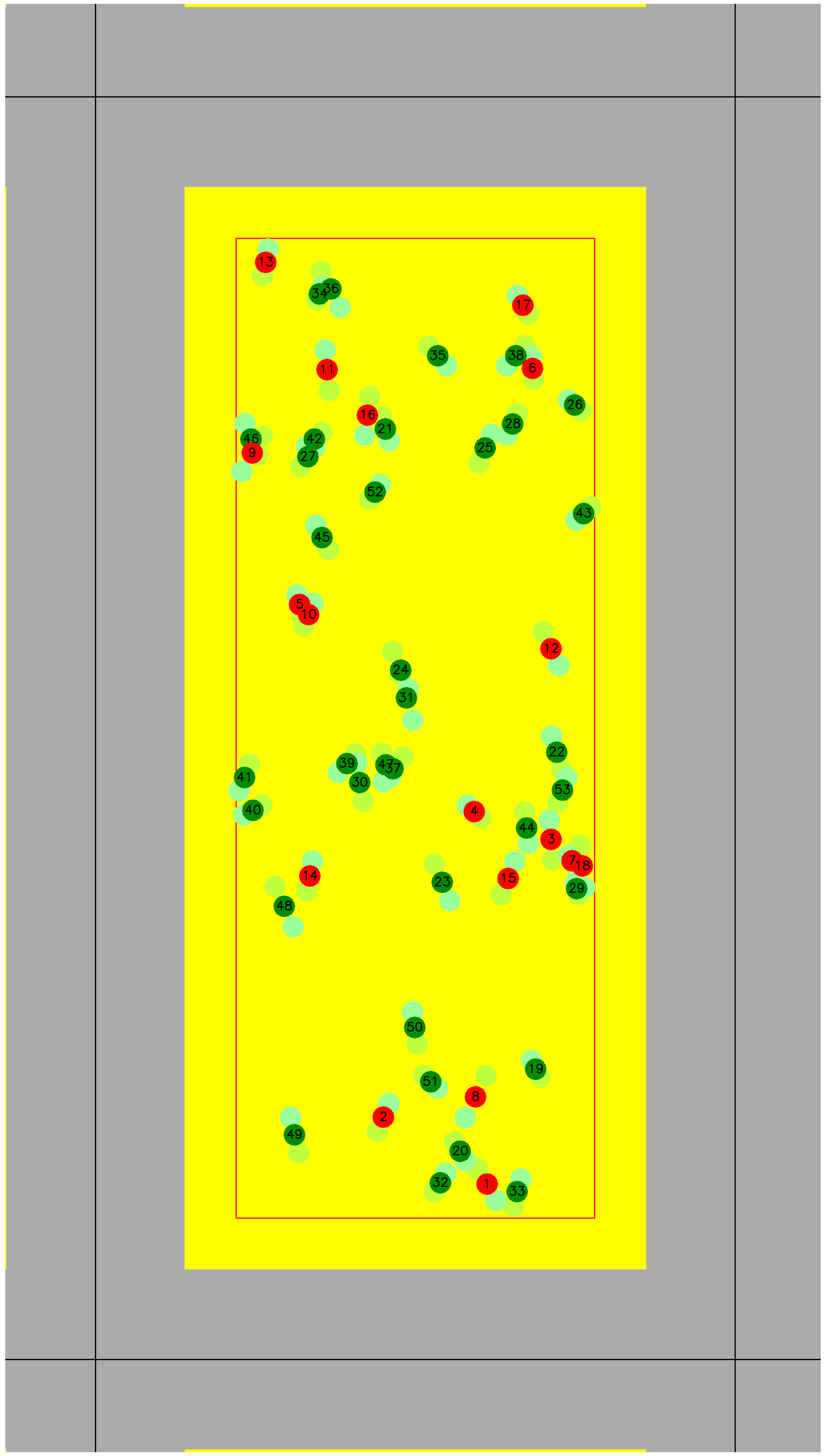
The Input Target Catalog & Target Priority Classes

- The IPA+eMPT target *Input Catalog* nominally lists all science targets of interest to which the NIRSpec observer has first rights, that are situated within a 3-arcminute radius of the nominal pointing of the observation called out in the proposal.
- The Input Catalog can also contain stars and other relevant foreground objects that may be potential sources of contamination.
- All targets in the Input Catalog must be assigned a *Priority Class* (a maximum of *n_class* ≤ 20 Priority Classes may be used)
- Priority Class 1 targets are *special* and treated differently
- Priority Class 0 is reserved for stars and other foreground objects
- The Priority Classes > 1 assigned in the Input Catalog mainly serve to group the targets into similar types of object and need not necessarily reflect the relative scientific importance of each group
- The prioritized order in which targets of different Priority Class are attempted observed is specified by the user separately.

Credit: P. Jakobsen

The Acceptance Zone

- The user-selectable parameters *acx,acy* determine how well accepted targets need to be aligned within the centers of their slitlets
- Measured in units of shutter facets in eMPT
- Full facet corresponds to *acx=0.500, acy=0.500*
- Default is *acx=0.343, acy=0.420* ('shaved' full open area)
- STScI MPT allows five fixed choices for the Acceptance Zone

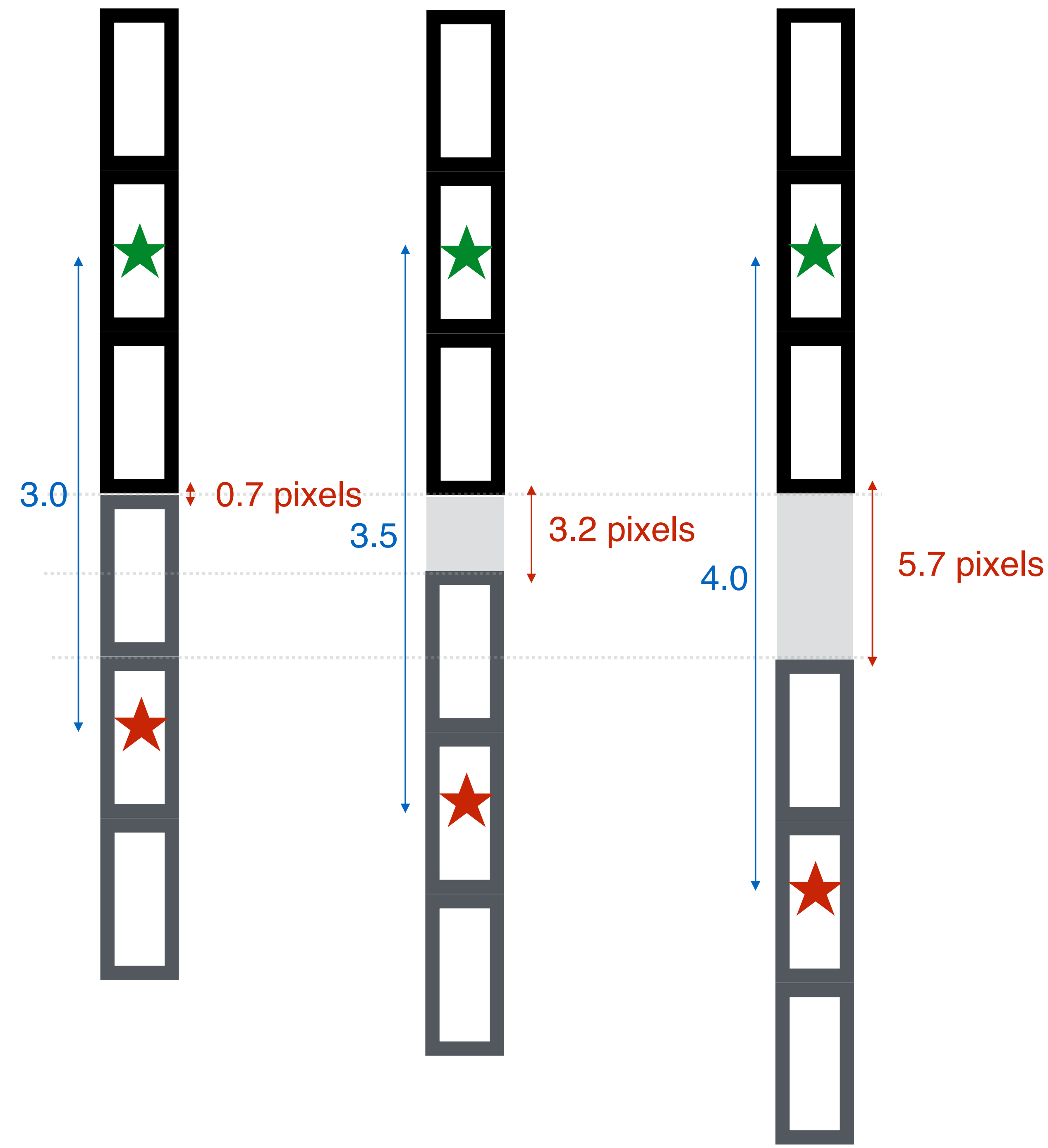


Credit: P. Jakobsen

Minimum Vertical Spectral Separation

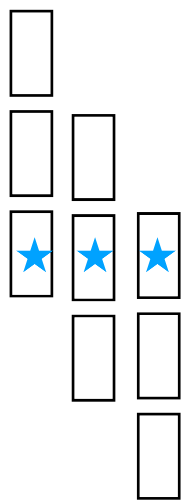
User-selectable parameter S_0 sets the minimum allowed vertical central shutter separation between two target spectra.

- S_0 = 3.5 recommended baseline value for eMPT;
- equivalent to ~3.2 (geometrically) un-illuminated pixels between spectra;
- STScI currently employs S_0 = 3.2 (hardwired);
- equivalent to ~1.7 (geometrically) un-illuminated pixels between spectra

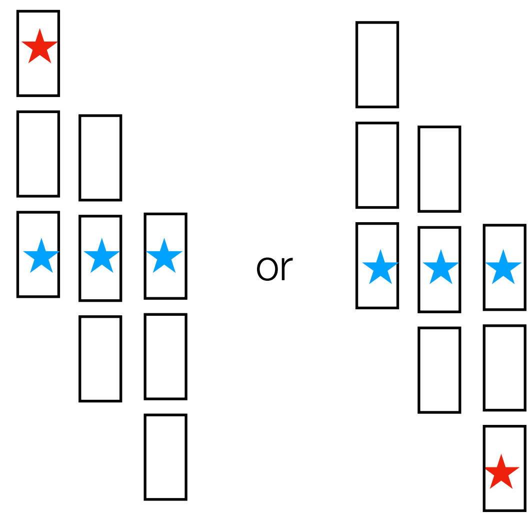


Credit: P. Jakobsen

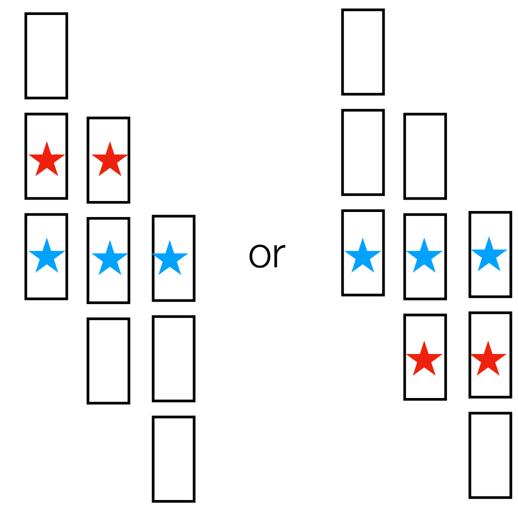
Contamination Threshold



Score=0

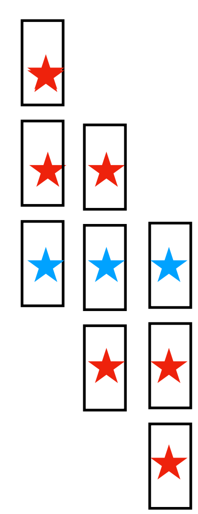


Score=1

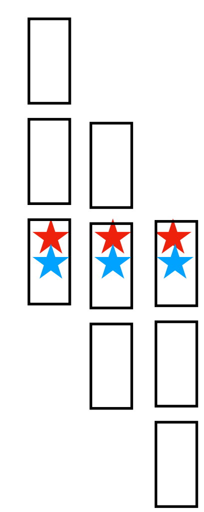


Score=2

Contamination scores from multiple contaminating objects *add* to give total score



Score=6



Score=6

Credit: P. Jakobsen

Dither options

n_dither = 1: The eMPT aims to find the *single* pointing among those identified by the ***ipa*** module having the ‘best’ observed target sample

n_dither = 2: The eMPT aims to find the *pair* of pointings among those identified by the ***ipa*** module yielding the ‘best’ *combined* observed target sample

n_dither = 3: The eMPT aims to find the *triple* of pointings among those identified by the ***ipa*** module yielding the ‘best’ *combined* observed target sample

Number of possible pointing pairs/triples that ***m_make*** needs to explore balloons rapidly with the number of pointings
⇒ *Need to be smart about it*

Credit: P. Jakobsen

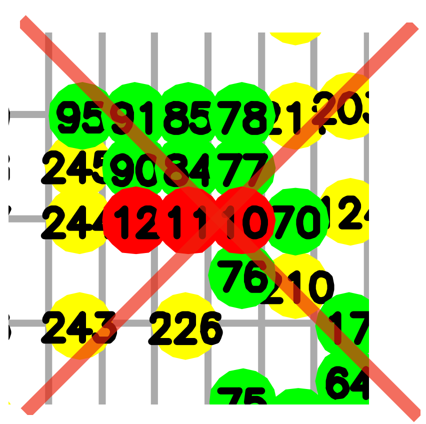
Dither options

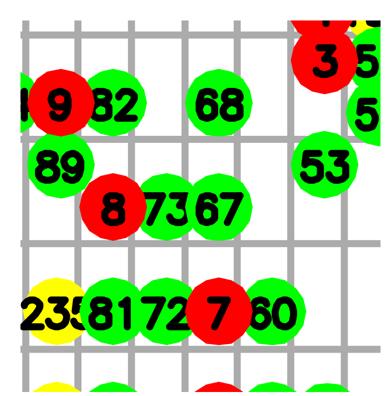
Parameters intended to *force* separation between dithered pointings:

min_dx, max_dx, min_dy, max_dy

Dither Separation Rules:

1. Legal dithers must be separated by $\Delta i \leq max\_dx$ in *X*
2. Legal dithers must be separated by $\Delta j \leq max\_dy$ in *Y*
3. If any two dithers are closer together than $\Delta j < min\_dy$ in *Y*, then they must be separated by $\Delta i > min\_dx$ in *X* to be legal





Serves to constrain the pointing pairs/triples to be considered 'reasonable' dithers on detector

Credit: P. Jakobsen

The placement sequence matrix

Ordering of the integer sequence 1, 2, .. *n_class x n_dither* into a matrix having *n_class* columns and *n_dither* rows

Determines the *order* in which the natural subsets of targets making up the *k-list* are attempted placed on the detector by the *Matrix algorithm* – and thereby the likelihood of the targets in each subset actually being observed

The placement sequence matrix is a very powerful and flexible tool that determines much of the overall behavior of the eMPT

It is the user's responsibility to wield this very sharp tool responsibly..

Credit: P. Jakobsen

The placement sequence matrix

Examples:

n_dither = 1, n_class = 5:

Priority Class

1	2	3	4	5

Targets placed in strict order of Priority Class, but can be arbitrarily reordered if desired (within reason!)

n_dither = 2, n_class = 5:

Priority Class

Common Targets	1	2	3	4	5
Unique Targets	6	7	8	9	10

Targets placed to maximize commonality between the pointing pairs in strict order of Priority Class (i.e. to achieve maximum exposure time per observed target)

Priority Class

Common Targets	6	7	8	9	10
Unique Targets	1	2	3	4	5

Targets placed to *minimize* commonality between the pointing pairs in strict order of Priority Class (i.e. achieve a combined target sample containing the largest number of targets)

Credit: P. Jakobsen

The placement sequence matrix

Examples:

n_dither = 3, n_class = 5:

	Priority Class				
Common Targets	1	2	3	4	5
Partially Common Targets	6	7	8	9	10
Unique Targets	11	12	13	14	15

Targets placed to maximize commonality between the pointing triples in strict order of Priority Class (i.e. to achieve maximum exposure time per observed target)

	Priority Class				
Common Targets	1	4	5	6	7
Partially Common Targets	2	8	9	10	11
Unique Targets	3	12	13	14	15

Targets placed to first maximize coverage of Priority Class 1 targets at any exposure level, and only then maximize commonality between the pointing triples of the remaining targets in strict order of Priority Class

An infinitude of possibilities… tread carefully!

Credit: P. Jakobsen

Figure of Merit Target Weighting

Purpose: Calculate a *Figure of Merit* from the *combined m-list* of each single/pair/triple pointing output by the ***m_make*** module

Assign *weights* to targets of each Priority Class: $w(i),\ i = 1, \ldots n_{class}$

FOM1 ~ Average *exposure time* per weighted target in combined target list

FOM2 ~ Total weighted *number* of targets in combined target list

n_{dither}=1:

$$FOM1 = 1$$

$$FOM2 = \sum_{i=1}^{n_{class}} w(i) n_u(i)$$

n_{dither}=2:

$$FOM1 = \frac{\sum_{i=1}^{n_{class}} w(i)(2n_c(i) + n_u(i))}{\sum_{i=1}^{n_{class}} w(i)(n_c(i) + n_u(i))}$$

$$FOM2 = \sum_{i=1}^{n_{class}} w(i)(n_c(i) + n_u(i))$$

n_{dither}=3:

$$FOM1 = \frac{\sum_{i=1}^{n_{class}} w(i)(3n_c(i) + 2n_p(i) + n_u(i))}{\sum_{i=1}^{n_{class}} w(i)(n_c(i) + n_p(i) + n_u(i))}$$

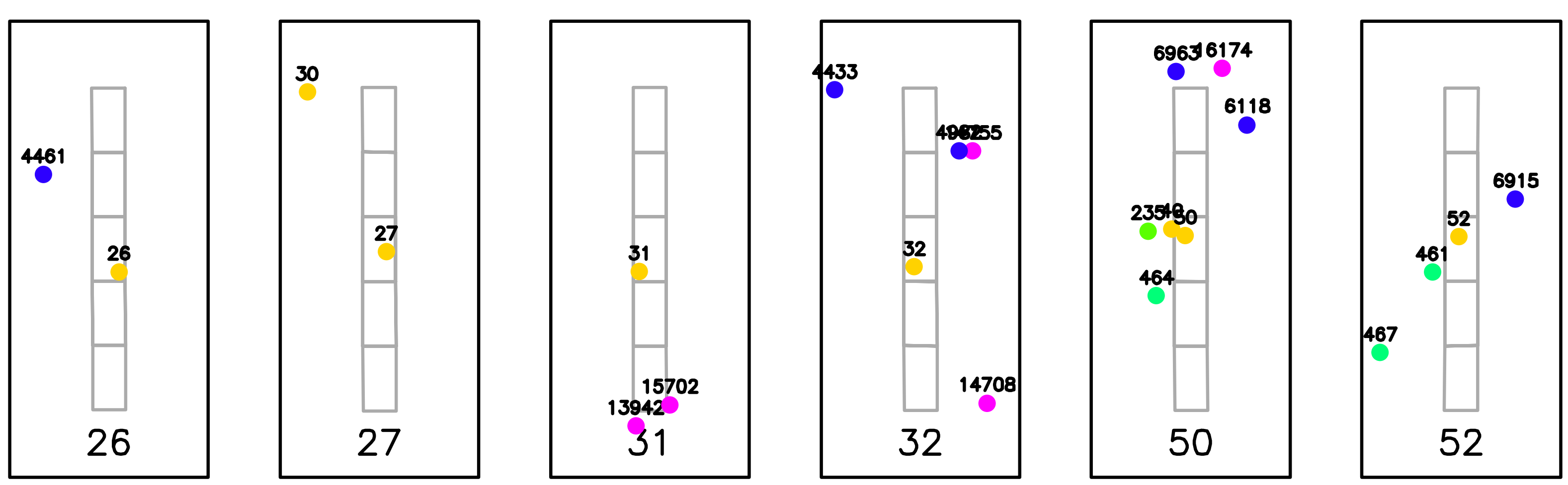
$$FOM2 = \sum_{i=1}^{n_{class}} w(i)(n_c(i) + n_p(i) + n_u(i))$$

Credit: P. Jakobsen

The *m_check* module

- Purpose: Generate a ‘rogue’s gallery’ of 1.6” x 3.7” closeups of each observed target’s nodded slitlet showing the detailed locations of the targets within their slitlets and the locations of any other nearby potentially contaminating objects

- The ***m_check*** module draws attention to any observed targets that may in reality still be contaminated on the basis of their having been previously classified as such at a different pointing by the ***k_clean*** module



Credit: P. Jakobsen