Spectral line imaging

Brian Svoboda (NRAO)

Outline

- Calibration
  - continuum subtraction methods
- Imaging
  - `tclean` for cube imaging
  - `multiscale` CLEAN
  - automasking with `auto-multithresh`
- Line analysis
  - Moments
  - Line fitting
  - PV diagrams

Continuum subtraction

Spectral analysis and visualization methods often require subtracting the continuum contribution to the flux, leaving just the spectral lines emission. This can be done in CASA via three main strategies:

- **uvcontsub** – performs continuum fitting and subtraction in the *uv*-domain. The continuum fit degrades with distance from the phase center and may not be suitable for multiple bright sources.
- **imcontsub** – estimate and subtracts continuum emission in the image domain. Requires deconvolving the continuum plus lines in one cube.
- **uvsub** – Image continuum, save model, and subtract the resulting model directly from the visibilities. Most accurate method.

Note that while the `*contsub` tasks produce continuum fits, continuum images should be made through multi-frequency synthesis CLEAN on the line-free channels.

Continuum subtraction: trade-offs

All methods require some trade-off in terms of accuracy versus convenience. The CASA notebook "UV Continuum Subtraction" provides in-depth discussion on the topic.

- Subtraction in the uv-domain is desirable if continuum emission dominates the source: deconvolution of the line emission will be more robust if it is not subject to the deconvolution errors of the brighter continuum.
- There is a also a performance penalty to deconvolving each channel's continuum contribution in a cube.
- However, interpolating visibilities between channels is only a good approximation for emission near the phase center.
- Sault (1994) describes uv-domain based algorithms in detail.

Continuum subtraction: trade-offs

All methods require some trade-off in terms of accuracy versus convenience. The CASA notebook "UV Continuum Subtraction" provides in-depth discussion on the topic.

Use `uvcontsub` if emission is bright, compact, and mostly nearly the phase center.

Use `imcontsub` if emission is extended across the primary beam and is relatively weak.

Generate a continuum model and subtract it with `uvsub` if bright, complex continuum is present. If self-calibrating the continuum, this method comes nearly for free.

Continuum subtraction: uvcontsub method

The `uvcontsub` task is straightforward and efficient. It should generally be attempted first. If baseline continuum artifacts are present, then the strategy can be re-evaluated.

```
uvcontsub(
    vis='my.ms',
    fitspw='0~1:13~46',
    excludechans=True,
)
# generates my.ms.contsub
```

As in other steps, the application is straightforward with the main effort going into specifying the line-free or line-containing channels in the `spw` selection parameter.

Note that `uvcontsub` task is undergoing a rewrite and the calling convention is likely to change over the next CASA release.

Continuum subtraction: uvcontsub method

The `uvcontsub` task is straightforward and efficient. It should generally be attempted first. If baseline continuum artifacts are present, then the strategy can be re-evaluated.

```
results = uvcontsub(
    vis='my.ms',
    outputvis='subbed.vis',
    fitspec='0:10~20;50~60',
)
```

As in other steps, the application is straightforward with the main effort going into specifying the line-free or line-containing channels in the `spw` selection parameter.

Note that `uvcontsub` task is undergoing a rewrite and the calling convention is likely to change over the next CASA release.

Continuum subtraction: imcontsub method

The `imcontsub` task generally gives good results if there is extended, relatively weak continuum emission in the field. Continuum emission will need to be deconvolved along with line emission during imaging.

```
imcontsub(
    imagename='my.image',
    linefile='line.image',
    contfile='cont.image',
    chans='10~20;50~60',
)
```

As this is an image based method, two new cubes are generated for the continuum-subtracted line image and fitted continuum image. The continuum is useful as a diagnostic of the fit results.

Continuum subtraction: uvsub method

The `uvsub` task is simple to apply if a continuum model is saved to the model column, as is done in self-calibration. A separate model image can also be used to populate the model column.

```
tclean(
    vis='my.ms',
    ...
    savemodel=True,
)

uvsub(vis='my.ms')
```

One can use the `split` task to create a second measurement set with the subtracted data.

The `reverse` parameter in `uvsub` can then be set to `True` to reverse the model subtraction in the original measurement set.

Spectral line imaging

Imaging and deconvolution in CASA is carried out in the tclean task. There are several considerations for imaging spectral line data:

- Spectral coordinates and reference frame
- Mapping between data channels and image channels
- Software Doppler tracking options

Spectral line imaging: coordinates

In spectral line imaging, the spectral coordinates are defined by the user inputs, adjusted through the `start`, `width`, `nchan`, and `outframe` parameters.

```
tclean(
    ...
    specmode='cube',
    start='-10km/s',
    width='',
    nchan=20,
    outframe='LSRK',
    veltype='radio',
)
```

The spectral coordinate parameters can be specified in channel number, velocity, or frequency.

The spectral reference frame definition controls the frame in the resulting image. Usually Local Standard of Rest Kinematic (LSRK) is the desired `outframe`.

Spectral line imaging: data and image chan.

During the gridding process, the tclean task makes a choice about how to map data channels (in topocentric frequency) to image channels. This is based on channelization encoded in the MS and user specified parameters.

- If the defined start channel is not a data channel boundary, the visibility data and weights are interpolated and then evaluated at the centers of the shifted frequency grid.
- When the image channels are wider than the data channels, visibilities and weights are gridded via multi-frequency synthesis (MFS) within each image channel.
- On-the-fly software Doppler tracking is applied before interpolation and binning choices are made.

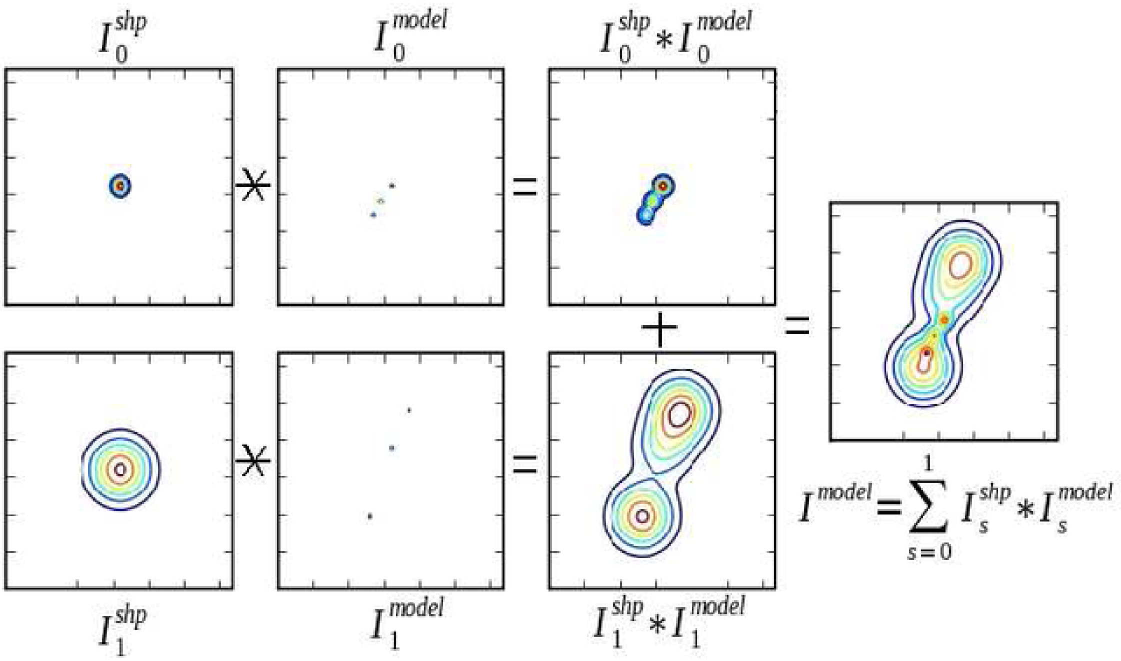
Spectral line imaging: Doppler tracking

VLA data are collected in the topocentric reference frame (that of the observatory). TOPO is a time dependent frame: sky frequencies of fixed astronomical sources will change with time. This means spectral lines will drift in frequency and occupy different data channels depending on the duration of the observation and frequency resolution.

- `specmode='cube'` – convert the resulting cube into the reference specified by the `outframe` parameter.
- `specmode='cubedata'` – do not apply Doppler tracking.
- `specmode='cubesource'` – for tracking moving Solar System sources. Frequencies will be in the source reference frame.

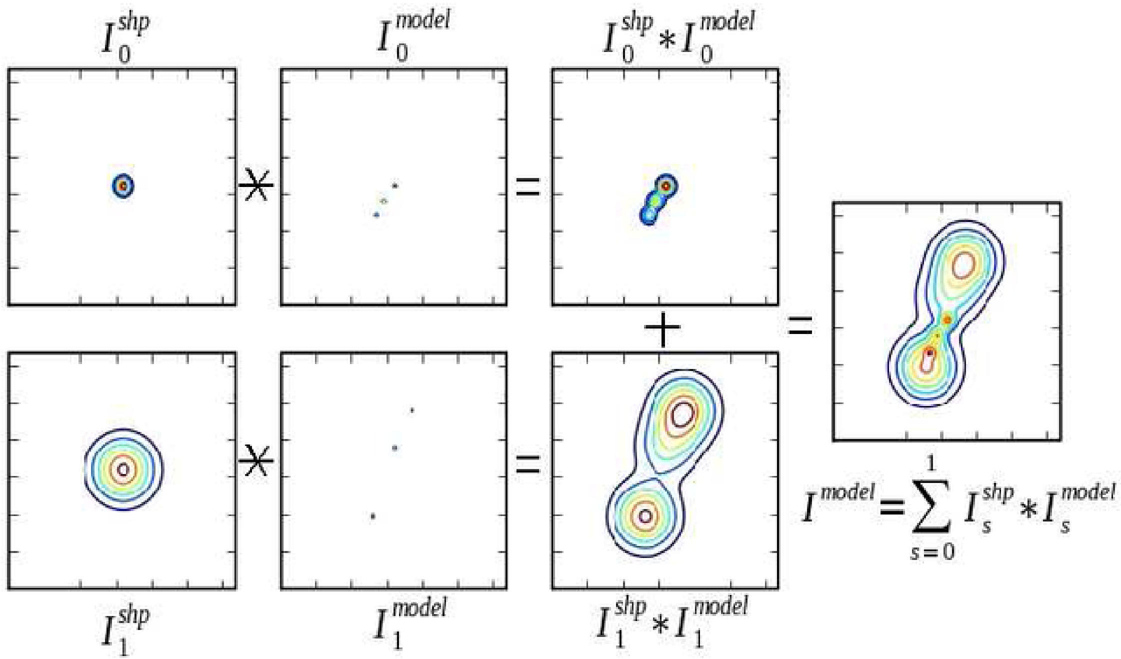
Imaging extended emission

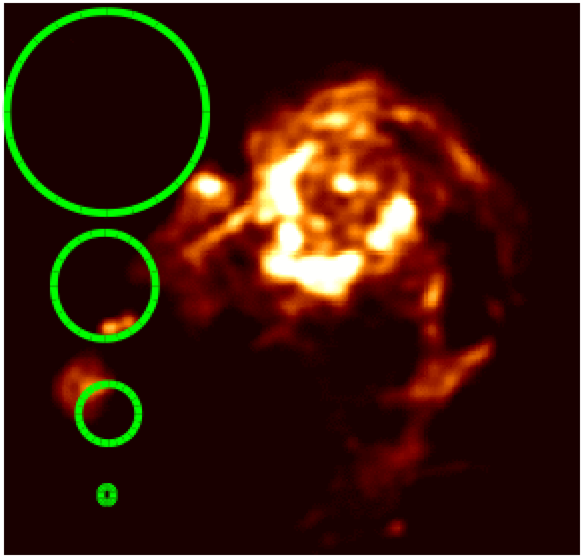
Although not unique to spectral line data, line emission is often spatially extended. Multiscale CLEAN and adaptive-scale pixel (ASP) CLEAN both provide better deconvolution of extended emission than the standard Cotton-Schwab/Hogbom CLEAN.



Imaging extended emission

Although not unique to spectral line data, line emission is often spatially extended. Multiscale CLEAN and adaptive-scale pixel (ASP) CLEAN both provide better deconvolution of extended emission than the standard Cotton-Schwab/Hogbom CLEAN.





Imaging extended emission

Although not unique to spectral line data, line emission is often spatially extended. Multiscale CLEAN and adaptive-scale pixel (ASP) CLEAN both provide better deconvolution of extended emission than the standard Cotton-Schwab/Hogbom CLEAN.

```
tclean(
    ...
    deconvolver='multiscale',
    scales=[0, 3, 9],
)
```

Described in Cornwell (2008), multiscale CLEAN uses Gaussian model components in addition to pixel size delta-functions to model extended emission.

The `scales` parameter defines the width of the component Gaussians in pixels. The 0 scale corresponds to a delta-function.

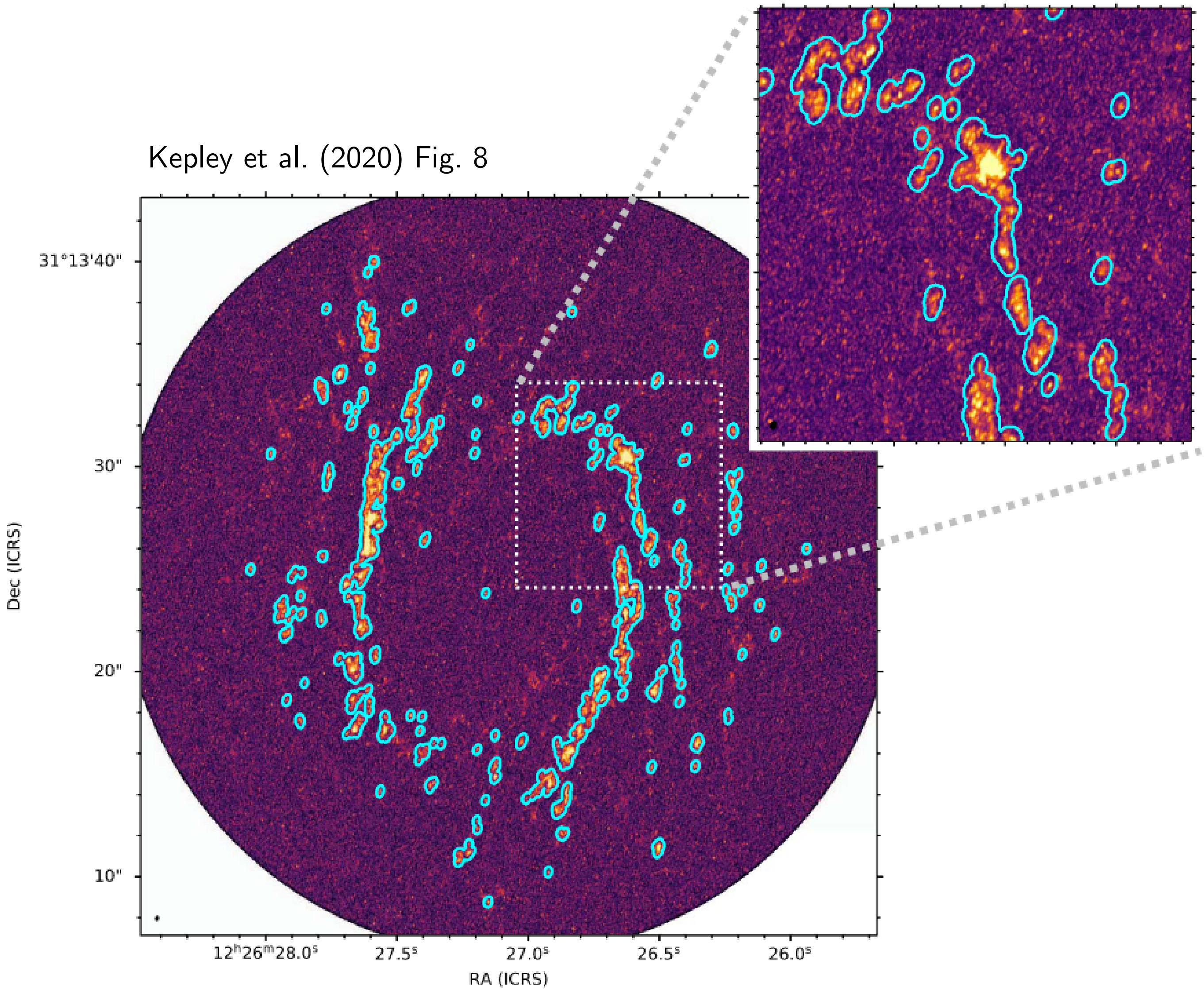
Imaging extended emission

Although not unique to spectral line data, line emission is often spatially extended. Multiscale CLEAN and adaptive-scale pixel (ASP) CLEAN both provide better deconvolution of extended emission than the standard Cotton-Schwab/Hogbom CLEAN.

```
tclean(
    ...
    deconvolver='asp',
)
```

Described in Bhatnagar & Cornwell (2004), ASP CLEAN adaptively selects the scale sizes and unlike multiscale CLEAN they do not need to be explicitly set.

Kepley et al. (2020) Fig. 8



Automasking with auto-multithresh

The auto-multithresh algorithm iteratively grows a CLEAN mask based on the SNR and sidelobe thresholds in the image. This method mimics the manual masking process of masking significant emission and increasing it encompass fainter emission as one CLEANs more deeply.

CASA Guide – https://casaguides.nrao.edu/index.php/Automasking_Guide

Kepley et al. (2020) – `DOI 10.1088/1538-3873/ab5e14`

Automasking with auto-multithresh

The auto-multithresh algorithm iteratively grows a CLEAN mask based on the SNR and sidelobe thresholds in the image. This method mimics the manual masking process of masking significant emission and increasing it encompass fainter emission as one CLEANs more deeply.

`usemask=''auto-multithresh''` – enable auto-multithresh

`noisethreshold` – SNR threshold above which emission is masked

`sidelobethreshold` – sidelobe threshold above which emission is masked

`minbeamfrac` – minimum region size as fraction of the beam size

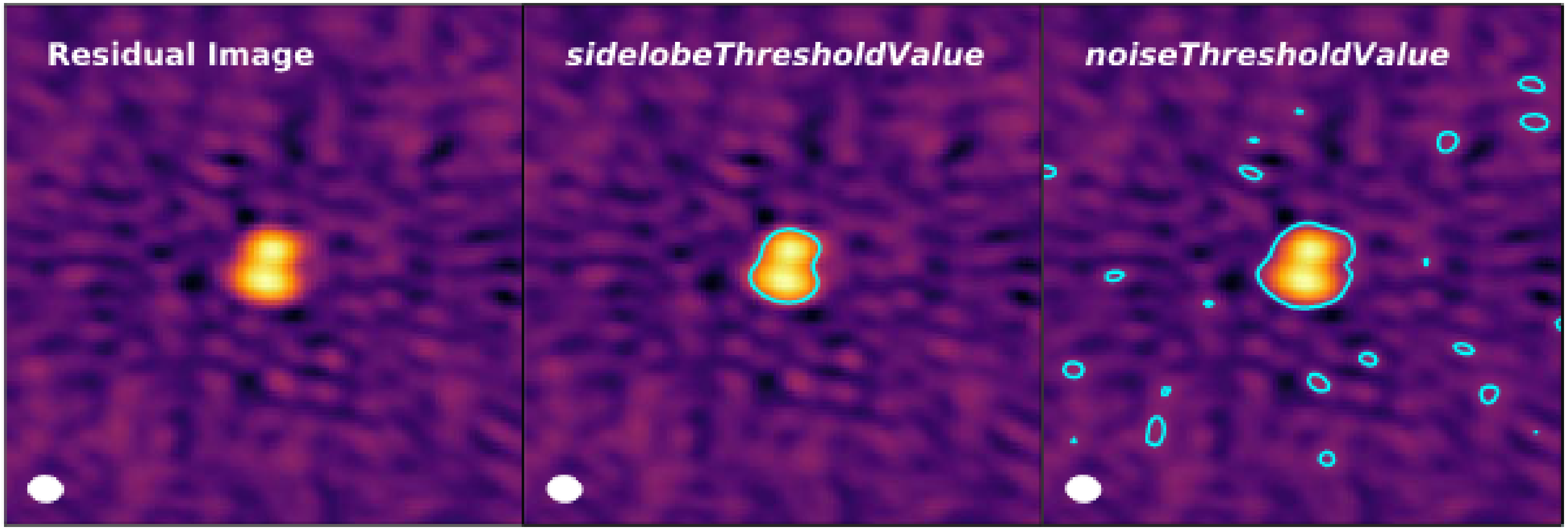
`lownoisethreshold` – SNR threshold that mask is grown down to

`negativethreshold` – SNR threshold for absorption features to be masked

See the automasking guide for further description of secondary parameters.

Automasking: multiple thresholds

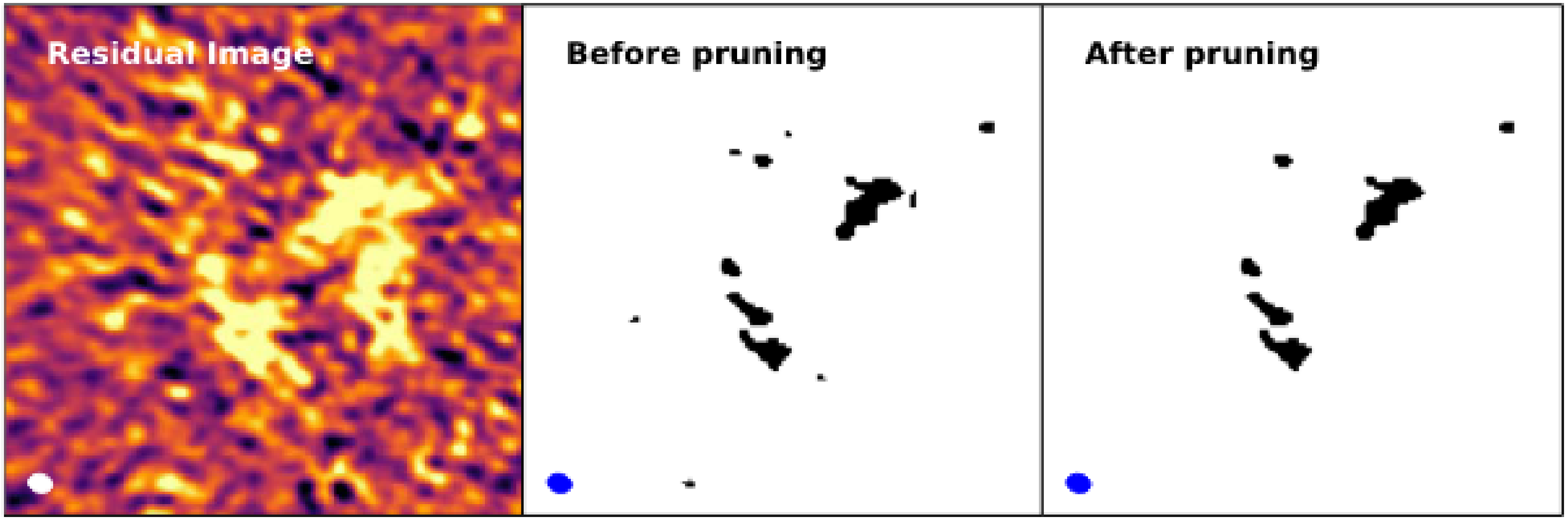
Auto-multithresh estimates two image thresholds and determines which is greater. The threshold based on the side-lobe level of the peak residual is likely to be greater initially. The second threshold, based on a multiple of the image RMS, is likely to be greater after partially CLEANing the image.



Kepley et al. (2020) Fig. 2

Automasking: pruning small regions

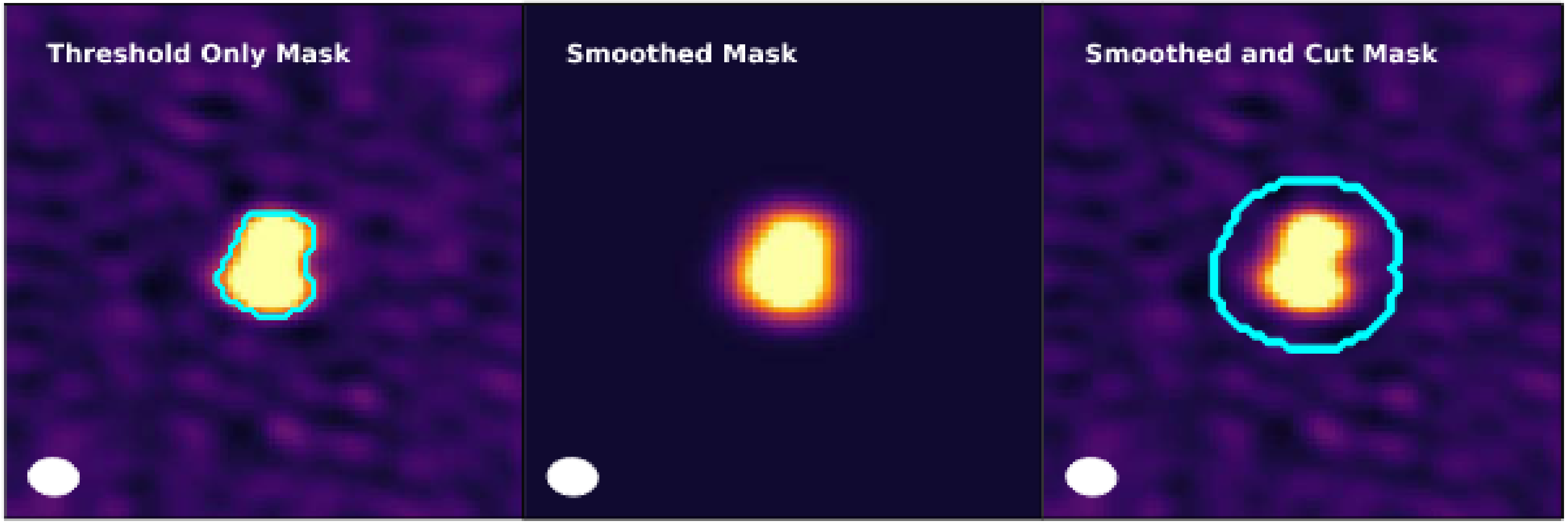
The noise-based threshold is likely to suffer from false positives by masking small noise spikes. Regions in the mask are pruned based on if they are smaller than a minimum area relative to the synthesized beam (in blue below). Typical values are 0.1 to 0.3.



Kepley et al. (2020) Fig. 3

Automasking: growing the mask

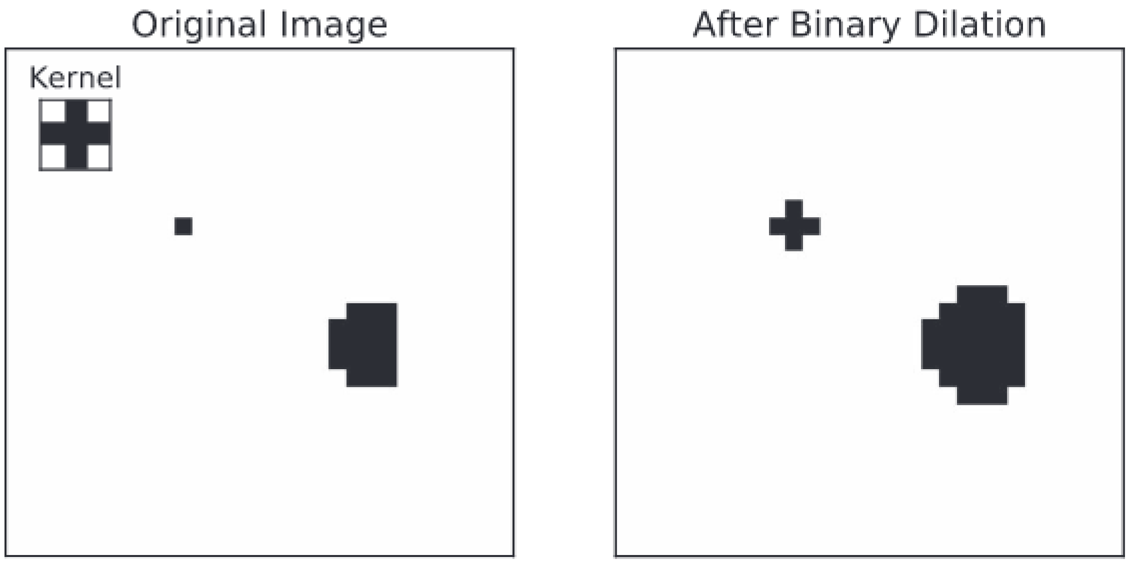
The selected and pruned mask is then smoothed and cut to extend partially beyond the initial masked region. These are secondary parameters that may need to be tuned in rare circumstances.



Kepley et al. (2020) Fig. 4

Automasking: growing the mask

The smoothed mask is then iteratively grown through a technique "binary dilation" in imaging processing. This process is similar to convolution and is used to grow the mask down to a configurable low-noise threshold. Typical values of the low-noise threshold may be one or two times the image RMS.

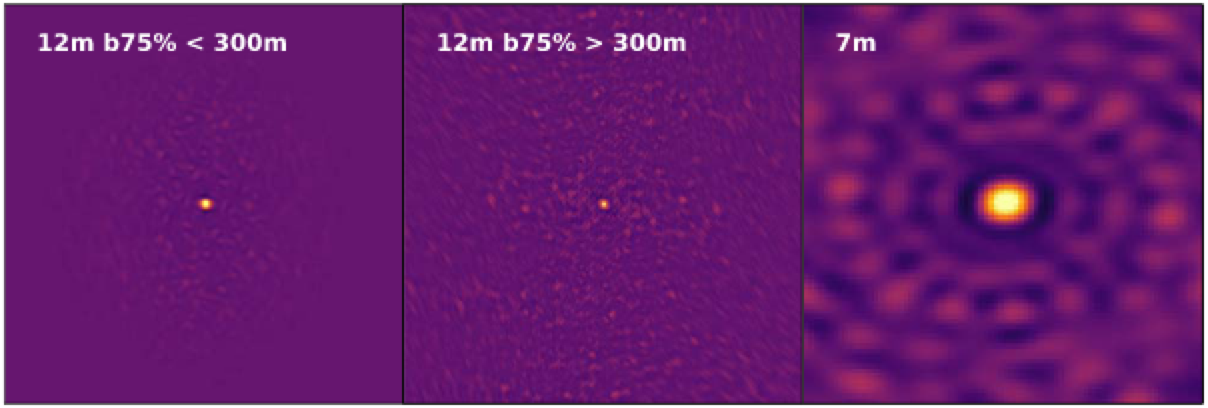


Kepley et al. (2020) Fig. 5

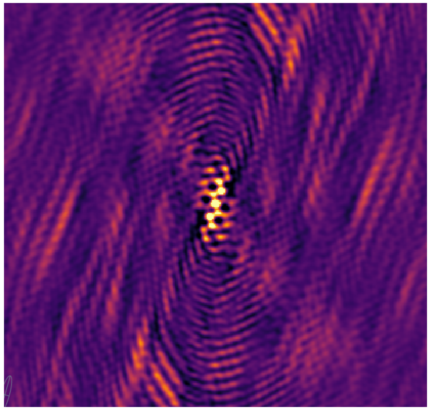
Automasking: parameter tuning

To achieve optimal results, the parameters for auto-multithresh need to be tuned. The parameters are particularly sensitive to the PSF.

ALMA short baselines, long-baselines, and ACA 7m array



VLA Ka low-dec for 2hr

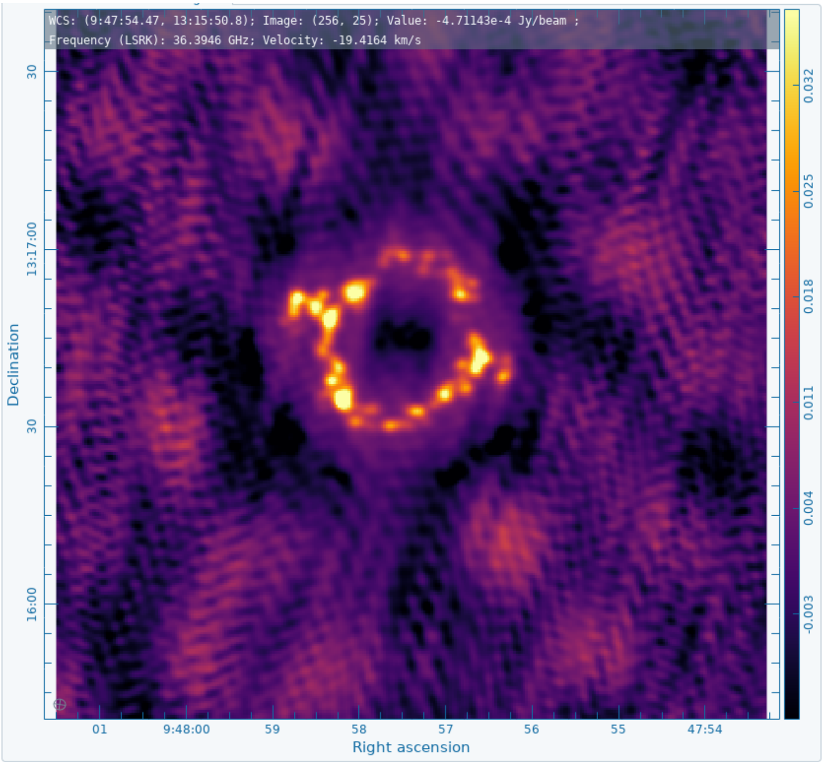


Kepley et al. (2020) Fig. 10

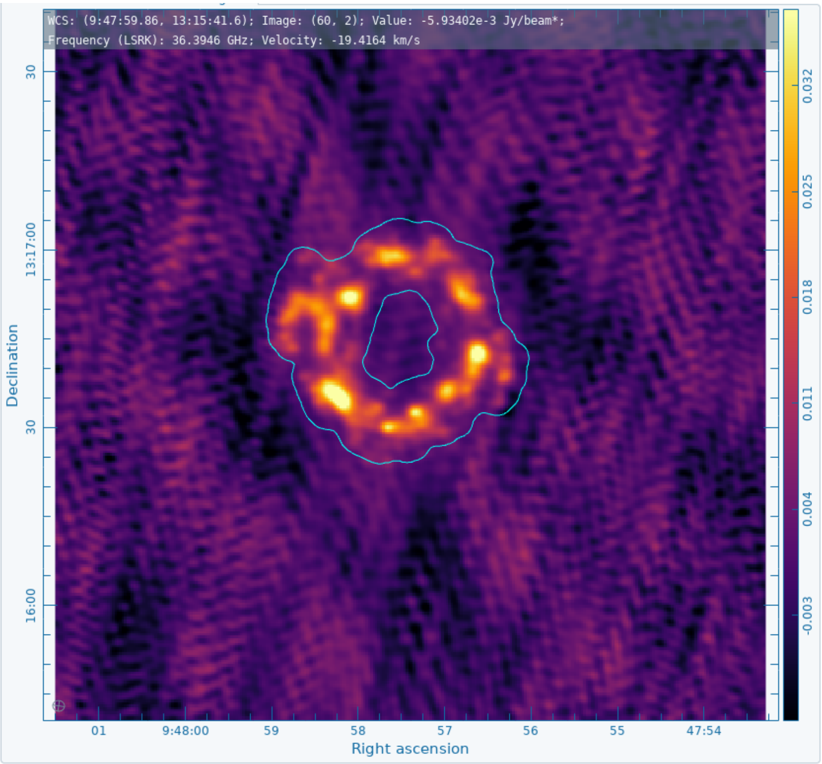
Array	*sidelobethreshold*	*noisethreshold*	*minbeamfrac*	*lownoisethreshold*	*negativethreshold*
12m (short) b75<300m	2.0	4.25	0.3	1.5	0.0 (continuum)/15.0 (line)
12m (long) b75>300m	3.0	5.0	0.3	1.5	0.0 (continuum)/7.0 (line)
7m (continnum/line)	1.25	5.0	0.1	2.0	0.0
12m + 7m combined **TENTATIVE**	2.0	4.25	0.3	1.5	0.0
Above VLA dataset	0.5	4.0	0.1	1.0	1000.0

Automasking: parameter tuning

It is best practice to investigate a few representative channels for faster turn-around when experimenting with parameters.



Circular mask, Hogbom



Auto-multithresh, multiscale

Tools for analyzing spectral lines

While analysis is highly goal dependent, inspection of **moment maps**, **PV diagrams**, **line frequency querying**, **spectral smoothing**, and **line model fitting** are common practices.

- CASA
  - Moment maps: `immoments`, `ia.moments`, `specflux`
  - PV diagrams: `impv` (casaviewer)
  - Gaussian fitting: `specfit` (casaviewer)
  - Spectral smoothing: `specsmooth` (casaviewer)
  - Line querying: casaviewer
- CARTA (user interface examples)
- Community developed tools

CASA: moment maps and aperture sums

Moment maps can be generated in CASA using the `immoments` task. For more complex masking and smoothing use the `ia.moments` task.

```
rms = 0.003  # Jy
inf = 1e8
immoments(
    imagename='my.image',
    moments=[0,1,2,8],
    includepix=[4*rms,inf],
    outfile='my_moments',
)
```

$$M_0 = \Delta v \sum_i I_i$$

$$M_1 = \frac{1}{M_0} \sum_i I_i v_i$$

$$M_2 = \sqrt{\frac{1}{M_0} \sum_i I_i \left(v_i - M_1\right)^2}$$

CASA: moment maps and aperture sums

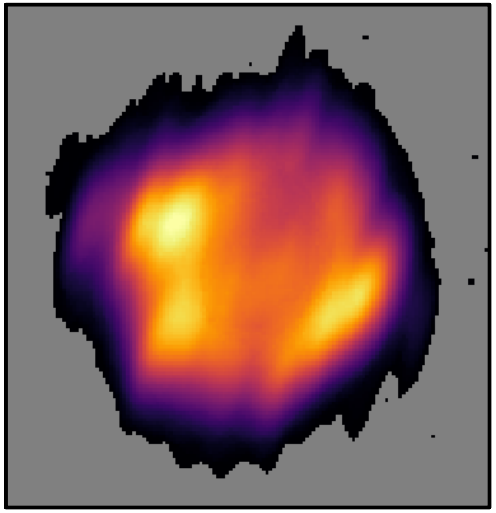
Moment maps can be generated in CASA using the `immoments` task. For more complex masking and smoothing use the `ia.moments` task.

```
rms = 0.003   # J
inf = 1e8
immoments(
    imagename='m
    moments=[0,1
    includepix=[
    outfile='my_
)
```

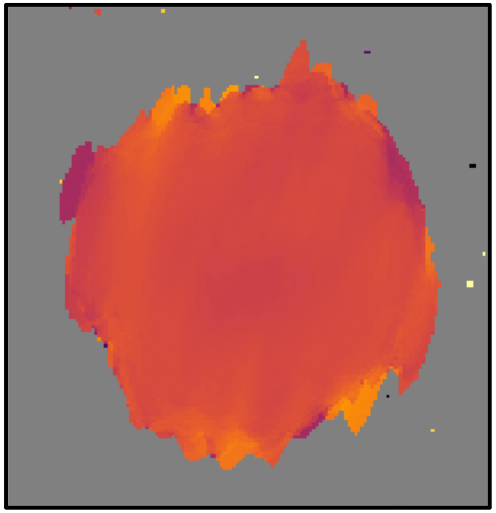
- moments = -1 - mean value of the spectrum
- moments = 0 - integrated value of the spectrum
- moments = 1 - intensity weighted coordinate; traditionally used to get "velocity fields"
- moments = 2 - intensity weighted dispersion of the coordinate; traditionally used to get "velocity dispersion"
- moments = 3 - median value of the spectrum
- moments = 4 - median coordinate
- moments = 5 - standard deviation about the mean of the spectrum
- moments = 6 - root mean square of the spectrum
- moments = 7 - absolute mean deviation of the spectrum
- moments = 8 - maximum value of the spectrum
- moments = 9 - coordinate of the maximum value of the spectrum
- moments = 10 - minimum value of the spectrum
- moments = 11 - coordinate of the minimum value of the spectrum

CASA: moment maps and aperture sums

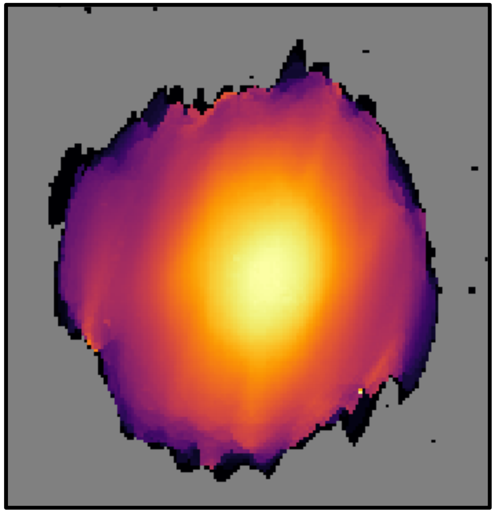
Moment maps can be generated in CASA using the `immoments` task. For more complex masking and smoothing use the `ia.moments` task.



Moment 0

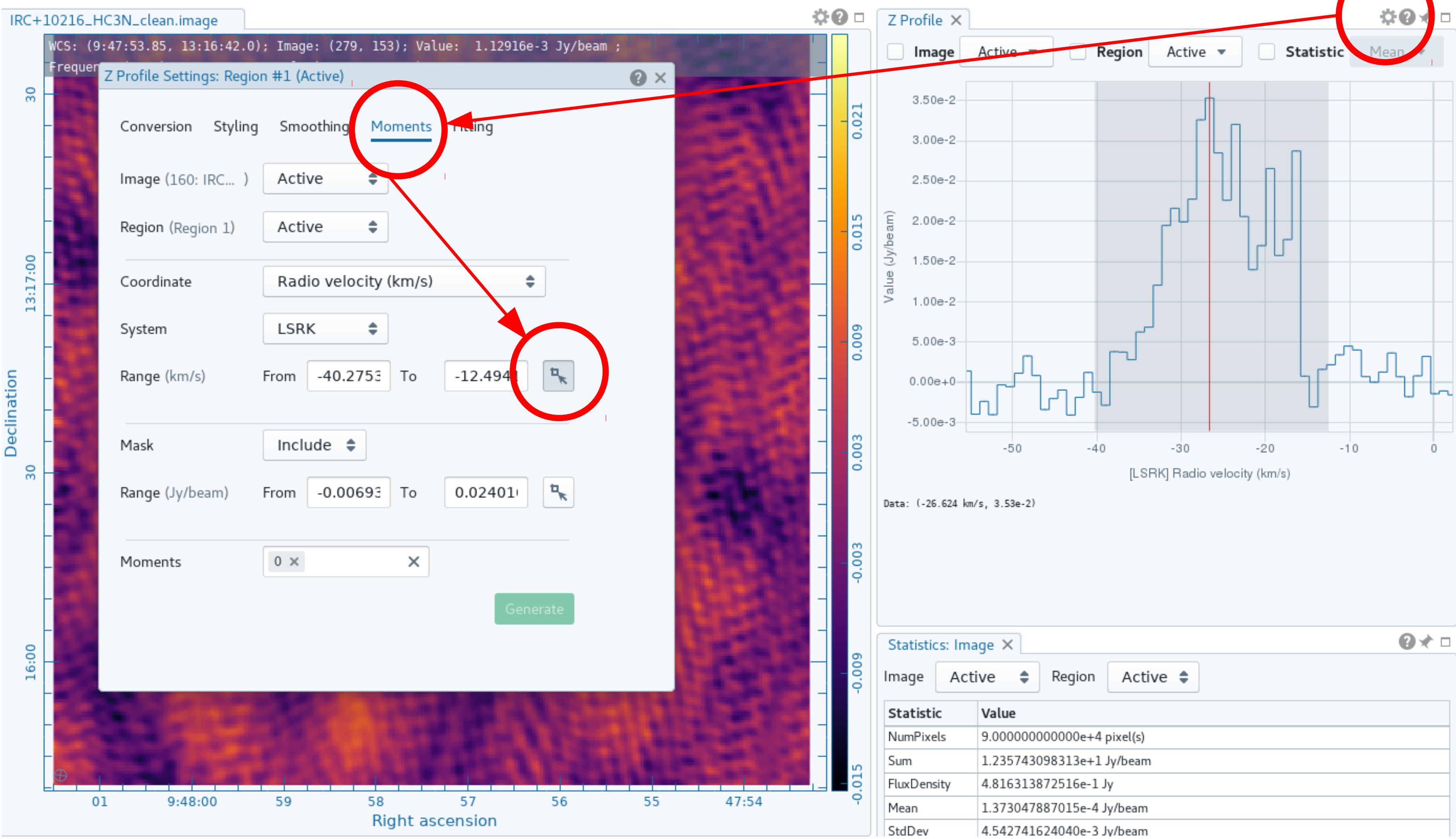


Moment 1



Moment 2

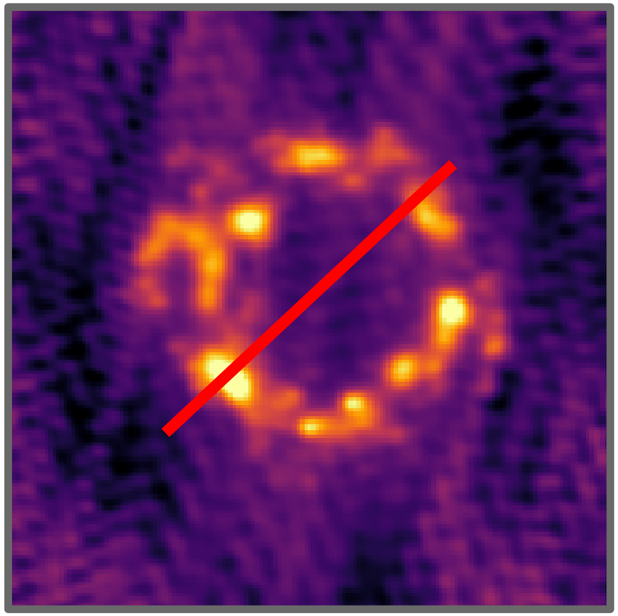
CARTA: moment maps

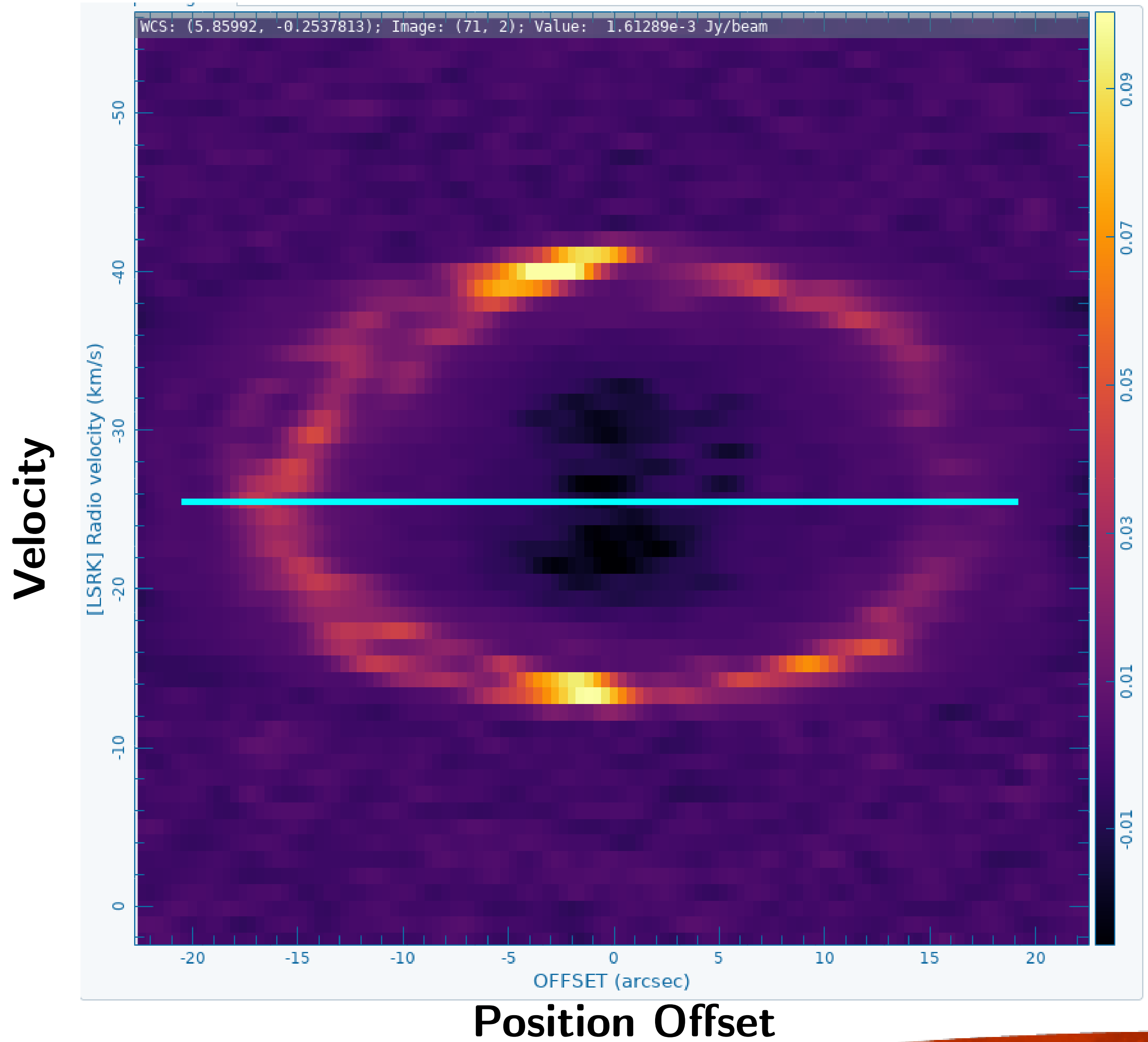


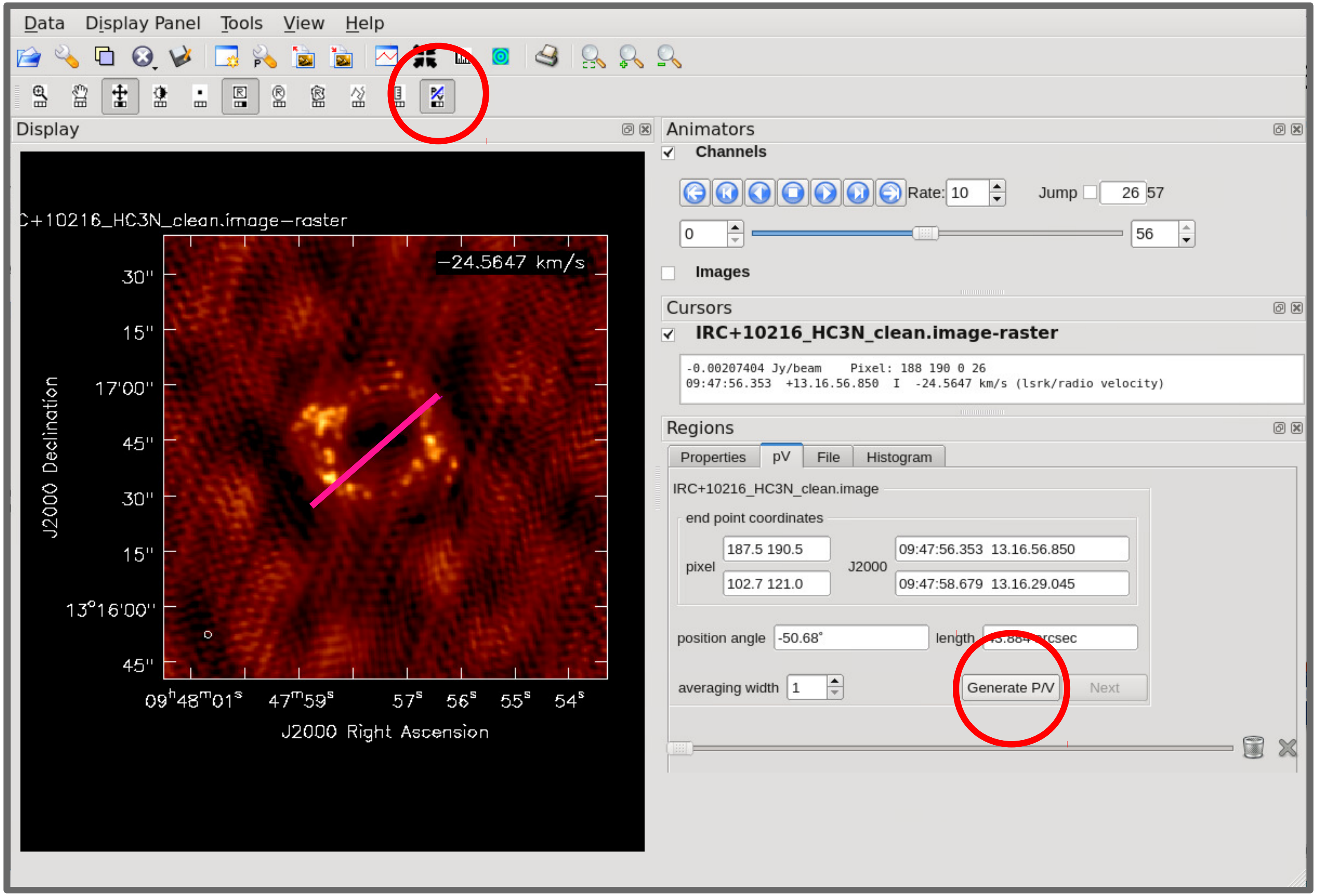
CASA: position-velocity image generation

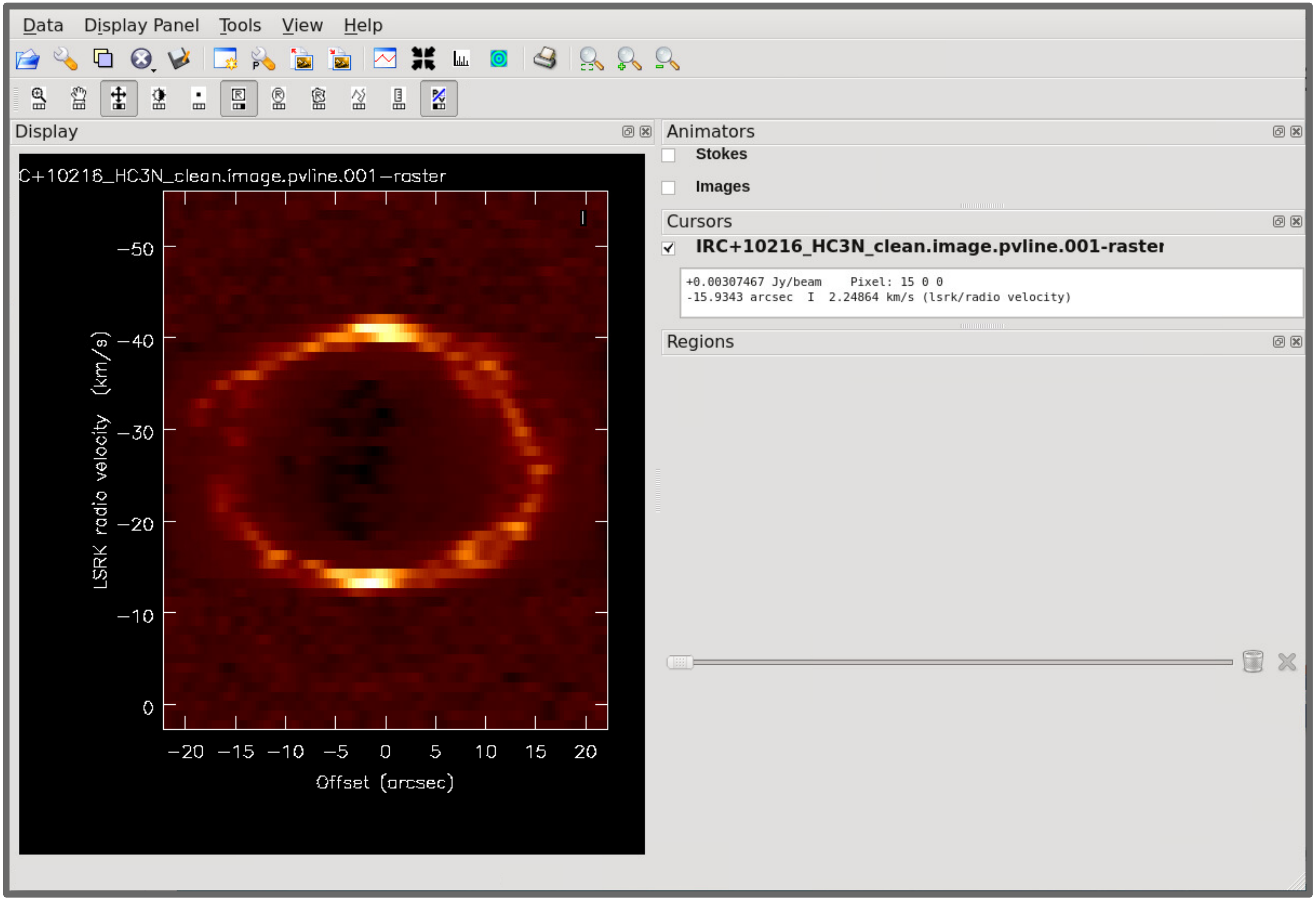
Outflows, disks, and expanding shells all have characteristic patterns when viewed along positive-velocity (PV) cuts. PV-diagrams can be generated in CASA with `impv` and in the casaviewer.

```
impv(
    imagename='my.image',
    outfile='pv.image',
    width=1,  # pix
    start=[100, 120],
    end=[180, 200],
    # ^ can also be coords
)
```

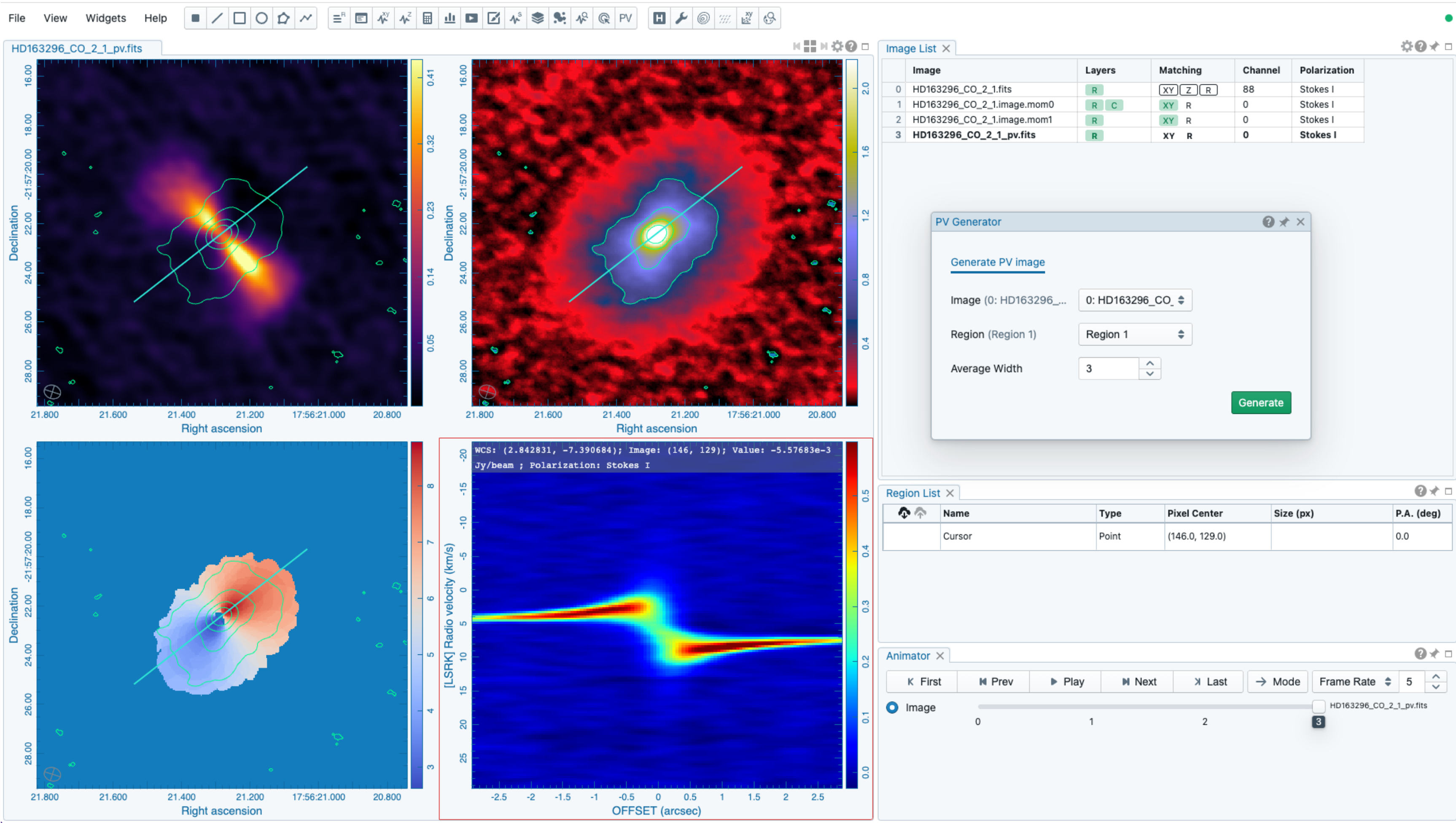



WCS: (5.85992, -0.2537813); Image: (71, 2); Value: 1.61289e-3 Jy/beam
Velocity
[LSRK] Radio velocity (km/s)
-50
-40
-30
-20
-10
0
-20
-15
-10
-5
0
5
10
15
20
OFFSET (arcsec)
Position Offset
0.09
0.07
0.05
0.03
0.01
-0.01


Data Display Panel Tools View Help
Display
C+10216_HC3N_clean.image–raster
–24.5647 km/s
J2000 Declination
J2000 Right Ascension
Animators
Channels
Rate: 10
Jump 26 57
0
56
Images
Cursors
IRC+10216_HC3N_clean.image-raster
-0.00207404 Jy/beam Pixel: 188 190 0 26
09:47:56.353 +13.16.56.850 I -24.5647 km/s (lsrk/radio velocity)
Regions
Properties
pV
File
Histogram
IRC+10216_HC3N_clean.image
end point coordinates
pixel
187.5 190.5
102.7 121.0
J2000
09:47:56.353 13.16.56.850
09:47:58.679 13.16.29.045
position angle
-50.68°
length
averaging width
1
Generate P/V
Next


Data
Display Panel
Tools
View
Help
Display
Animators
Stokes
Images
Cursors
IRC+10216_HC3N_clean.image.pvline.001-raster
+0.00307467 Jy/beam Pixel: 15 0 0
-15.9343 arcsec I 2.24864 km/s (lsrk/radio velocity)
Regions
C+10216_HC3N_clean.image.pvline.001-raster
LSRK radio velocity (km/s)
Offset (arcsec)

CARTA: position-velocity image generator



CASA: spectral smoothing

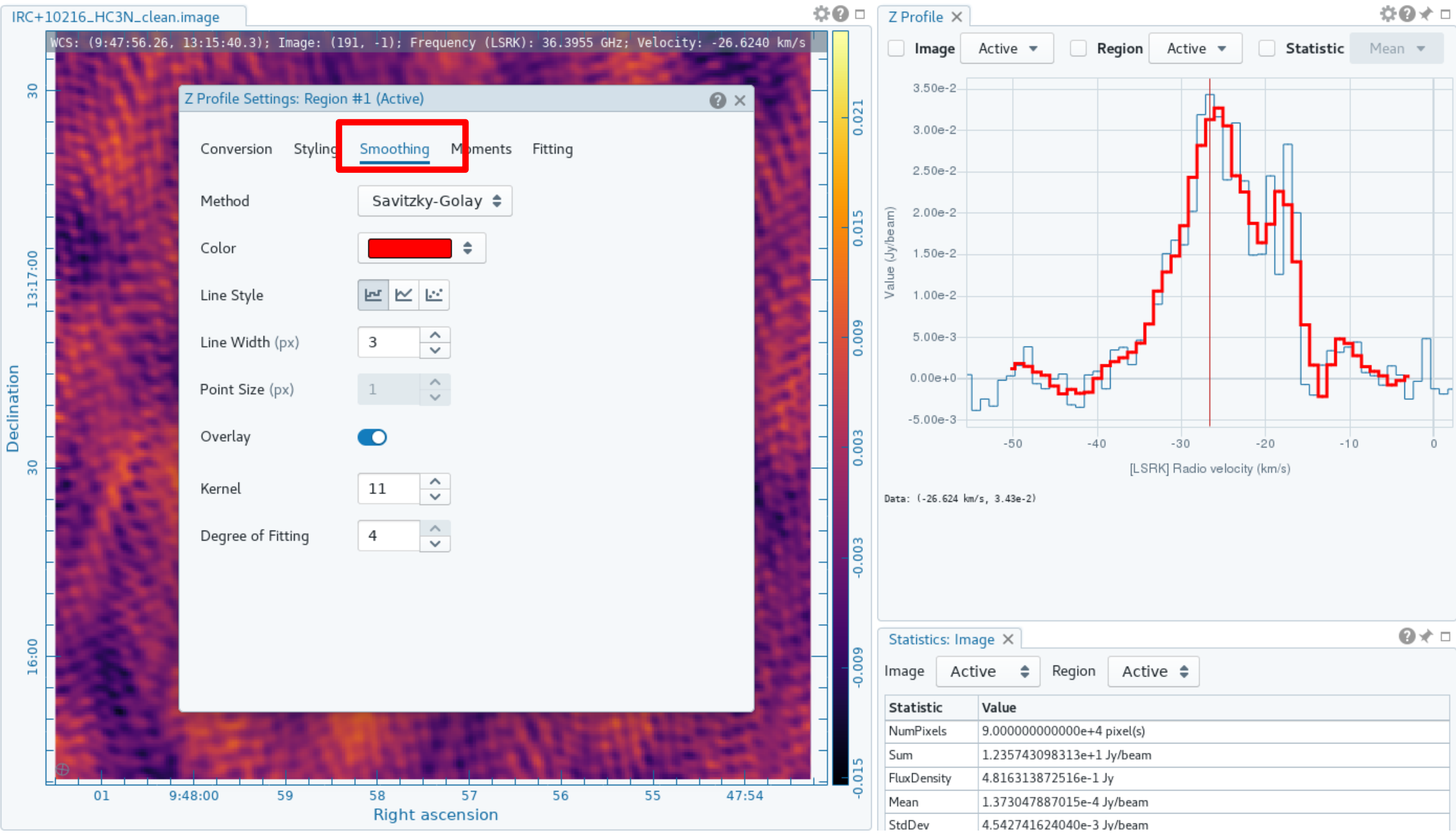
The task `specsmooth` can be used to convolve a template along the spectral axis in order to better match the channel width to the signal-shape and reduce the data size.

```
specsmooth(
    imagename='my.image',
    outfile='smooth.image',
    function='boxcar',
    width=3,  # chan
)
```

Note that `specsmooth` requires a cube with a single restoring beam. The task `imsmooth` can be used to convolve an image to a common-beam scale.

See community-developed Python library: `spectral_cube`

CARTA: spectral smoothing (Savitzky-Golay)



CASA: Gaussian line fitting

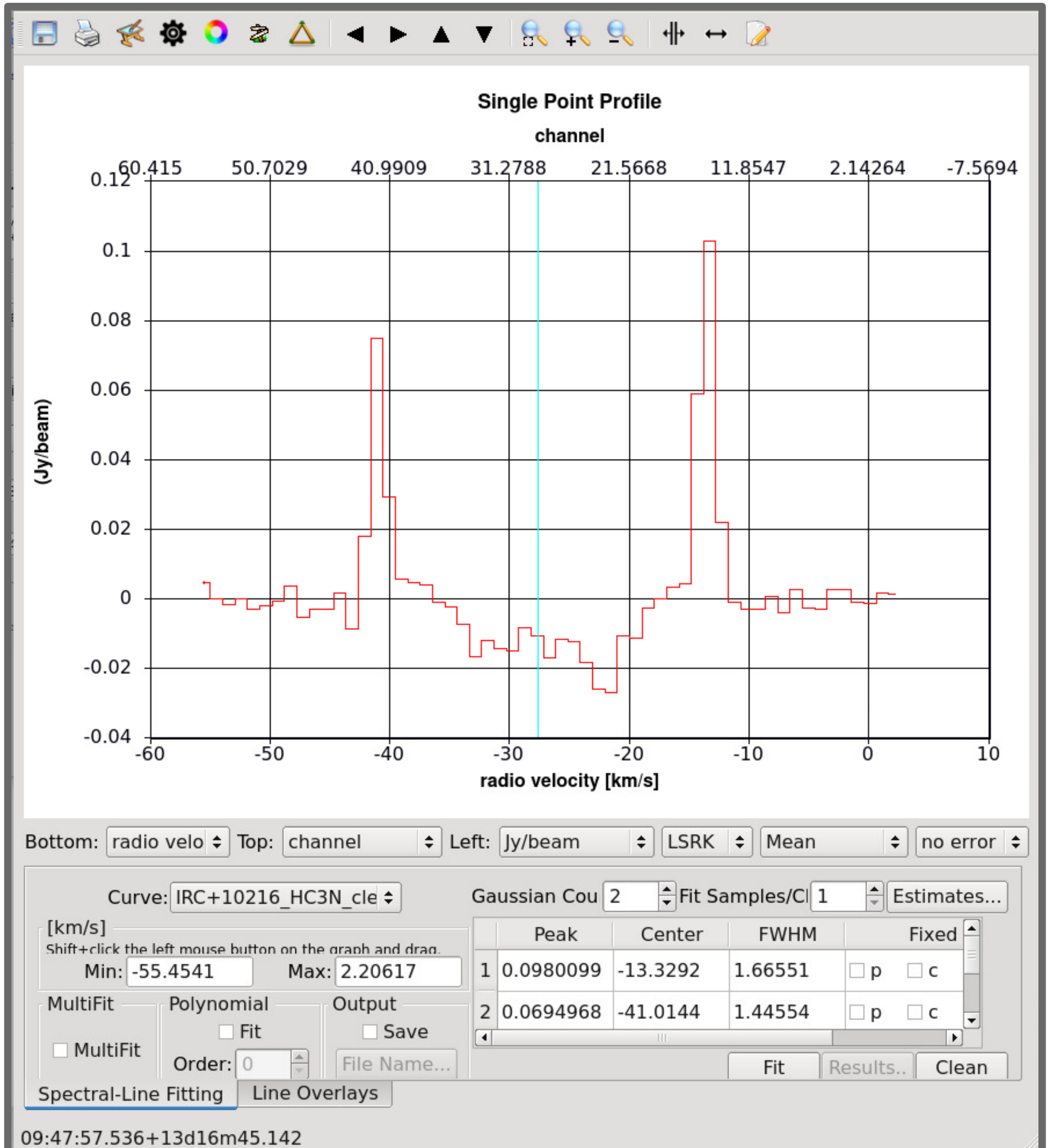
A variety of mechanisms produce Gaussian line profiles in astrophysics. Gaussian functions can be fit to spectra using the `specfit` task and in the casaviewer. A variety of community tools also exist to fit spectra.

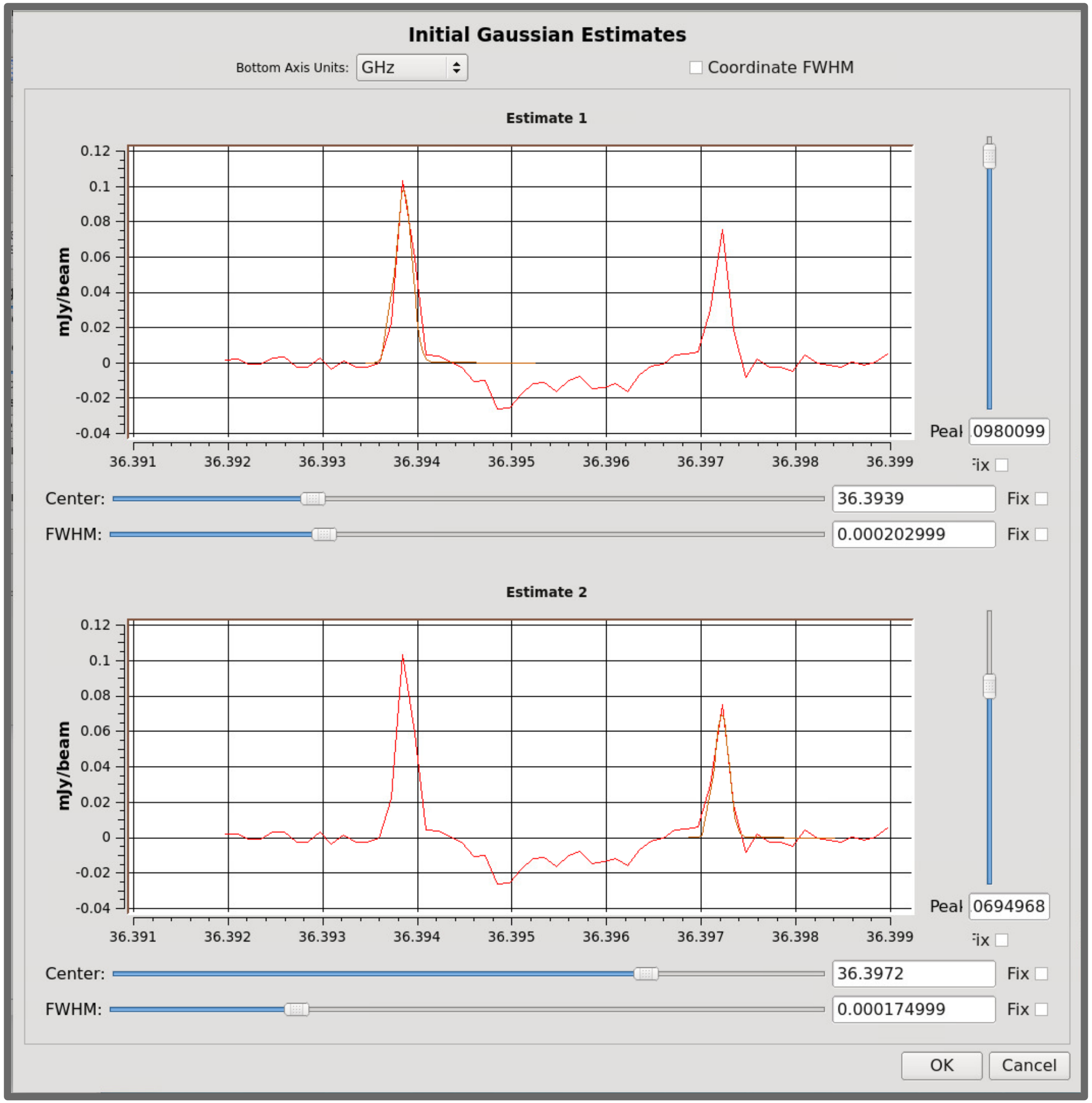
```
fit_results = specfit(
    imagename='my.image',
    ngauss=2,
    box='143,156,147,160',
    multifit=True,
    wantreturn=True,
)
```

Here we use the specfit task to fit each pixel in a 4x4 pixel rectangular region from the bottom-left corner (145,156) to the top-right corner (147,160).

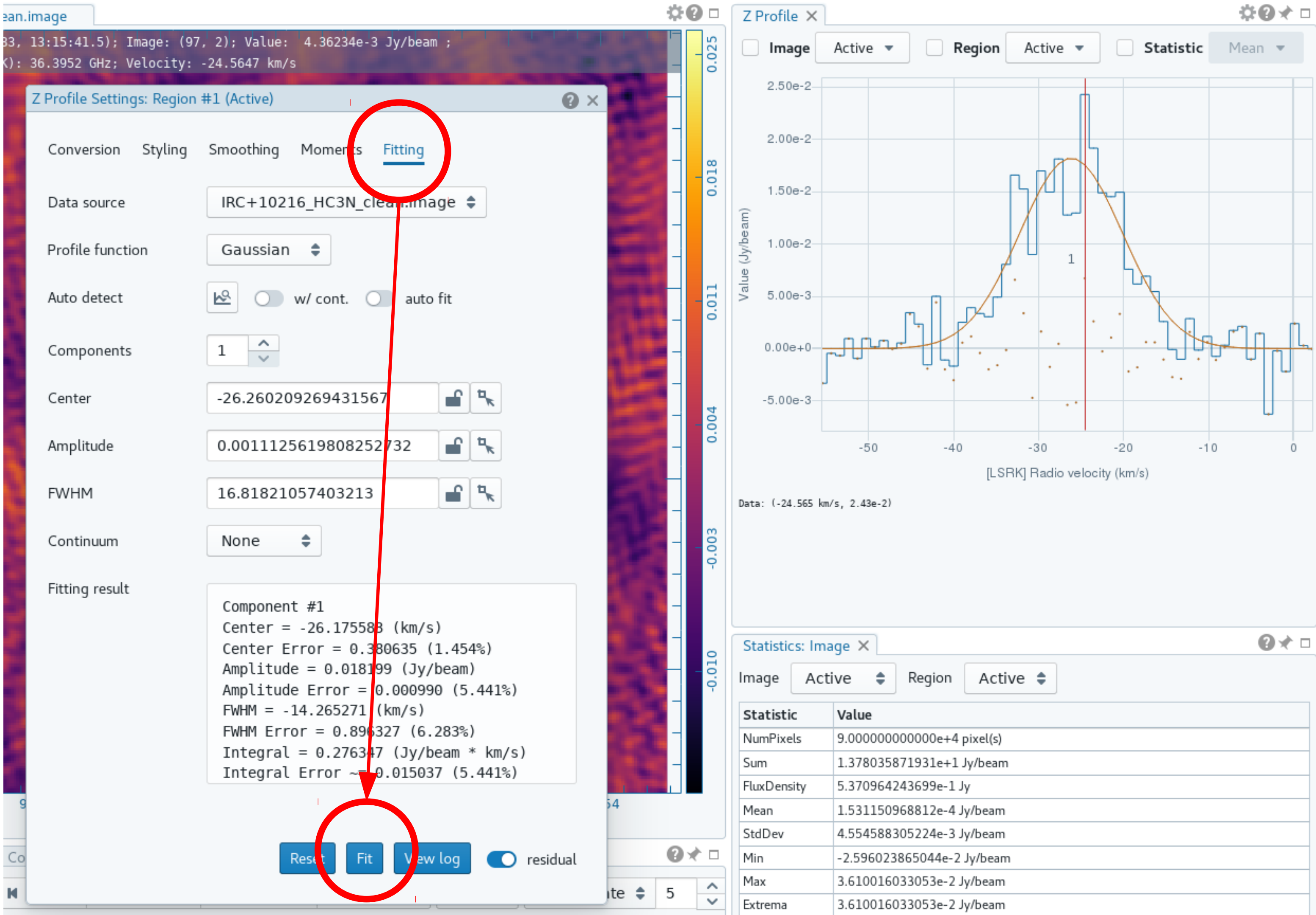
The default behavior is to average the region into a single-spectra, the multifit parameter fits each pixel. The fit results are returned in a Python dictionary.

See community-developed Python libraries: `pyspeckit`, `scouse`, `gausspy+`

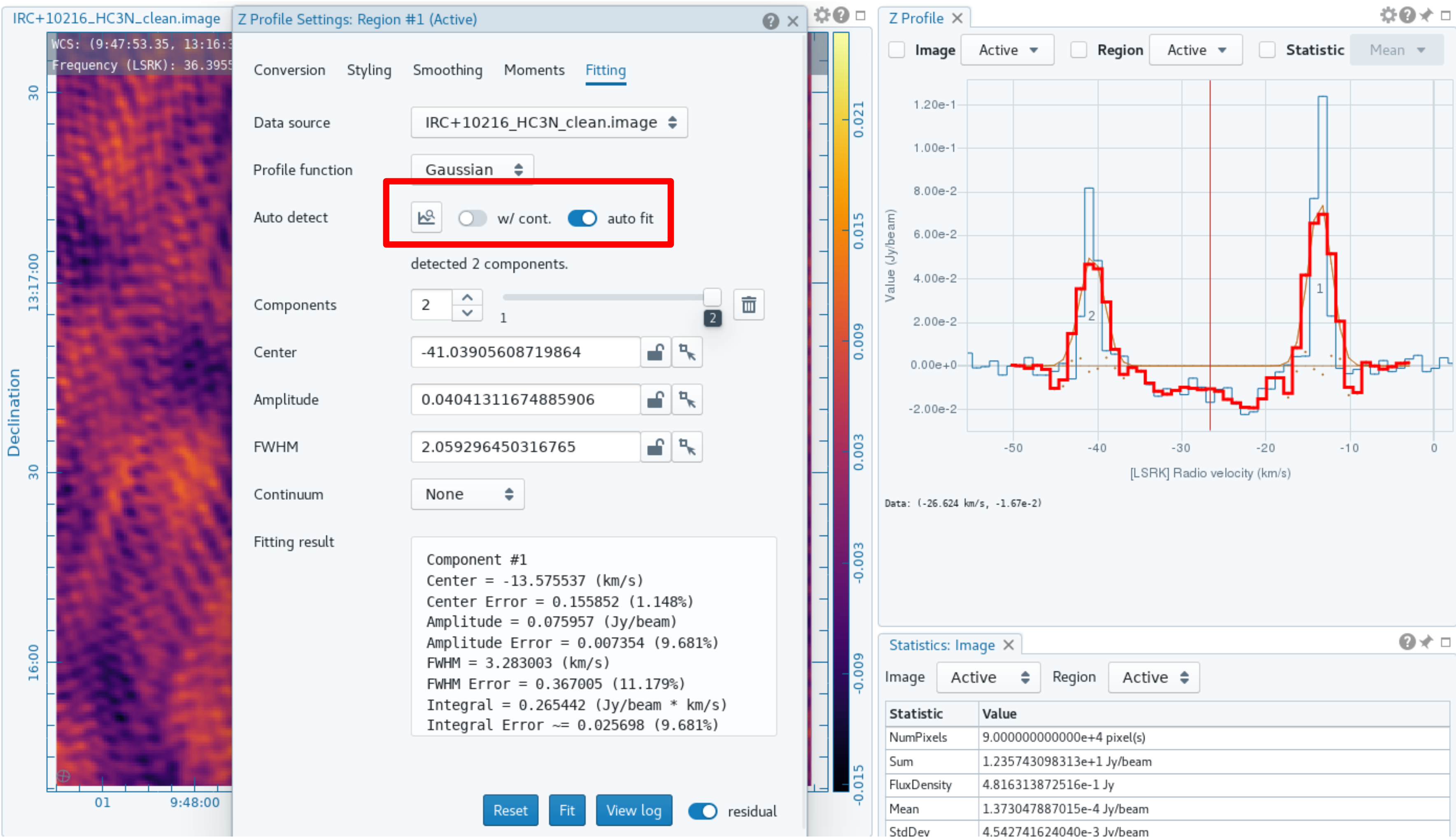

Single Point Profile
channel
60.415
50.7029
40.9909
31.2788
21.5668
11.8547
2.14264
-7.5694
0.12
0.1
0.08
0.06
0.04
0.02
0
-0.02
-0.04
(Jy/beam)
-60
-50
-40
-30
-20
-10
0
10
radio velocity [km/s]
Bottom: radio velo
Top: channel
Left: Jy/beam
LSRK
Mean
no error
Curve: IRC+10216_HC3N_cle
Gaussian Cou 2
Fit Samples/Cl 1
Estimates...
[km/s]
Shift+click the left mouse button on the graph and drag.
Min: -55.4541
Max: 2.20617
Peak
Center
FWHM
Fixed
1 0.0980099 -13.3292 1.66551 p c
2 0.0694968 -41.0144 1.44554 p c
MultiFit
MultiFit
Polynomial
Fit
Order: 0
Output
Save
File Name...
Fit
Results..
Clean
Spectral-Line Fitting
Line Overlays
09:47:57.536+13d16m45.142


Initial Gaussian Estimates
Bottom Axis Units: GHz
Coordinate FWHM
Estimate 1
mJy/beam
Peak 0980099
Fix
Center: 36.3939 Fix
FWHM: 0.000202999 Fix
Estimate 2
mJy/beam
Peak 0694968
Fix
Center: 36.3972 Fix
FWHM: 0.000174999 Fix
OK
Cancel

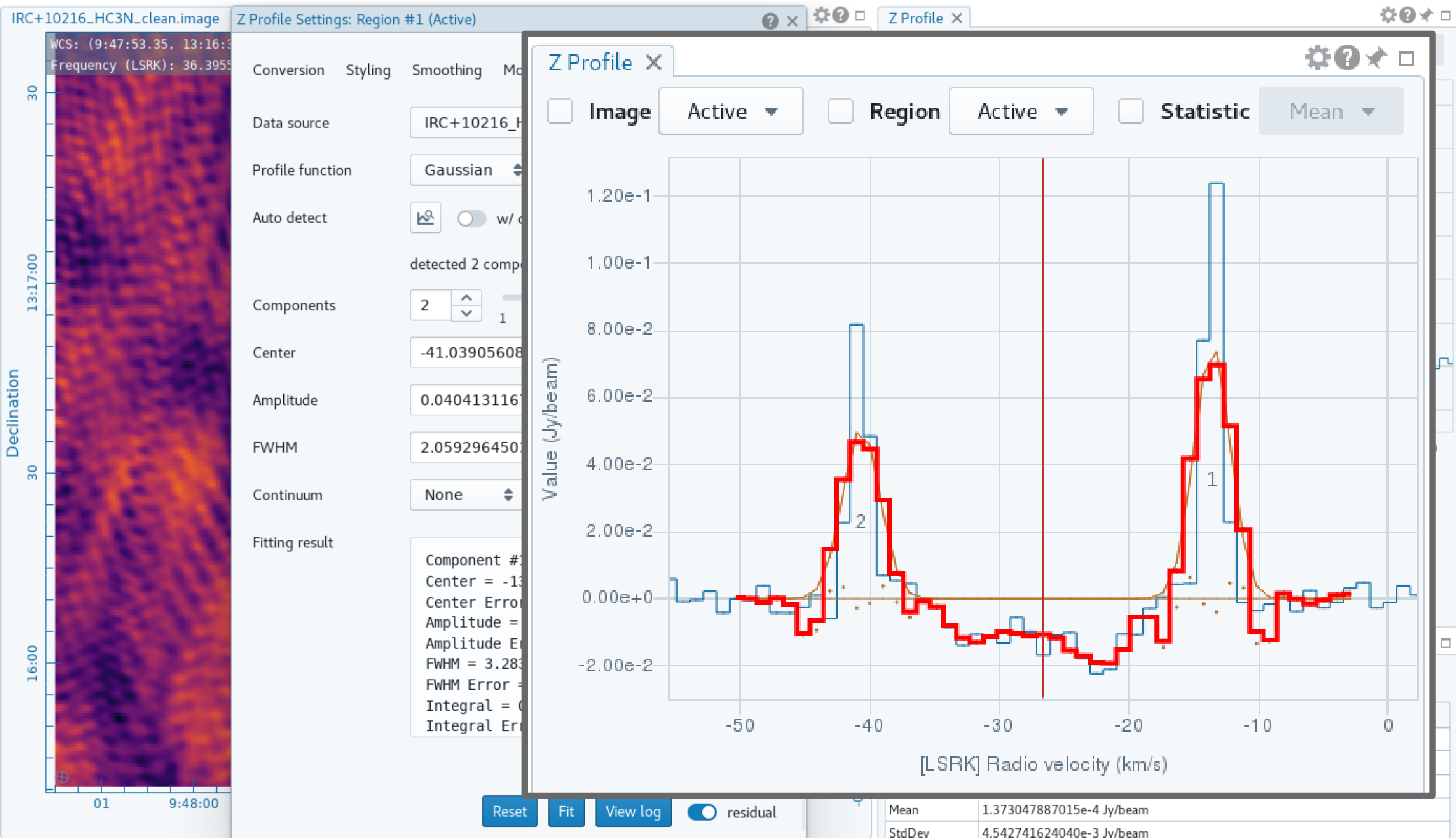
CARTA: Gaussian model fitting



CARTA: automated spectral line fitting

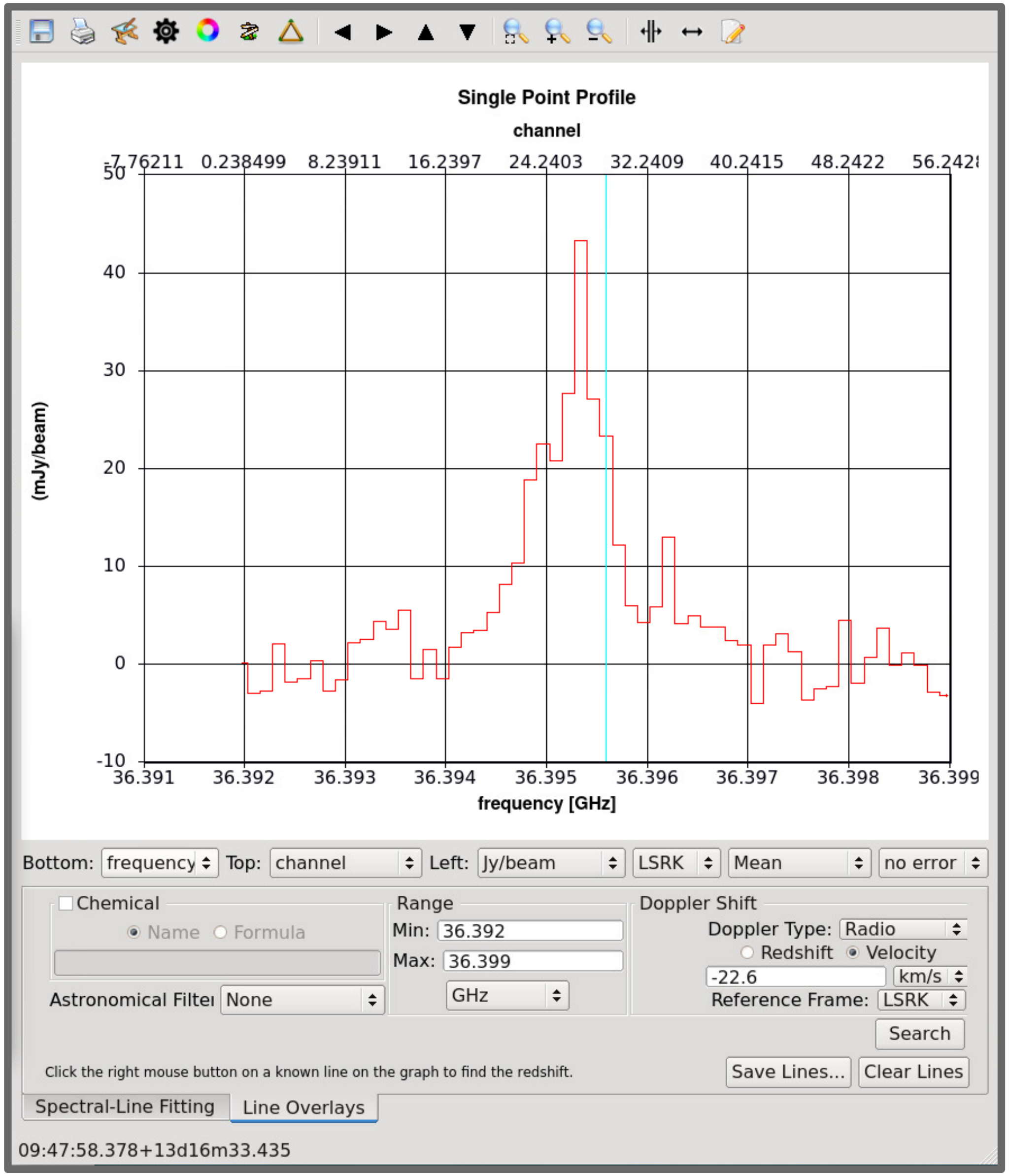


CARTA: automated spectral line fitting



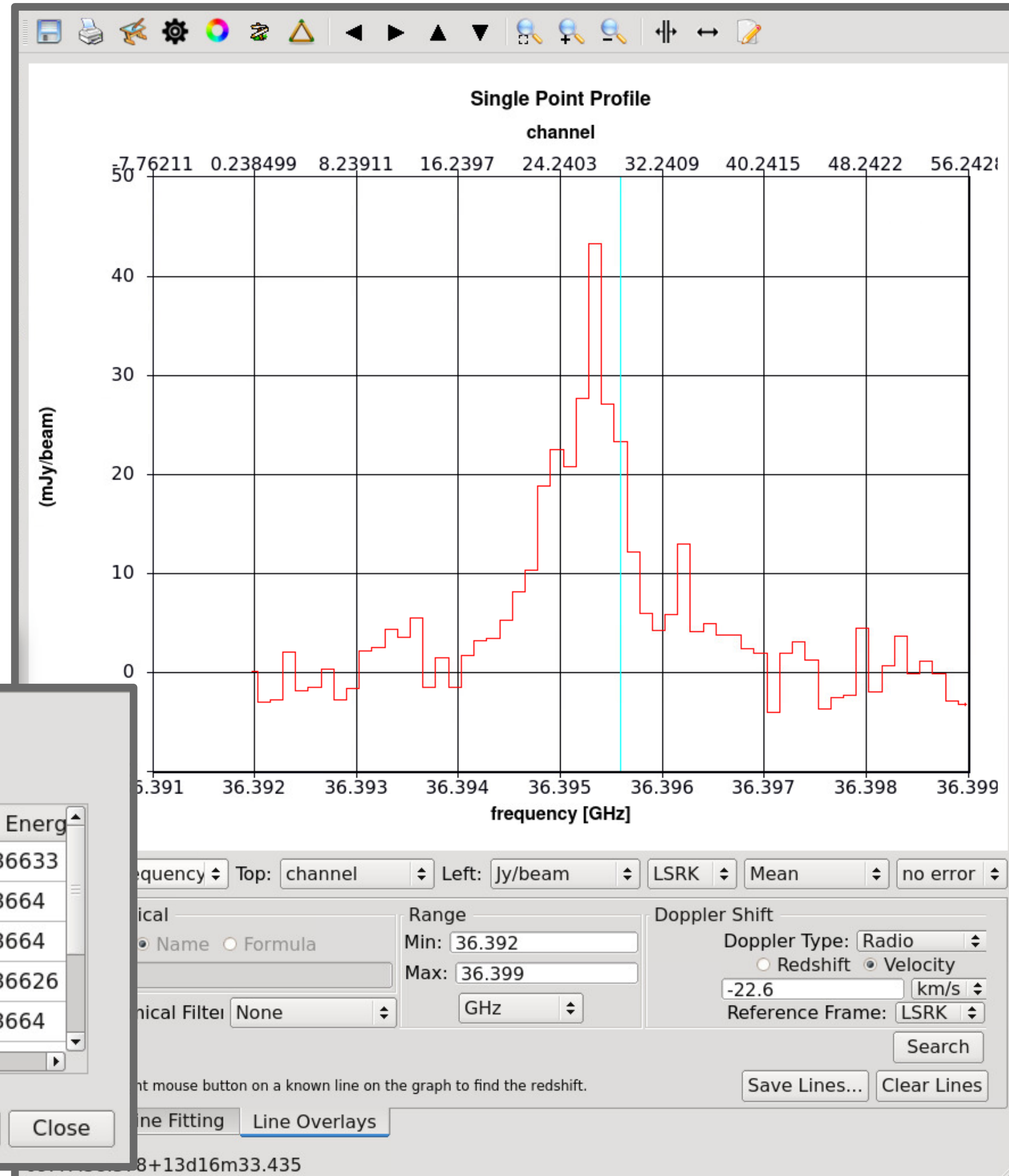
CASA: line querying

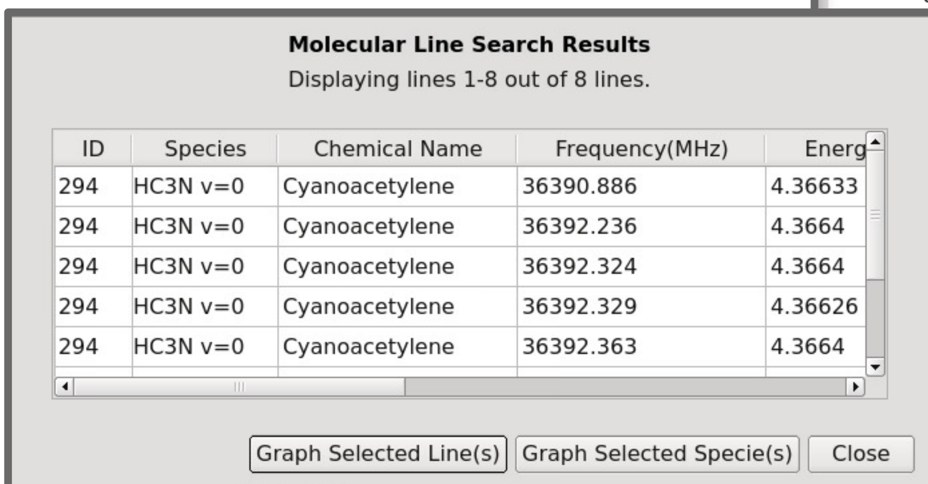
The casaviewer includes an integrated spectral line list query service in the spectra viewer. For a further guide on the casaviewer, see the “Image Cube Visualization” notebook on casadocs.readthedocs.io



CASA: line querying

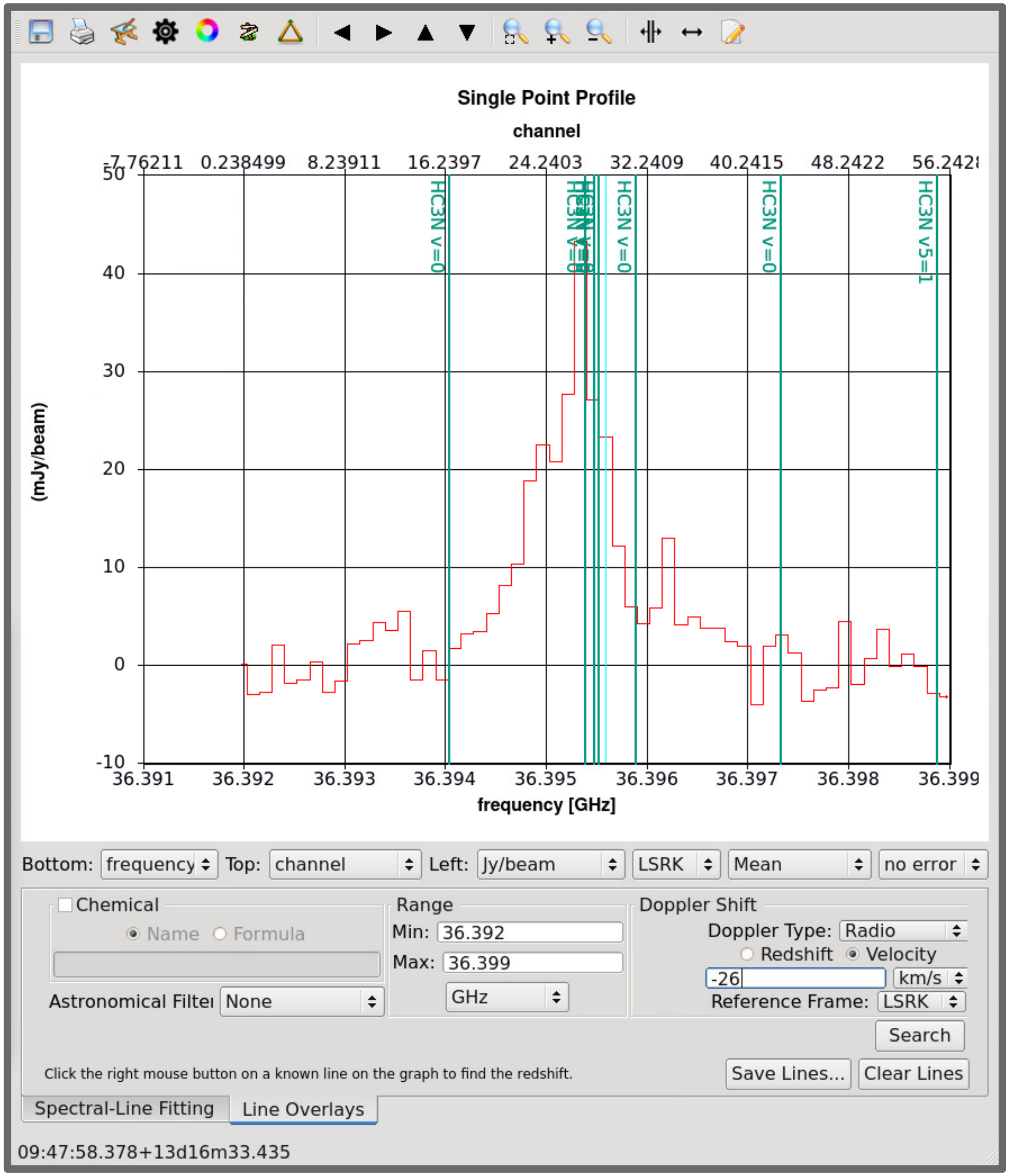
The casaviewer includes an integrated spectral line list query service in the spectra viewer. For a further guide on the casaviewer, see the "Image Cube Visualization" notebook on casadocs.readthedocs.io



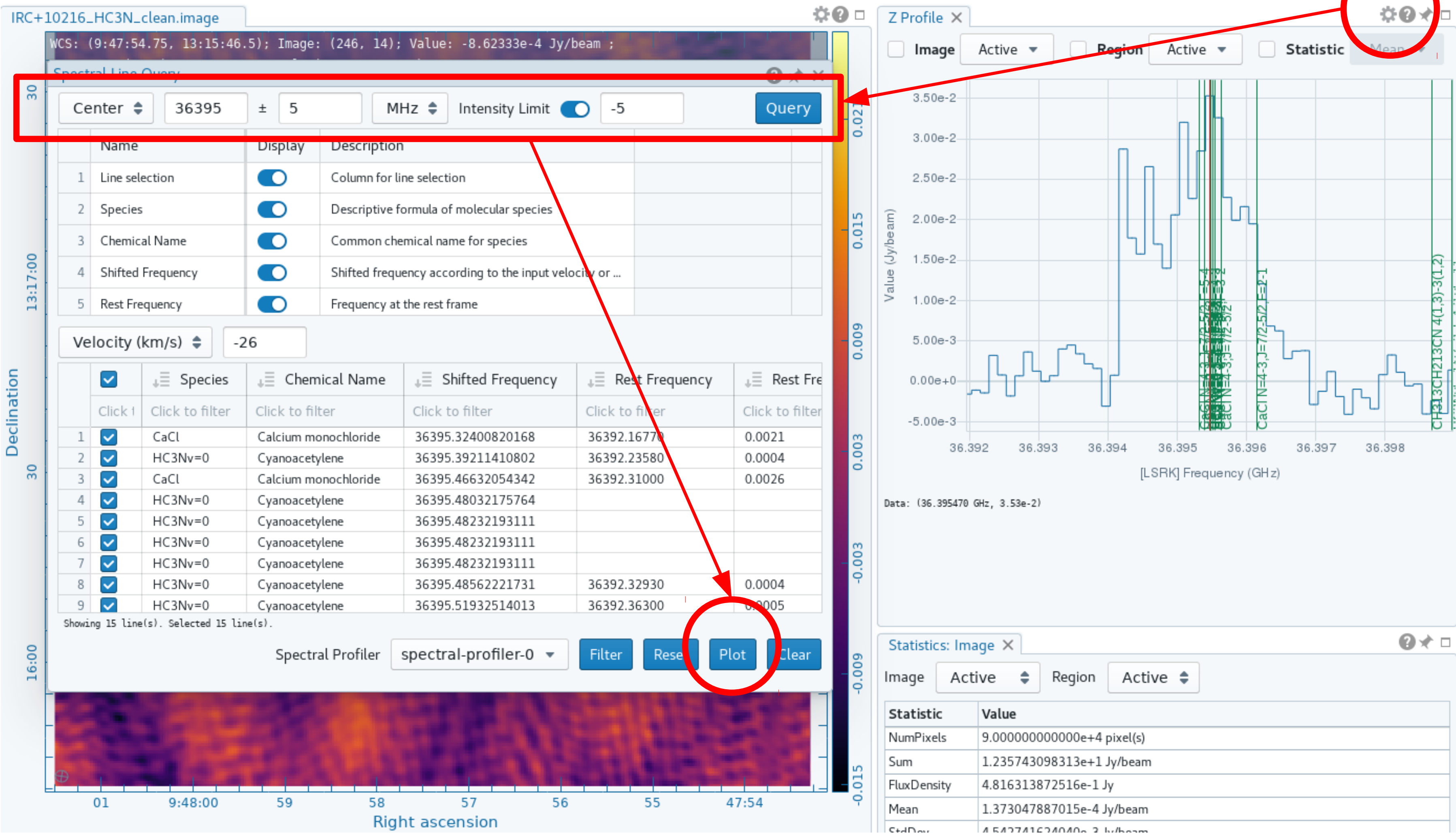


CASA: line querying

The casaviewer includes an integrated spectral line list query service in the spectra viewer. For a further guide on the casaviewer, see the "Image Cube Visualization" notebook on casadocs.readthedocs.io



CARTA: line querying



Other spectral line considerations

- Single-dish and interferometric image combination
  - feather, startmodel, and sdint methods
- Imager parallelization for cube imaging
- Polarization (for, e.g., Zeeman splitting)
- Scripting CASA to image multiple targets and spectral windows
- Using the PySynthesisImager API for novel masking approaches

www.nrao.edu
science.nrao.edu
public.nrao.edu

The National Radio Astronomy Observatory is a facility of the National Science Foundation operated under cooperative agreement by Associated Universities, Inc.