A CASSIS Cookbook for the ALMA Era

Contributing authors: Pooneh Nazari, Martijn van Gelder, Magnus Persson, Mihkel Kama, Daniel Harsono, Aida Ahmadi, and Michiel Hogerheijde

September 14, 2022

Contents

1 Introduction

The sensitivity of ALMA provides rich spectral line data on many astronomical objects. CASSIS is a powerful tool to find the column densities and excitation temperatures of different molecules from a spectrum. CASSIS is a free, interactive spectrum analyser developed by IRAP-UPS/CNRS. Detailed information on how to use CASSIS are available on the CASSIS website.

This document offers a *simple cookbook* on how to use CASSIS, with a focus on ALMA data. It describes how to use CASSIS with a special emphasis on spectra extracted from ALMA data (§2), and how to extract a spectrum from ALMA data in a format that can be read by CASSIS (§3). Both sections contain references to hands-on practice material that can be obtained from the Allegro website[1]. Here you will find the tarball `CASSIS_cookbook_material.tar.gz` that can be unpacked with the command '`tar -xvzf CASSIS_cookbook_material.tar.gz`'. It contains a copy of this cookbook, scripts, an ALMA data set that you can use to extract a spectrum from, a spectrum formatted for CASSIS, and a folder with codes and files relevant for scripting CASSIS. Table 1 lists the files to be used with this cookbook and their descriptions.

Enjoy CASSIS in the ALMA era!

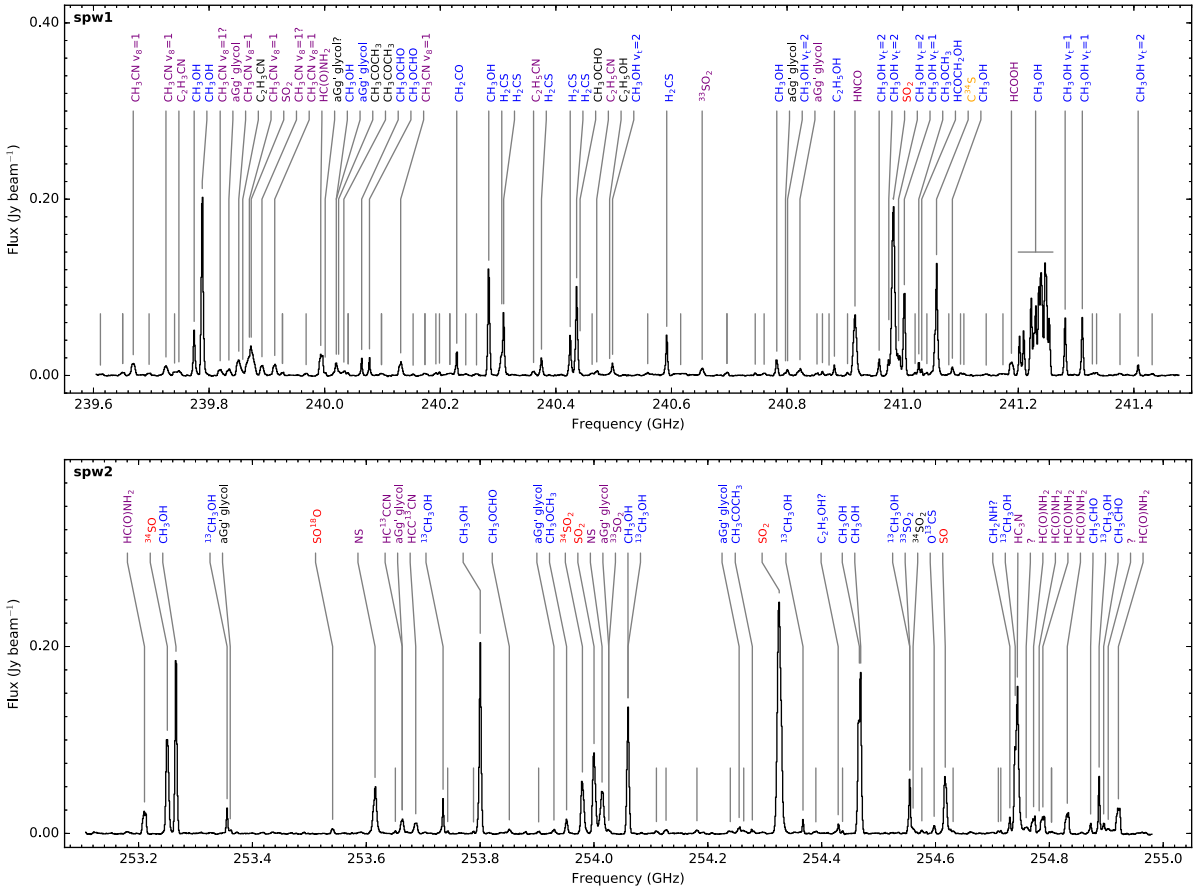

Figure 1: Example ALMA spectrum showing the rich chemistry in the forming O-star AFGL 4176 obtained from Johnston et al. 2020a, ApJ 896, 35.

[1] https://www.alma-allegro.nl/software/cassis

File name	File type	Description
`CASSIS_cookbook_ALMA_era.pdf`	PDF	This cookbook
`afgl4176.contsub.image`	CASA	Example data cube containing ALMA observations of the hot core AFGL 4176 from project 2012.1.00469.S (PI: Johnston)[2] that has already been continuum subtracted and is ready for analysis
`afgl4176.contsub.fits`	`.fits`	Example data cube in `FITS` format
`afgl4176_spec_casa_extracted.txt`	space-separated `ASCII`	Example spectrum extracted in CASA (see Sect. 3.2)
`afgl4176_spec_carta_extracted.txt`	tab-separated `ASCII`	Example spectrum extracted in CARTA (see Sect. 3.3)
`make_cassis_files.py`	Python script	This script will perform three tasks: 1) Shifts the frequencies of the provided spectrum from sky to rest frame using the supplied source velocity (see Sect. 2.6) 2) Converts the spectrum from step 1) to CASSIS (`.fus`) format (see Sect. 2.6) 3) Creates a telescope file. For this, the user must either provide the angular resolution and frequency of the observations, OR provide the `FITS` cube from which this script will automatically obtain the needed information (see Sect. 2.4)
`cassis_spectrum.fus`	`ASCII`	Example spectrum file in CASSIS format (see Sect. 2.6)
`alma_968`	`ASCII`	Example ALMA Telescope file (see Sect. 2.4)

Table 1: Description of files included with this cookbook.

2 CASSIS 101

This section explains how one can use the CASSIS spectral analysis tool to find the column densities and excitation temperatures of different molecules from a spectrum. There are already other resources available on how to use this tool, for example the CASSIS website.

The section is structured as follows. First, we explain how to start CASSIS and what the most useful tools of CASSIS are for ALMA data spectral analysis. Then we describe how to get a few files that are needed for those tools. Finally, we give a few tips that might be helpful for someone who is new to working with spectra. What is explained here works on CASSIS version 5 and most of it should work on version 6. However, it is important to note that there are some issues with importing databases into CASSIS version 6 and hence what is explained here especially on databases might not work for version 6.

2.1 Starting CASSIS

Starting of CASSIS will be different based on what operating system you are using. If you have downloaded CASSIS yourself and are using it on your own computer you can use the information on CASSIS website to learn how to start it up based on your own computer setup. However, if you are using CASSIS on the Leiden Observatory computers you can write 'cassis' on the command line in the terminal to start it up. CASSIS by default will use 1024 MB of memory but this can be specified when starting up CASSIS by writing 'cassis 1024' for example.

2.2 Tools

Once CASSIS starts, a bar shown in Fig. 2 pops up. In this figure the functions that will be explained in detail are numbered and we will refer to these in the rest of this section. The rest of the functions not explained here are either easy to use/similar to the ones explained in detail or they are not of interest when fitting for different parameters using synthetic spectra assuming LTE conditions.

2.2.1 Sepctrum Analysis

Let's start with the tool number 1, Sepctrum Analysis. With this we can get a quick look at a spectrum. This is useful for example to find whether certain species are detected/have transitions in the frequency range of the data. To use this tool one can click on it and then another window opens (Fig. 3). By clicking on 'Load' and then on 'open local resource' (assuming the spectrum is already saved on your computer in .fus format, see Sect. 2.6 on how to make that), one can choose the spectrum and then click on 'Select spectrum'. Once this is done, the telescope file (see Sect. 2.4) can be chosen by clicking on the '???' in Fig. 3. Once the telescope file is chosen you can click on 'Display' and then the spectrum should appear. The formats of the telescope file and the spectrum itself are described in Sect. 2.4 and 2.6. Please note that CASSIS uses spectra in units of Kelvin.

Now that the window in Fig. 2 is expanded with the spectrum being displayed underneath as shown in Fig. 4, there are a few tabs on the right that are useful. The 'InfoPanel' shows what is plotted in the window on the left and the colour of these elements can change by clicking on the coloured bar next to name of the plotted file. This is useful in case many spectra/models are over plotted so you can choose the best colour combination depending on your preference. The 'Overlays' tab can be used to overplot another spectrum/a model/line list by selecting a 'Data file' or 'LineList' already saved on your computer. The 'Species' tab can be used to see what molecules have transitions in the frequency range of the data using CASSIS or your own imported database. You can select the molecule you want to check from the list and put any cut you want on the upper energy level ($E_{\rm up}$) and Einstein $A_{\rm ij}$ coefficient and then click 'Display' on the bottom. Make sure that the 'show' option is checked. Then you will see if there are any transitions of that molecule in the frequency range of the data. If the $V_{\rm lsr}$ is set correctly in your .fus file for the spectrum there is no need to change it here, otherwise, you can change the $V_{\rm lsr}$ at this stage too.

Please note that you need to make sure CASSIS is using the most updated database available in its 'database' directory located in your CASSIS folder. Not checking this can result in CASSIS using an

outdated database or a not complete one that misses many molecules. More on how to check this and databases in general can be found in Sect. 2.3.

Tip 1: You can change the template of the species shown to have a smaller sized group of species rather than 'All Species' possible which can be overwhelming. For example you can choose the ISM template if you are interested in the most popular species detected in the ISM.

Tip 2: To find the molecule you want to show its transitions, the best way is to have your template as 'All Species' and then ***right click*** *on where it is written 'Species' just next to where it is written 'Tag' and write the formula of your molecule in the box that pops up, e.g. write CH3CN (important to write in capitals). If you do not know how to write the molecule in the format that CASSIS wants, e.g. for isotopologues, you can do the same by right clicking on the 'Tag' option and writing the tag of the molecule.*

Tip 3: If you left click on the Species you can order the molecules in alphabetical order or left clicking on Tag can order the species going from low to high tags. Left clicking again will reverse the order. This is useful if you know the first two numbers of the species' tag, i.e its molar mass. Moreover, left clicking on 'Sel.' will check all and left clicking again will uncheck all the boxes.

Tip 4: As mentioned in the previous tip, the first two digits of of a species' tag are its molar mass. For example, the first two digits of CH_3OH tag are 32 because 12 (C) + 3 × 1 (H) + 16 (O) + 1 (H) = 32.

Tip 5: To identify a single line you can zoom-in on the line of interest (by clicking and dragging from left to right) and check all the boxes of the species in your template of interest. This gives all the species possible for the line of interest and you can decide based on the Einstein A_{ij} and E_{up} which one(s) is more probable to fit for as explained in Sect. 2.2.2.

2.2.2 Line Analysis

Line Analysis, number 3 in Fig. 2, is perhaps the most useful tool to find by-eye measurements of the column density and excitation temperature of a molecule. After clicking on its icon on the bar shown in Fig. 2 a window shown in Fig. 5 should pop up. Again here you need to load your data and the telescope file, adjust the cut-off you want for E_{up} and A_{ij}. Enter the V_{lsr} and make sure to check the box next to 'LTE-RADEX'. The 'Mode' should be set to 'Full LTE' (assuming that you want to find the column density and excitation temperature in LTE conditions). Then you can choose the molecule you want to fit for from the list of species on the top right (the tips mentioned in Sect. 2.2.1 also apply here). Once that is chosen a line should pop up at the bottom where you can change the column density, excitation temperature, full width half maximum (FWHM) and the source size. If the size of the emitting region, i.e., the source size, is unknown and the emission is unresolved, often the source size is set equal to the size of the beam, although this is not correct and changes the column densities you derive[3]. No need to worry about 'Abundance (/H2)' option, that will change automatically. You can also change the background temperature (Tbg) and the rms.

After setting all these parameters, you can click on 'Display' and you should see various subplots

[3]The choice of source size is important; see Sect. 2.5 for a more in-depth discussion on what source size to choose.

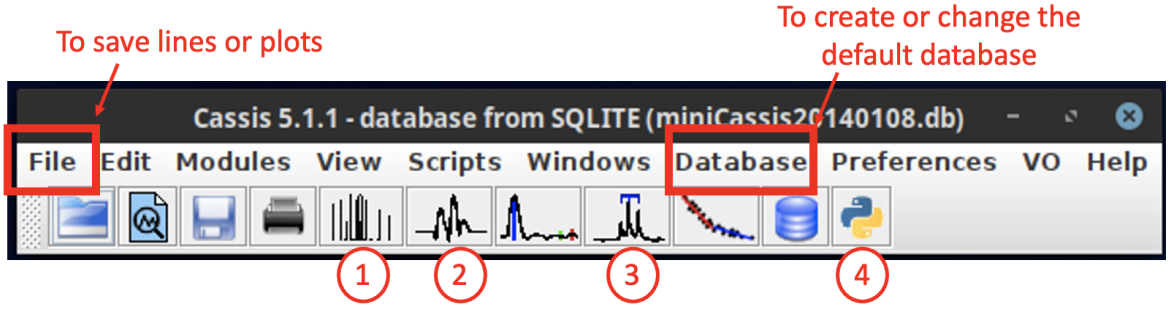

Figure 2: The bar you see when starting CASSIS.

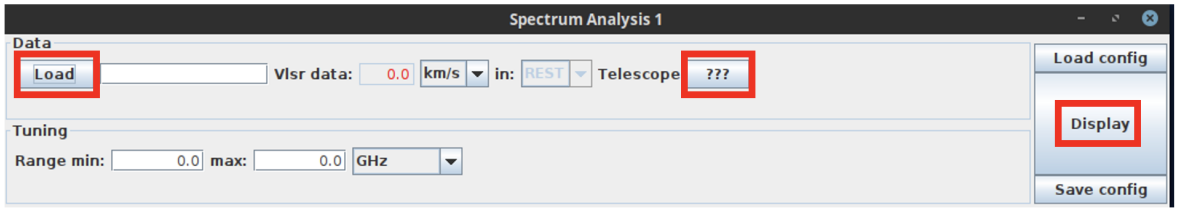

Figure 3: After clicking on Spectrum Analysis.

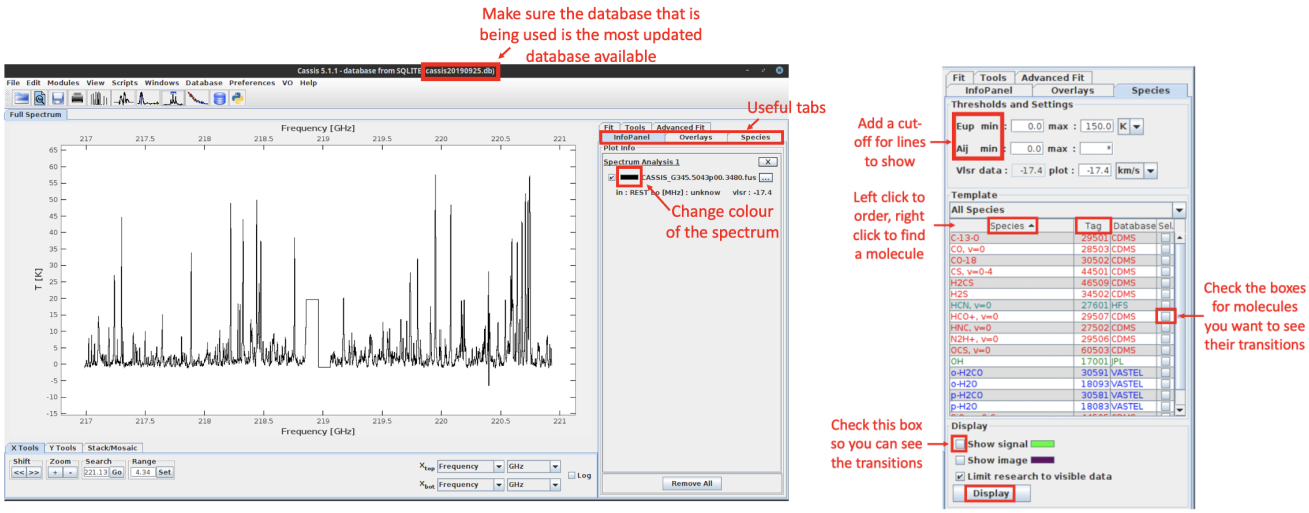

Figure 4: After loading in a spectrum.

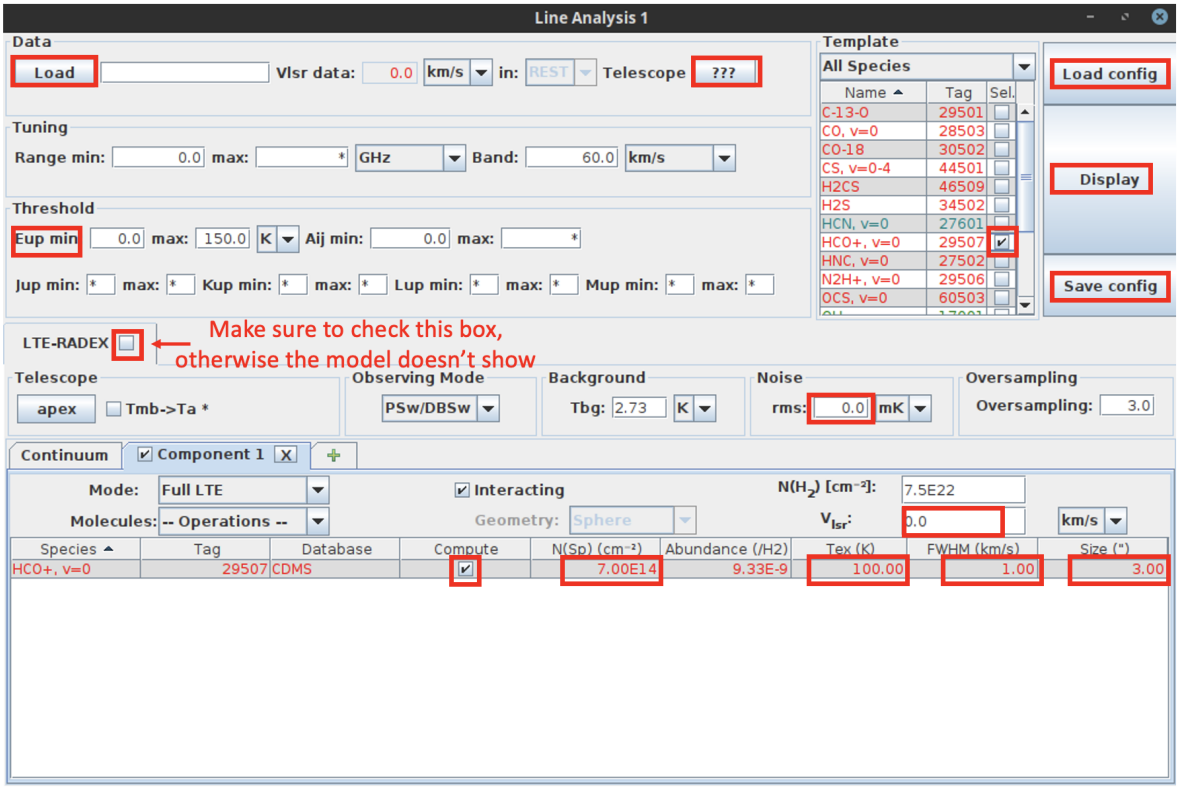

Figure 5: After clicking on Line Analysis.

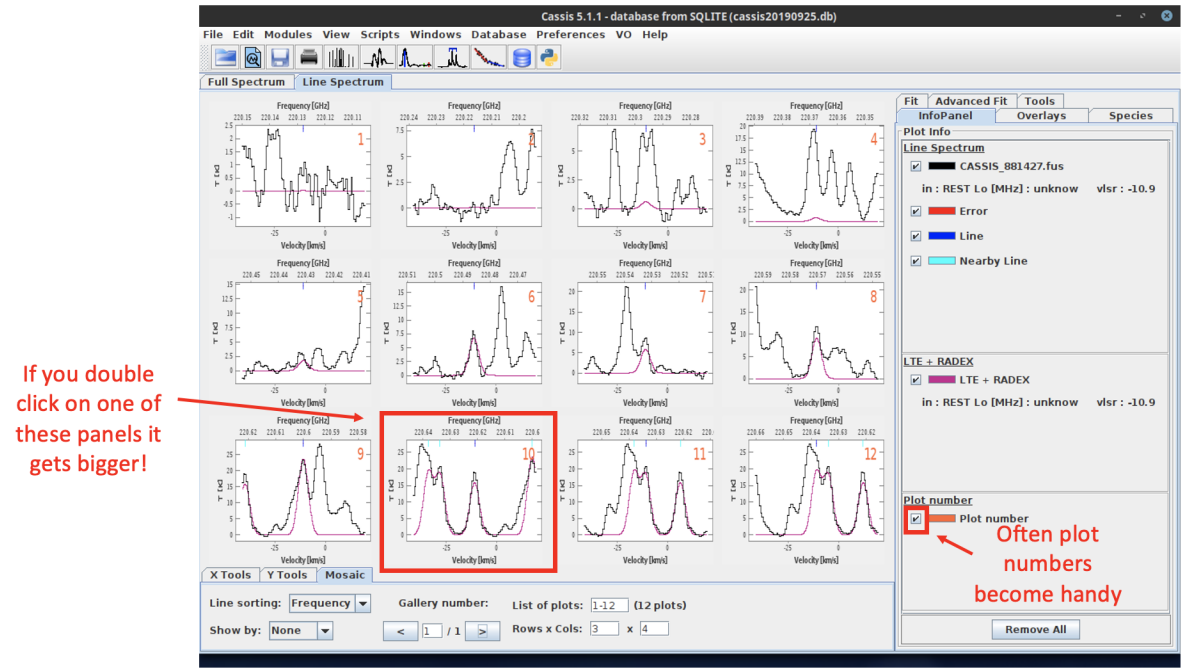

Figure 6: The output of Line Analysis.

centred on the transitions of all the selected species that are covered in the data. Over plotted in purple is the LTE model calculated based on the values entered for N, T, FWHM and source size (Fig. 6). You can then play with these parameters to find the best-fit model. If you double click on a panel in Fig. 6 you get Fig. 7. This method is incredibly valuable as a first check and can be used to get constraints on the physical parameters by eye especially if the spectrum of interest has many blended lines so grid fitting explained in Sect. 2.2.3 cannot be used.

You can save the configuration for your fitted model for easier access later. It is also possible to load an already saved configuration by clicking on 'Load config'. Moreover, you can always have more than one species selected when dealing with blended lines. For example, you can first fix N, T and FWHM of a molecule that has many lines (some non-blended lines) by first fitting for it. Then you can select another molecule that has some/all of its lines blended with the molecule(s) already fitted in the first step and fit for it so that the sum does not over fit the data. Furthermore, it is possible to deal with double peaked line profiles by fitting two components for an individual molecule. The second component can be added by clicking on the plus sign next to 'Component 1' in Fig. 5.

2.2.3 LTE RADEX

LTE RADEX, tool number 2 in Fig. 2, is very similar to Line Analysis but now instead of showing the model blown up for each line, it shows it for the whole spectrum. This is mainly useful after you have found the column densities and excitation temperatures of your desired molecules and would like to see how the fit for all the molecules together looks like and if you would like to change any of them in case of over fits.

Once you click on its icon a window pops up in which you need to set the range of frequencies that you would like the model to be displayed over. Again set the telescope file, required cut-off on $E_{\rm up}$ and $A_{\rm ij}$, set the $V_{\rm lsr}$ and choose the 'Molecules' by clicking on the preferred option in the drop down window. Once a molecules is selected from the list then you can again set the parameters of your fit for its N, T, FWHM and the source size. Once you click on 'Display' the model should show up (Fig. 8). You can again save the final configuration and access it later using the 'Load config' button.

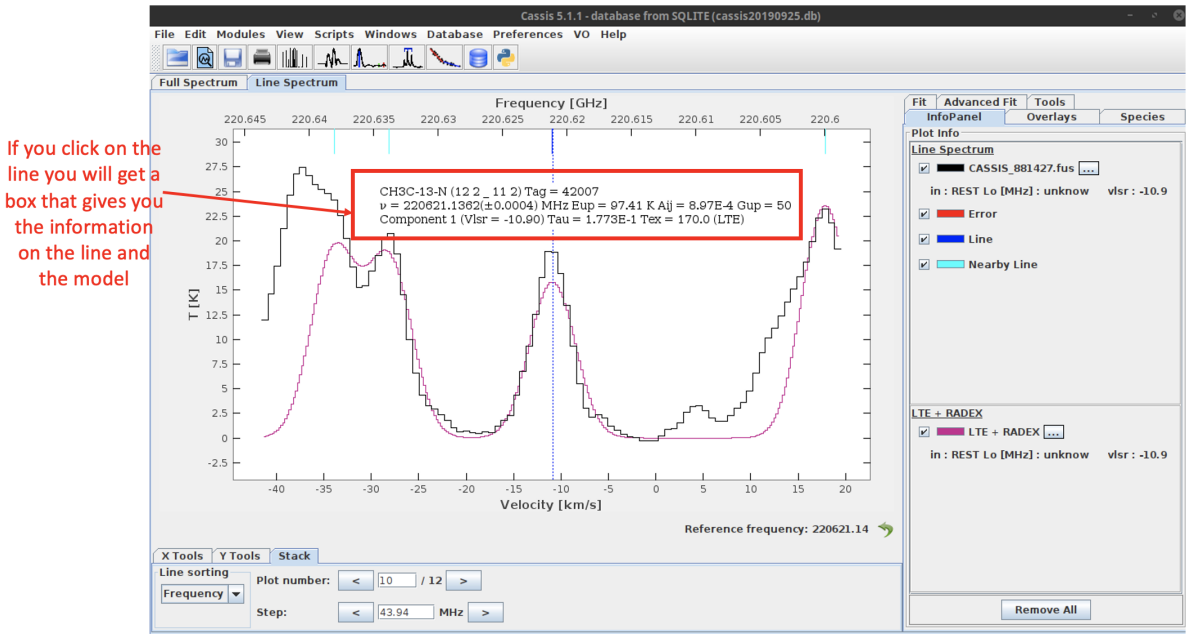

Figure 7: The output of Line Analysis zoomed in on one line.

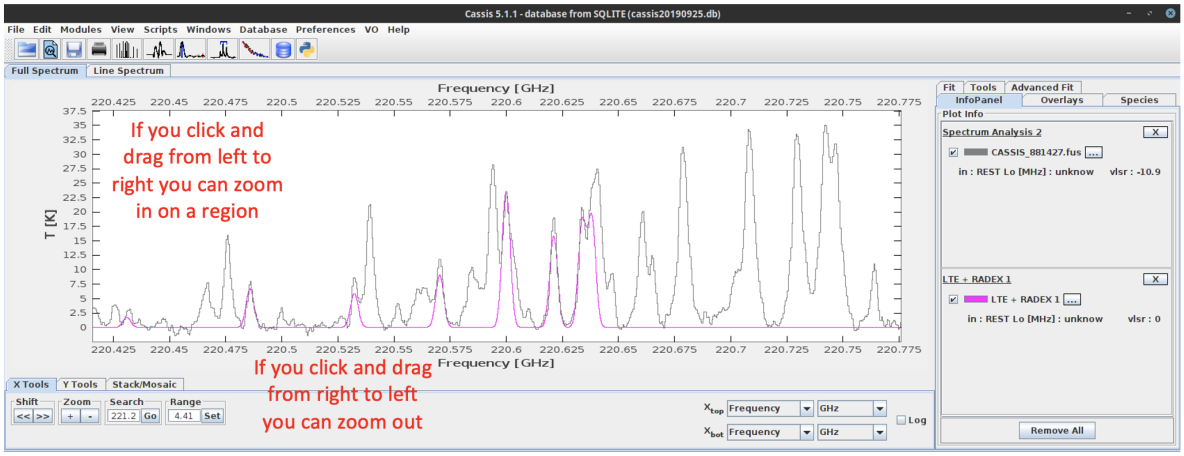

Figure 8: The output of LTE RADEX.

2.2.4 Jython scripting

This function, number 4 in Fig. 2, is useful when dealing with many molecules/sources so one can use CASSIS functions in a loop. Moreover, one can use this function to make a grid of column densities, excitation temperatures and FWHMs to calculate the synthetic spectrum at each point of the grid and then by comparing each model with the data one can find the best-fit model with the lowest χ^2 value. However, this is only useful if the lines are not blended for a source otherwise fitting by eye using the Line Analysis tool (number 3) gives more reliable measurements.

To use this function you need to write a Jython script (which combines JAVA and Python) with `.py` extension that does what you want to do. For example, you might want to do some grid fitting so you can write a script that takes a range of column densities and excitation temperatures as input and calculates the synthetic spectra for those ranges and finds the best model with the lowest χ^2. For such a script a few CASSIS functions become handy and it would be useful to import the following at the start of your script:

```
import ScriptEnvironment
from Range import Range
from Component import Component
from LineAnalysisScripting import UserInputs
from eu.omp.irap.cassis.properties import Software
from cassisStats import writeStats
from java.io import File
from Plot import Plot
```

The useful CASSIS scripting tools are:

- `UserInputs()`: This class defines the inputs. See CASSIS website for more information on how to use it.
- `Component()`: This function takes the column density, excitation temperature and FWHM range to define a model to simulate later. See CASSIS website for the full description of the function.
- After setting the inputs and defining the model parameters one can use
  `userInputs.computeChi2MinUsingRG()`
  to calculate the χ^2, you need to specify your defined component in the brackets here. For example
  `comp = Component(nmol=nmol,temp=temp,...)`
  and then
  `userInputs.computeChi2MinUsingRG(comp)`.
- You can then plot and save the best model using

```
lineModel = userInputs.plotBestModel()
lineModel.saveConfig(File( filename+".lam"))
bestPhysicalModels = userInputs.getBestPhysicalModels()
userInputs.saveBestPhysicalModels( filename+"bestModel.lis")
```

A set of files are provided with this cookbook in the folder (`CASSIS_grid_fitting_script`) that show examples of how CASSIS can be scripted.

2.3 Databases

As mentioned in Sect. 2.2.1 you need to make sure CASSIS is using the most updated database. To do that you can click on 'Database' in Fig. 2, then click on 'Database Settings' and change the Database Path to the most updated database available in the 'database' directory of CASSIS and make sure to click 'Save' afterwards.

2.3.1 Importing a pre-existing database

Another useful function in CASSIS is that you can import your own database. First you need to 'create a database' and then use it in the same way explained above by clicking on 'Database Settings' and

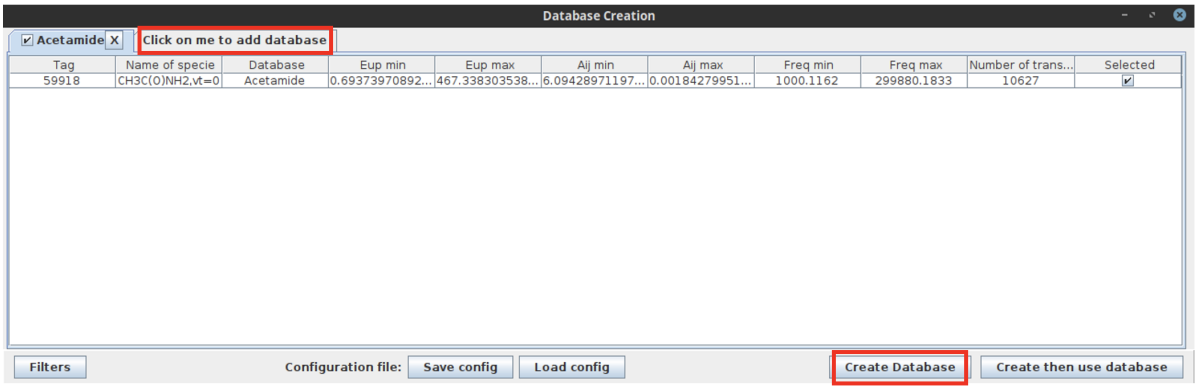

Figure 9: The window for creating a database.

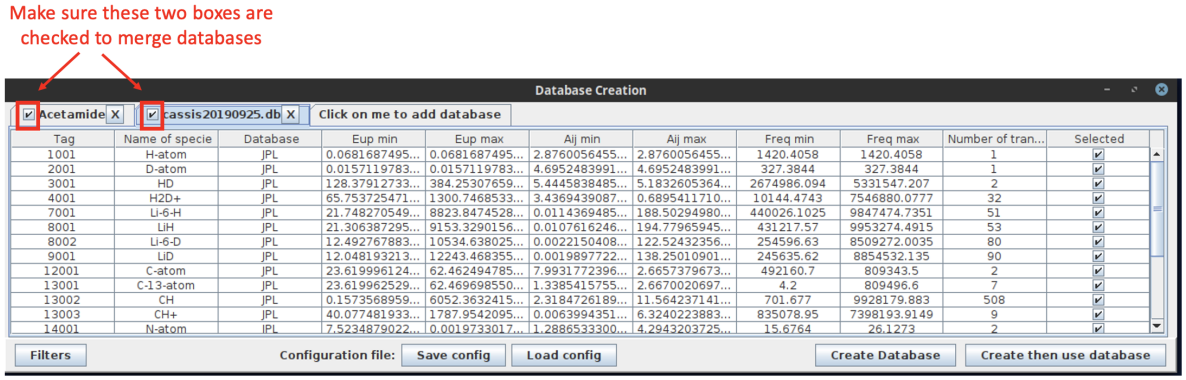

Figure 10: The window for merging two databases.

changing the path to it. To create a database you can click on 'Database' in Fig. 2 and then 'Database Creation', a window pops up (Fig. 9) in which you need to choose 'Click on me to add database', then change the 'Type' to whatever the type you have your database in. Here I explain how to do it if you select 'FILE'. Once the Type is selected you need to change the path to your 'FILE' with the new database and make sure the 'Format' that automatically is selected is correct (see Sect. 2.3.2 on how to create the 'FILE' containing your new database). Once that is done click on 'Validate' and check all the boxes of species you want to import and click on 'Import Selected'. You will then find a window similar to what is presented in Fig. 9. You can then click on 'Create Database' and save it where you want.

Tip 7: If you import a database using the method explained above you will have a database for one or a few new molecules separate from the main database and when looking at the spectra you need to keep changing back and forth between databases. This is not ideal and hence you can merge your new database which perhaps contains a few new species with the main database to create a complete one. To do that in Fig. 9 choose once more 'Click on me to add database', this time keep the Type as 'SQLITE' and then change the path to the main CASSIS database and click Validate. After that import all the molecules you care about. Then make sure the boxes next to both your new database and the CASSIS database are checked (see Fig. 10) before clicking on 'Create Database' and saving this merged database on your computer.

2.3.2 Preparing your own database

You can prepare the database to be imported as explained in Sect. 2.3.1 using Type 'FILE' by creating a folder (that you will need to set the path to when importing) including a .cat file and a partition function file. Let's say there is a new molecule on the CDMS or JPL website and is not yet included in CASSIS's database. Then you can find the .cat file on the CDMS or JPL website and you need to call it 'c0TAG.cat', where TAG is the species tag for example c058003.cat for Acetone.

For a CDMS entry you need a file called 'partition_function.html' which has its first two lines as:

```
tag    molecule    #lines    lg(Q(1000))  lg(Q(500))  ........  lg(Q(5.000))  lg(Q(2.725))
=============================================================================
```

then you need to put the value for each label underneath it. 'tag' is the tag of that molecule in the CDMS database, 'molecule' is what you want to call it, e.g. Acetone, '#lines' is the number of lines in the .cat file and the rest are the log in base 10 values for partition function at different temperatures. All these can be found from the CDMS website if the molecule is already on the CDMS database. In fact the 'partition_function.html' file should contain the first two lines shown on this website with an additional line for the molecule you are interested in importing. You can always copy paste the whole of what is on this website into your 'partition_function.html' and only import the molecule you want when asked to check boxes in CASSIS importing procedure.

For JPL entry you need to have your partition function in a file called 'catdir.cat' and the file can include one single line with the Species Tag, Species Formula, Number of Lines in your 'c0TAG.cat' file, $\log_{10}$ of the Partition Function values at different temperatures going from high to low temperatures and the Version. All these can be found at JPL catalogue directory. As an example for H2D+ you should have the following line in your 'catdir.cat':

```
4001   H2D+     32  1.8834  1.6986  1.4401  0.9882  0.4919  0.0846  0.0016  2*
```

Please note that you can again have a general 'catdir.cat' file with the partition function of many molecules each molecule on a single line and only choose your molecule of interest during the importing procedure in CASSIS. In fact the full 'catdir.cat' file can be downloaded from the bottom of this page.

2.4 Telescope files

A telescope file is needed in CASSIS to produce an LTE synthetic spectrum. The telescope file in CASSIS mimics the resolution of the observations in order to estimate the beam dilution (see §2.5). This section explains how one can create one.

A few template telescope files are already available in the cassis directory under 'delivery/telescope/'. However, ALMA can produce images in a wide variety of angular resolutions. In CASSIS, the angular resolution of the telescope is always tied to its 'diameter', under the implicit assumption that it is a single-dish telescope. For ALMA, the angular resolution (usually referred to as 'synthesised beam') depends on the *distribution* of baseline lengths in the configuration(s) that were used to obtain the image *as well as* several paramaters used to create the image, such as *uv* weighting and *uv* tapering. So to provide CASSIS with the correct information about the angular resolution of your ALMA data, do the following

1. Find the angular resolution in your ALMA data. It is stored in the header of the `.fits` file and, in CASA, accessible via the `imhead` command; in CARTA, get the beam info from `View - File Header - File information`. Let us assume the beam is $\theta''_a \times \theta''_b$, since in general ALMA does not have a circular but an elliptical beam.

2. Next, calculate the equivalent size of a circular beam θ_c, that has the same area (cf. eqn. 7),

$$\theta_c = \sqrt{\theta_a \theta_b}. \tag{1}$$

3. Now, calculate the equivalent size (diameter) D (in meters) of a single aperture that gives you the same effective resolution,

$$D = 1.22 \frac{\lambda}{\theta_c / 206264.806247}, \tag{2}$$

where the factor 206264.806247 converts θ_c given in arcsec to radians ($3600'' \times 180^\circ/\pi$), and λ is your observing wavelength.

4. Finally, copy the telescope file `alma_400m` to `alma_mydata` (or any other name of your liking, and replace the value of '400' in its second line to the value in m you found for D from the expression above. Select this new file as telescope file in CASSIS and you are good to go.

The script `make_cassis_files.py`, which is provided with this cookbook, takes as input the synthesized beam and frequency information from the header of a supplied `.fits` cube or alternatively provided as input parameters, and creates the telescope file `alma_`D where D is the equivalent size (diameter) in meters calculated according to Eq. 2.

2.5 Adopted source sizes, beam dilution, and summing columns over extended regions

CASSIS also requires you to provide a source size (in arcsec). As was the case for the telescopes files discussed in the previous section, this originates from the (implicit) context of single-dish observations with large beams in comparison to the source. The high resolution of ALMA means that the sources are not (always) smaller than the beam, so some counter-intuitive steps may be needed here.

For a source with a size θ_s smaller than the telescope beam θ_b (either single-dish or ALMA), the measured average intensity is reduced with respect to the true intensity by a factor F_dilution,

$$f_\mathrm{dilution} = \frac{\theta_\mathrm{s}^2}{\theta_\mathrm{s}^2 + \theta_\mathrm{b}^2}. \tag{3}$$

CASSIS corrects the reported column densities by $1/f_\mathrm{dilution}$ to ensure that values for different species, possibly obtained with different telescope beams, are directly comparable. For both single dish and ALMA observations, it is relatively straightforward to decide which source size to assume:

1. If the emission is extended on scales much larger than the telescope beam (well resolved), you want to have $f_\mathrm{dilution} = 1$. To enforce this, choose a source size at least 10 times larger than the telescope beam (because $10^2/(10^2 + 1^2) = 0.99 \approx 1$).

2. For unresolved emission (< the telescope beam), supply a reliable estimate of the source size (either from previous knowledge, or, for example, the expected size of the >100 K region where organics may evaporate with the water ice) *or, if this unknown,* a value 10 times the beam size, as above. In the latter case, you will only get a 'beam averaged' column density. Note that a fully *unresolved* source will *appear* to have the same size as the beam, because the beam spreads out the (point-like) emission over this size. Do not confuse this *apparent* source extent with its *true* size[4]

3. A source with an intrinsic size comparable to the beam, will appear to be extended over a size ~ 2 times the beam size. In that case, provide $\theta_\mathrm{s} = \theta_\mathrm{b}$ as the estimate of the source size. This will result in a correct beam dilution factor of $f_\mathrm{dilution} = 0.5$.[5]

Section 3 discusses how to correctly extract a spectrum from an ALMA image, including which area to average over (if any), so that CASSIS will give you the correct column density; the same guidelines also apply to maps obtained with a single dish telescopes. When you have images of extended emission, either from ALMA or from a single dish telescope, you may want to sum the column densities over this extended region and obtain the total amount of molecules. To do this correctly, follow these steps.

Case 1: You are using an *average spectrum extracted over an extended region*:

1. Run CASSIS to determine the column density, using $f_\mathrm{dilution} = 1$.

2. Multiply this column density with the area of the region in cm^2.

Case 2: You are using *a set of spectra, one for each pixel in the extended region*:

1. Run CASSIS to determine the column density at each of the pixels, using $f_\mathrm{dilution} = 1$.

[4] If you do supply the beam size as an estimate of the source size in this fully unresolved case, CASSIS will report column densities that are a factor 2 too large – cf. item (3). The only case where it is permissible to supply the beam size as the source size for unresolved emission, is when the spectrum of the unresolved emission was extracted (averaged) over an extended region the size of the beam. While the latter is incorrect (see Sect. 3), it introduces the same factor of 2 error that the (incorrect) beam-dilution factor of 0.5 will then fix – in a typical case of two wrongs making a right.

[5] If you want to be a bit more accurate, for sources that are marginally resolved with sizes of the same order of the beam, take an estimate of the extent of the emission in the image or map (θ_extent) and calculate the intrinsic source size θ_s from $\theta_\mathrm{s}^2 = \theta_\mathrm{extent}^2 - \theta_\mathrm{b}^2$.

2. Multiply each column density with the area of the *pixel* in cm^2.

3. Sum these values for each pixel together *and divide by the number of pixels per beam.*

The last step in this case corrects for the fact that adjacent pixels do not contain independent values, and that only measurements that are a beam apart are independent. You can calculate the number of pixels per beam from

$$N_{\rm pixels} = \frac{\Delta\Omega_{\rm beam}}{p^2} = \frac{\pi/(4\ln 2)\,\theta_a\theta_b}{p^2}, \tag{4}$$

where θ_a, θ_b are the FWHM beam size (assuming the general case of an elliptical beam; a circular beam has $\theta_a = \theta_b$), and p is the pixel size. Make sure the express θ_a, θ_b and p in the same units (e.g., arcsec).

2.6 Format of CASSIS spectra

The format of CASSIS spectra are not as easy as a column with the frequency of the data and a column with the intensity. Instead it has some additional columns that are zeros for the data but they need to be included. Before giving you the exact format, it is good to add two words of caution. First, the intensity should be in antenna-temperature units of Kelvin. Section 3 describes how to convert from other units (such as Jy beam^{-1} for ALMA data) to K. Note that the conversion always applies the Rayleigh-Jeans approximation of the Planck function, by definition (cf. this useful document). Second, the spectrum needs to be in rest-frequency. So you need to shift your spectrum from sky- to rest-frequency with a script of your own using the $V_{\rm lsr}$ of your source before producing the CASSIS spectrum (see last step in Sect. 3). The $V_{\rm lsr}$ is saved in the CASSIS formatted `.fus` file and is mostly used for book-keeping in CASSIS when the spectrum is loaded. Having the spectrum in the rest-frequency frame, overlays of fitted spectra or species will automatically also be in rest frequency frame. Marked species can be shifted with respect to this frequency by adjusting the 'plot' parameter under the 'Species' tab (see the right panel in Fig. 4)

Now that you have the `ShiftedFreq` and your intensities (`I`) in units of Kelvin you can use the following line of code in Python to produce your spectrum

```
nlines = len(freq)
outfile = open('YourFileName.fus', 'w')
outfile.write('// number of line :'+str(nlines)+'\n')
outfile.write('// vlsr : '+str(YourSourceVLSR)+'\n')
outfile.write('FreqLsb VeloLsb FreqUsb VeloUsb Int DeltaF DeltaV \n')
for lin in np.arange(nlines):
    outfile.write('0:.6f\t0\t000.00\t000.00\t{1:3.10f}\t0.0\t0\n'
    .format(ShiftedFreq[lin], I[lin]))
outfile.close()
```

The first few lines of an example CASSIS file are as follows:

```
// number of line :120
// vlsr : -52.0
FreqLsb VeloLsb FreqUsb VeloUsb Int     DeltaF  DeltaV
256071.210585   0       000.00  000.00  0.4074870403    0.0     0
256073.163538   0       000.00  000.00  0.5281615259    0.0     0
256075.116490   0       000.00  000.00  0.4379369649    0.0     0
256077.069443   0       000.00  000.00  -0.0320003526   0.0     0
256079.022396   0       000.00  000.00  0.0434240776    0.0     0
256080.975349   0       000.00  000.00  0.0056657517    0.0     0
```

The script `make_cassis_files.py`, which is provided with this cookbook, takes as input the spectrum extracted from an ALMA data cube in Kelvin (see §3), shifts the frequencies from sky to the rest frame using the provided source velocity, and creates a CASSIS-formatted `.fus` file ready to be loaded in CASSIS.

2.7 A few final tips

Sometimes you want to make a quick figure of your spectrum with your model but you don't want to write an extra short script to do it. For that you can use CASSIS to produce a pdf of your spectrum. You can load a spectrum or use the Line Analysis or LTE RADEX tools (Sect. 2.2.1, 2.2.2 and 2.2.3) to produce a figure of the spectrum with a model on top. It might be useful to know that you can save these plots as pdf files by clicking on 'File' in Fig. 2 and then 'Save', choose where you want to save the figure and then make sure 'File of Type' is set to .pdf format and then click 'Save'.

A table that is usually included in observational papers with spectral analysis includes line transitions of the molecules you are studying in the frequency range of your data. You can get these transitions by first loading in your spectrum (Sect. 2.2.1) and then using the 'Species' tab on its right hand side (Fig. 4) check the box of the molecule you are interested in and then click on 'File' in Fig. 2, then click on 'Save lines' and save the transitions where you like. This creates a file including the frequency, upper energy level and A_{ij} of the transitions of the molecule of interest which can then be read in a separate python script to produce the LaTeXtable you need.

3 Extracting a spectrum from ALMA data for CASSIS

3.1 Starting with ALMA data

Before you can run CASSIS you need to obtain a spectrum in a file format that CASSIS can understand. This section describes how you can extract such a spectrum in the correct way and in the appropriate format from ALMA spectral-line data. We will discuss two methods, one using CASA (Common Astronomy Software Applications[6]) and using CARTA[7]. Both methods can be used for data that you have stored locally on disk; CARTA can also be used via the ALMA archive[8] for pipeline processed data.

The starting point for your spectral-line analysis is an image *cube* obtained from continuum-subtracted ALMA data and appropriately deconvolved (cleaned). This cube will contain images at a series of channels covering a range in frequency space or, equivalently, velocity space for a given rest frequency. Each channel will have a channel width (defined in frequency or velocity) and each image will have a pixel size (arcsec). Each pixel contains the intensity of the emission at that location and frequency (velocity). The units of this intensity are Jy beam^{-1}, where 'beam' refers to the synthesized beam. This is defined as a two-dimensional gaussian, with FWHM $\theta_a \times \theta_b$ (in arcsec) and a P(osition)A(ngle) (in degrees, measured from North to East). Confusingly, this intensity is often referred to as 'flux', because it is expressed as a flux-per-beam, but it is important to realize it really is an intensity. The beam represents the angular resolution of the data. Usually, the pixel size will be 1/3–1/5 of the beam size, so that the beam is covered in 3–5 pixels. This means that adjacent pixels do not contain (fully) independent data!

As we will see below, extracting spectra for ingestion in CASSIS requires an appropriate choice of region to average yur spectrum over, and conversion from intensity units Jy beam^{-1} to intensity units K.

First the conversion from Jy beam^{-1} to K: for this conversion, we use the equation

$$T_b = \frac{c^2}{2k_B\nu^2} I_\nu = \frac{\lambda^2}{2k_B} I_\nu, \tag{5}$$

where T_b is the Rayleigh-Jeans equivalent brightness temperature in units of K, I_ν the intensity in SI units (W m^{-2} Hz^{-1} sr^{-1}), ν the frequency in Hz and λ the wavelength in m, c the speed of light, and k_B the Boltzmann constant. Because our data are in units of Jy beam^{-1}, the conversion becomes

$$T_b = \frac{c^2}{2k_B\nu^2} \frac{I_{\rm Jy/bm}\, 10^{-26}}{\Delta\Omega_{\rm beam}}, \tag{6}$$

where $I_{\rm Jy/bm}$ is the intensity in units of Jy beam^{-1} and $\Delta\Omega_{\rm beam}$ is the beam size in sr. For a gaussian synthesised beam of dimensions $a \times b$ in arcsec, you find the beam size in sr from

$$\Delta\Omega_{\rm beam} = \frac{\pi}{4\ln 2} \frac{\theta_a\theta_b}{(180\cdot 3600/\pi)} = \frac{\pi}{4\ln 2} \frac{\theta_a\theta_b}{206264.806247}. \tag{7}$$

Depending on how you extract the spectra, the conversion from Jy beam^{-1} to K is already done for you (CASA & new versions of CARTA) or needs to be applied later.

Second, the choice of region. Here you need to choose if you want to work with the spectrum in a single pixel or averaged over a larger region. If you choose to work with the spectrum in a single pixel, you get the spectrum *as measured in a synthesised beam* pointed at the location of that pixel. This is a sensible choice for unresolved emission and extended emission alike. If the emission is extended, you can also choose to extract the spectrum averaged over a larger region. This may increase the signal-to-noise ratio, but only if you average over a region that is at least two beams across. In the literature, you may come across cases where people averaged over a region the size of the synthesised beam. This introduces a factor 2 reduction of the signal if the emission is unresolved (dropping to a factor 1 if the emission is much larger than the beam and of uniform intensity; cf. Sect. 2.5). CASSIS can correct for this by adopting a source size equal to the beam size (Sect. 2.5), but this really is a case of two wrongs making a right, and is better avoided.

[6] https://casa.nrao.edu/
[7] https://cartavis.org/
[8] https://almascience.eso.org/aq/

3.2 Extracting a spectrum with CASA

To extract a spectrum using CASA and convert it to the CASSIS format, follow these steps.

1. **Open the image cube** with the `CASA viewer` by typing in the terminal

   ```
   > casaviewer afgl4176.contsub.image
   ```

   It should look like Figure 11.

   Alternatively, one can first open CASA and then open the viewer. To open CASA, if you have installed it yourself, you can likely open it by typing `casa` in the terminal. If you are using the Allegro computing resources[9], first log into one of our computing nodes (e.g. `helada`), then type `casapy` and press the ⇥ (tab) button to see all the available CASA versions. CASA versions below 6 can be opened by typing the `casapy-#` in the terminal (e.g. `casapy-570`). The most recent version of CASA on `helada` (version 6.2.1), can be opened by typing the following in the terminal

   ```
   > env -u PYTHONPATH -u LD_LIBRARY_PATH casapy-621p
   ```

   After CASA opens, the viewer can be opened by typing `imview` in the CASA terminal. Once the viewer has opened, the image can be loaded as a raster image via `File > Open` or by clicking the leftmost icon in the toolbar (see Fig. 11).

2. **Mark a region** from which you want to extract the spectrum. You can either extract the spectrum at a single pixel ('point region') or the *average* spectrum over an extended region ('box', 'oval' or 'polygon'). The single-pixel option will give you the spectrum averaged over the synthesized beam at the location that you select; the other options give you the spectrum averaged over a larger region which you define. It makes no sense to define an extended region smaller than, or comparable to, the synthesized beam: these pixels do not contain independent information and you may accidentally end up introducing a factor 2 error in the derived column densities (cf. Sect. 2.5). You are strongly advised to either use a 'point region' or define an extended region that is at least two synthesised beams across.

   To select a region, use one of the four buttons in the `casaviewer` toolbar as shown in Fig. 11. From left to right, the buttons allow users to mark a point, a box, an oval, or a polygon. To mark a point region, left or right click the point symbol, then click on the image on the pixel where you want to extract the spectrum. To mark an extended region, click one of the other three buttons, then click on the image ad drag to draw the region. After you create a region, you can move or resize it. To delete a region, click on the region and press the escape key on the keyboard.

3. **Plot the spectrum**: Click on the spectral profile tool button (see Fig. 1) to see the spectrum at the pixel where the point region has been marked (or the averaged spectrum in the case of a box/oval/polygon region). If you marked a pixel somewhere near the center of the image, you should see a spectrum similar to that shown in Figure 12. Moving the region around or resizing it will update the spectrum that is shown. The cyan vertical line corresponds to the channel whose image is shown in the casaviewer. By clicking the play or next/previous buttons under *Animators* in the CASA viewer, you can see other planes in the data cube.

4. **Convert the axes units**: CASSIS requires the spectral axis to be in units of GHz and the intensity in Kelvin. Click on the drop-down menus for the 'Bottom' and 'Left' axes in the Spectral Profile viewer below the spectrum and change the bottom axis units to frequency (GHz) and the left axis units to Kelvin (see Fig. 12).

5. **Save the spectrum**: click on the save button in the Spectral Profile Tool (see Fig. 12), specify a name for the file, and save the spectrum in ASCII format.

6. **You can now exit CASA viewer.**

[9] Email us to gain access to the Allegro computing resources.

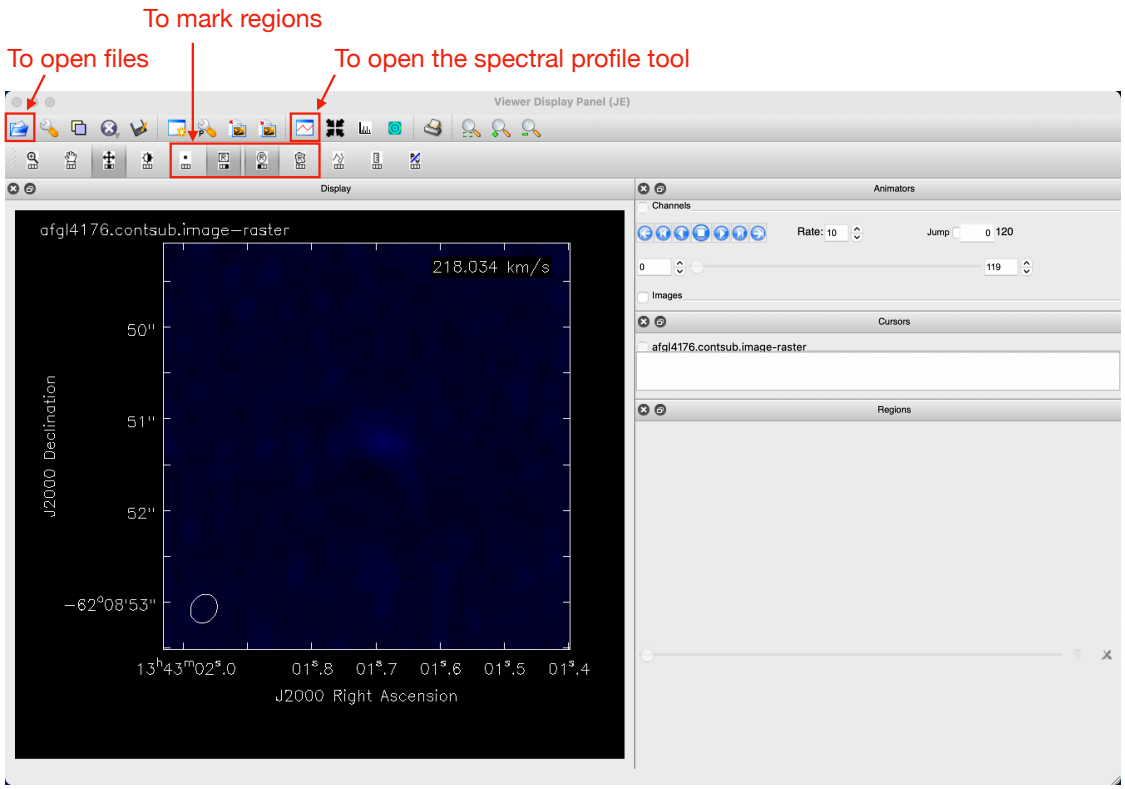

Figure 11: The CASA viewer.

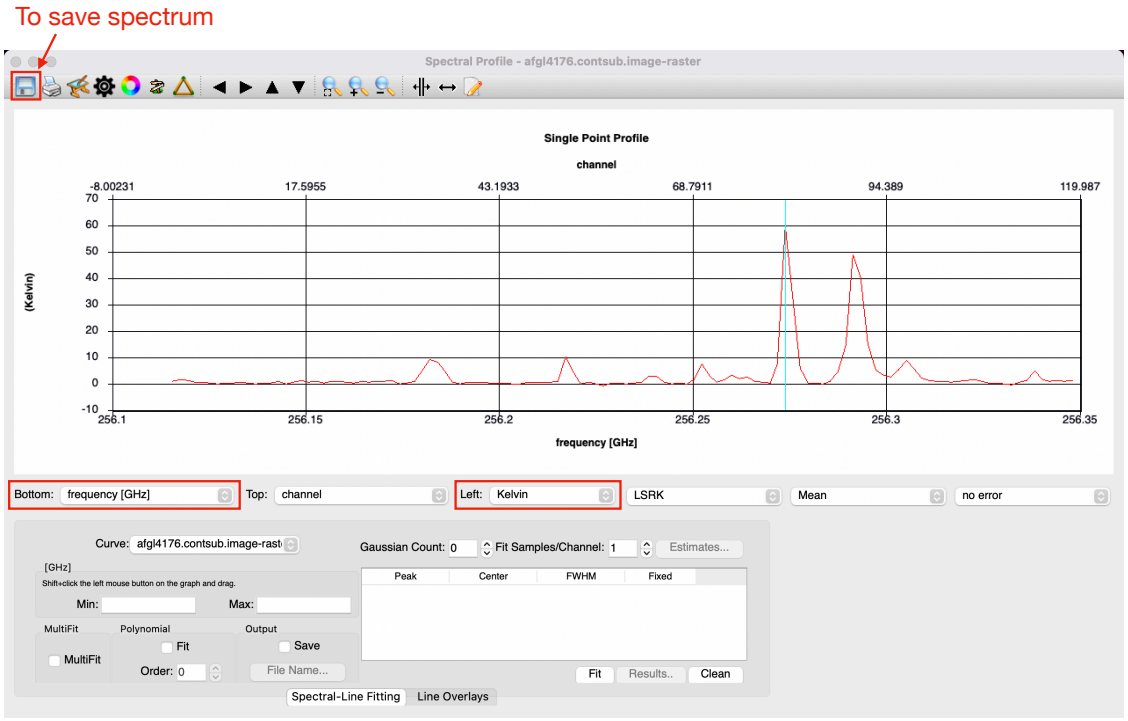

Figure 12: The Spectral Profile Tool in CASA viewer showing the spectrum averaged over the region of choice in units of Kelvin with the frequency axis in units of GHz. The red boxes mark the drop-down menus that convert the units for the bottom and left axes.

7. **Convert spectrum to CASSIS format**: Download the script `make_cassis_files.py` from the Allegro website, edit the following lines at the top of the file to your desired values:

```
# To shift the frequencies from sky to rest frame:
# ------------------------------------------------

# path to extracted spectrum
path = '/path/to/folder/'
# name of extracted spectrum (space- or tab-separated ascii file)
file_spectra = 'afgl4176_spec_casa_extracted.txt'
# source velocity
vlsr = -52 * u.km/u.s
# desired name of output (CASSIS) file
outfile = 'cassis_spectrum.fus'

# To make the telescope file:
# ---------------------------

# name of FITS cube from which the data was extracted (optional)
file_fits = 'afgl4176.contsubs.fits'
# if you do not have a FITS cube, provide the angular resolution
# Note that the information below is ignored if you supply a FITS file above
frequency = 2.563020350000E+11 * u.Hz  # frequency of the observations in Hz
beam_major = 0.33 * u.arcsec   # beam semi-major in arcseconds
beam_minor = 0.28 * u.arcsec   # beam semi-minor in arcseconds
```

and run the script from terminal in same folder:

```
> python make_cassis_files.py
```

The file with the `.fus` extension is now ready to be imported into CASSIS for further analysis.

3.3 Extracting a spectrum with CARTA

An alternative way to inspect ALMA data cubes and extract spectra is by using the CARTA tool[10]. The advantage of CARTA is that you can also access data products from the ALMA archive directly without the need for downloading data. To access data on disk with CARTA, first start CARTA on your computer and then choose `Open image` (a file browser might already open when you start CARTA); load the `.fits` file that you want to open. To use CARTA to access an ALMA archive data set without downloading, go to the ALMA Science Arachive at `https://almascience.eso.org/aq/`, search for the data set of your interest, add a check mark and click the `Explore and download` button on the top right (see Fig. 13). If you select the 'OLD' method, a new page will open that shows the structure of the data set. To see the available data products, click the triangle next to the 'product' folder. If you select the 'NEW' method, a pop-up in the same window will appear where you can select any `.fits` file that you want to explore. From the overview of the data products, open any data product with CARTA by clicking on the CARTA symbol . (If you click on the file name itself, the fits file of the data product will download.)

To extract a spectrum from a cube using CARTA and convert it to the CASSIS format, follow these steps (see Fig. 14):

1. **Open the file with the cube**. The first channel will now display. Generally, since this will be continuum-subtracted data, the first channel may only contain noise. Use the `Animator` widget to step through the channels to find the region(s) from which you want to extract a spectrum.

[10] `https://cartavis.org`

2. **Define region (point/box/oval/polygon)** using the buttons above the image display. You can draw multiple regions and activate a region by clicking on it.

3. **Plot the spectrum**: press the button marked `z`. The spectrum associated with the active region will now pop up. If the region is ever deactivated, you can also re-select it from the drop-down menu in the spectrum window (`Z Profile`) under 'Region'.

4. **Convert the axes units**: in the spectrum window (`Z Profile`) press the 'wheel' (settings) symbol, choose the 'Conversion' tab and set 'Coordinate' to 'Frequency (GHz)'. This should already be the default, but this is how you change it if it is not. Also convert the 'Intensity unit' to 'K' to make the conversion from Jy beam^{-1} to Kelvin[11].

5. **Save the spectrum to a file**: Press 'export data' (list symbol) at the bottom right, select a name for your file, and change the extension to `.txt` or `.tsv`. Note that you may need to hover over the spectrum for the button to appear.

6. **You can now close CARTA.**

7. **Convert spectrum to CASSIS format**: Download the script `make_cassis_files.py` from the Allegro website, edit the following lines at the top of the file to your desired values:

```
# To shift the frequencies from sky to rest frame:
# ------------------------------------------------

# path to extracted spectrum
path = '/path/to/folder/'
# name of extracted spectrum (space- or tab-separated ascii file)
file_spectra = 'afgl4176_spec_carta_extracted.txt'
# source velocity
vlsr = -52 * u.km/u.s
# desired name of output (CASSIS) file
outfile = 'cassis_spectrum.fus'

# To make the telescope file:
# ---------------------------

# name of FITS cube from which the data was extracted (optional)
file_fits = 'afgl4176.contsubs.fits'
# if you do not have a FITS cube, provide the angular resolution
# Note that the information below is ignored if you supply a FITS file above
frequency = 2.563020350000E+11 * u.Hz  # frequency of the observations in Hz
beam_major = 0.33 * u.arcsec   # beam semi-major in arcseconds
beam_minor = 0.28 * u.arcsec   # beam semi-minor in arcseconds
```

   and run the script from terminal in same folder:

```
> python make_cassis_files.py
```

   The file with the `.fus` extension is now ready to be imported into CASSIS for further analysis.

[11] This is a new feature in CARTA, so if you do not see this option, upgrade your CARTA installation.

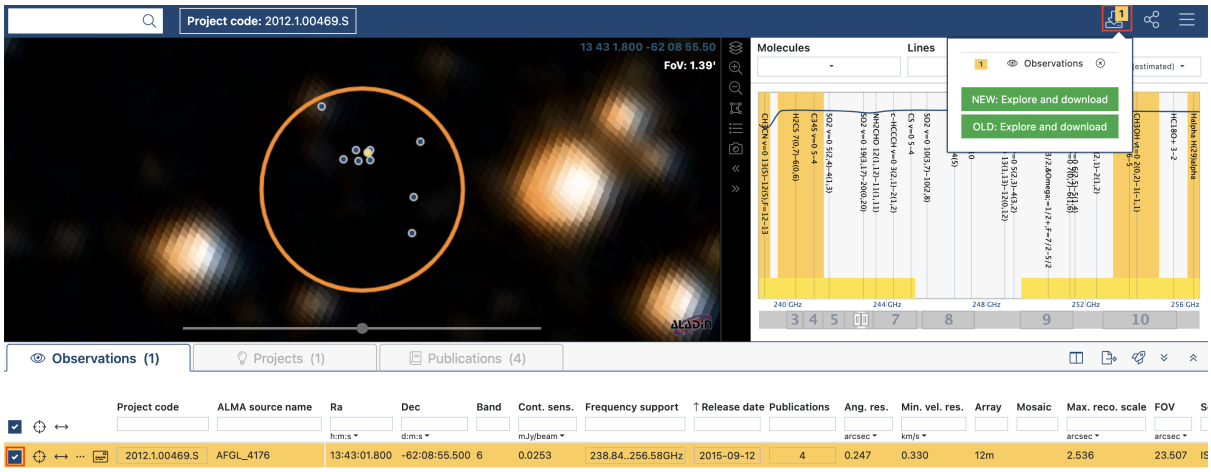

Figure 13: The archive interface, highlighting the data selection tickbox on the bottom left and the Explore and download button on the top right.

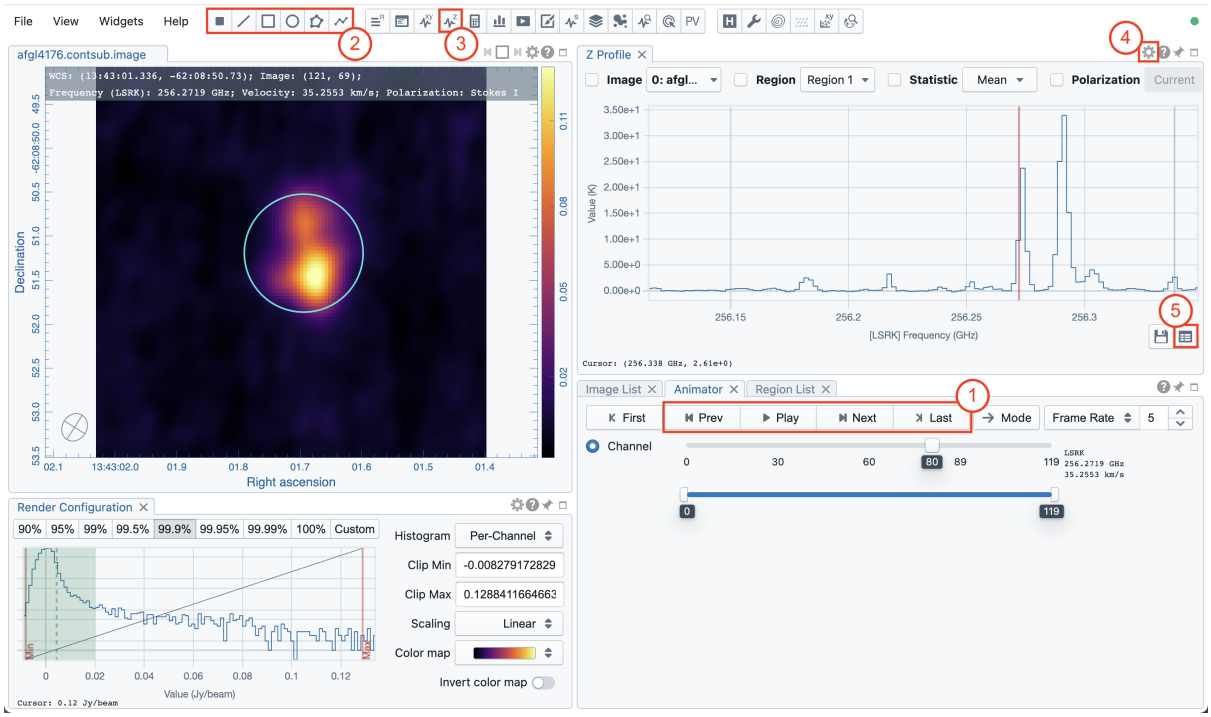

Figure 14: The CARTA interface showing the spectrum averaged over the region of choice in units of Kelvin with the frequency axis in units of GHz. The numbers correspond to steps in Sect. 3.3.