esac

European Space Astronomy Centre (ESAC)
P.O. Box, 78
28691 Villanueva de la Cañada, Madrid
Spain

DOCUMENT

# Dealing with extended sources observed with PACS Spectroscopy

| | |
|---|---|
| **Prepared by** | **Katrina Exter** |
| **Reference** | **HERSCHEL-HSC-DOC-2314** |
| **Issue** | **1** |
| **Revision** | **0** |
| **Date of Issue** | **01/05/2018** |
| **Status** | **For release** |
| **Document Type** | **Release Note** |
| **Distribution** | **HSC, SAT** |

# CHANGE LOG

| Reason for change | Draft Issue | Revision | Date |
|---|---|---|---|
| First version of document | 1 | 0 | 01/05/2018 |
| | | | |
| | | | |

## Table of Contents

# 1 INTRODUCTION

*This user note is to explain how to work with extended sources with PACS spectroscopy.*

Extended sources are any sources larger than a point. The analyses possible on extended sources observed with PACS spectroscopy are the same as those for any source observed with an integral field spectrograph – you can, for example, fit emission lines to make intensity maps, or extract the spectra from apertures and analyse those as 1d spectra. This user note is rather about some issues to be aware of – and in some cases to correct for – when dealing with the integral field spectroscopy that PACS produced. The most important issues to consider are (i) the spatial sampling and (ii) the flux calibration.

**The spatial sampling** affects how well you can interpret your observation, both spatially and spectrally. This user note explains what the uncertainties are.

**The accuracy of the flux calibration** of PACS spectroscopy depends on the target. For a completely flat source, all of Level 2 and above products are fully calibrated. The spectra of point sources require an extra correction; a point-source calibrated spectrum is also provided at Level 2 and above, although users should read the document on Dealing with Point Sources observed with PACS Spectroscopy to know how (and whether they can) use this product. For semi-extended sources (< 25—30″ at their fullest extent), read the document Dealing with Semi-Extended Sources observed with PACS Spectroscopy to learn about the flux corrections to apply to the Level 2 and above cubes. For extended sources that are larger than this, and which are not fully flat in the observed field, corrections to the flux calibration are also necessary, and these are explained in this note.

*The package of HIPE scripts that accompany this user note* can be found in (HELL: PACS section, Level 2, specifically here). The script includes an explanation of what it is doing, and the images and values that are included in this user note are taken from the running of that script. The example source in the script is a point source. "Why a point source?", you may ask, "this note is about *extended* sources?". Two advantages of working with a point source make this the most useful example we can provide: (1) the model of a point source is very easy to create, and (2) as we can compute the true flux of the point source – using the point-source corrections – and this provides a fixed comparison for the extended source correction results.

The process that will be explained here is a *forward modelling*: we say what we think the observed source *does* look like, and produce an example of what that source *should* look like when observed with PACS and its pattern of illumination set by the beams of each part of its detector. It is the difference between the output and input that is your extended source correction. As this correction depends on the shape of the source, the cube used, and the pointing pattern of the observation, this is a *unique Extended Source Correction: uESC.*

Before reading this user note you should be familiar with PACS spectroscopy, in particular with the various observing modes that were used and different types of cubes that were produced. These are explained in the Quick-Start Guide, and in more detail in the PACS Products Explained and the PACS Handbook, all of which can be found in the Herschel Explanatory Legacy Library (HELL: PACS section). To follow the advice given in this release note, the user should have at least read the Quick-Start Guide, and should know what observing mode their observation was taken with and hence where to get their cubes from (i.e. Level 2 or Level 2.5). Some definitions are provided at the end of this release note, but to *understand* the terms defined, it is necessary to read the other documentation. Additionally, the tasks discussed in this note are explained, and their parameters defined, in the Help documentation available from HIPE, in particular the PACS Data Reduction Guide (spectroscopy) and the Users Reference Manual.

Additional material can be found in the form of video tutorials on the YouTube *Herschel Academy* channel. There is one tutorial on extended sources and one that explains the different PACS cubes – both are highly recommended to the readers of this note. There is no tutorial that demonstrated the Forward Modelling script, however.

## 2 PRE-REQUISITES

The first pre-requisite is that the user knows the observing mode of their observation. This is important when discussing which products to use and create, what issues to be aware of, and the more general calibration uncertainties. This information can be obtained from various sources: the FITS header keywords of the final-level cubes in an observation, a spreadsheet containing these keywords for all observations, and the PACS Observation Summary which is provided in all observations (but can only be read in HIPE; although it may soon also be provided on the HSA search results page). How to find this information is explained in the PACS Products Explained.

The second pre-requisite is that the user knows how to open a cube in HIPE, and they are willing to work in HIPE. The HIPE software has been provided not only to reduce Herschel data, but also to inspect and analyse Herschel data. The Spectrum Explorer and Standard Cube Viewer are especially useful for working with cubes.

*The third and most important pre-requisite* is that the user has a pretty good idea of the surface brightness morphology of the source, i.e. its location and shape on the sky. If the source is more complicated that a circular Gaussian, it will be necessary to code the equation describing their source shape into HIPE following the few examples given. The user also needs to know what the source spectrum is, however there is a workaround for those who do not know.

# 3 SPATIAL SAMPLING AND RESOLUTION

The spatial resolution of PACS spectroscopy data is fixed by the FWHM of the PACS optical beam, and the way it is "imaged" by the detector (the PACS detector was in fact 25 separate mini detectors, each with their own beam). Information about the beam(s) can be found in the Quick-Start Guide and the PACS Handbook. The *sampling* of the beam is set by the observing mode. The native spaxel size of the PACS IFU is 9.4″, and for pointed observations the spatial grid is that of the instrument itself: a slightly irregular 5x5 grid of these spaxels. This grid is also described in the above-mentioned documentation (as well as in the video tutorials on the *Herschel Academy* channel of YouTube) and shown in Fig. 1. The spatial sampling is therefore fixed to this 9.4″ for pointed and tiling observations (a tiling observation being one where the offset between pointings is designed to image a large field rather than the improve the sampling). For mapping observations – rasters with offsets between individual pointings of a few to two dozen or so arcsec – the spatial sampling is higher because the cubes from each pointing in the raster are mosaicked together by tasks in the pipeline, to create a new cube. Users were free to choose any raster pattern, but the three recommended modes that were followed by most users – these can be found in the PACS Products Explained – had steps of 3″ to 24″ in patterns of 2x2 or 3x3. In that document you can also find an explanation of the different sampling names: *Nyquist-sampled*, *oversampled*, and *undersampled* observations.

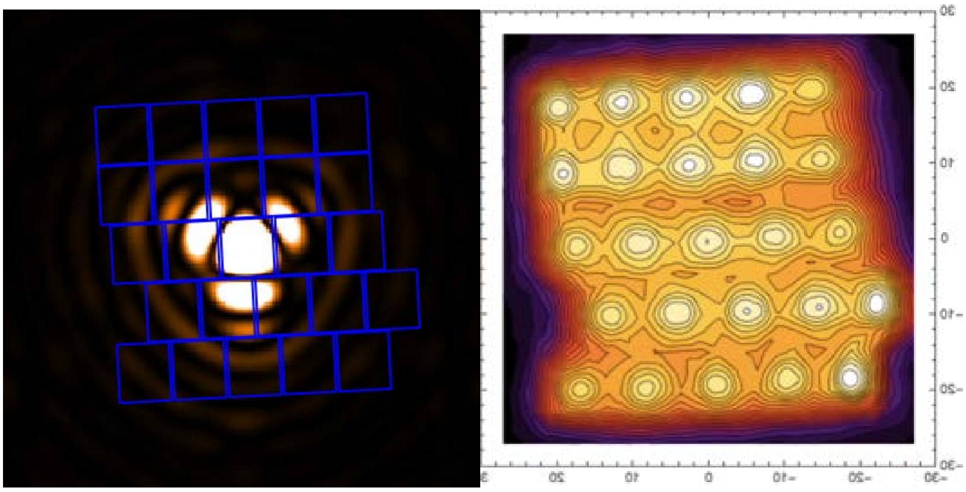

**Figure 1 Left:** The PACS beam at 75 µm with the footprint of the PACS integral field unit. **Right**: the highly oversampled spatial response of the PACS IFU (blue camera) to a uniform illumination pattern

The sizes of the spatial pixels of the mosaic cubes produced by the pipeline and provided in the observations downloaded from the HSA are:

- *9.4″ for all re-binned cubes*, which are provided for all observations and are the input to the mosaic tasks.
- *3″ for all interpolated cubes*, whether from pointed, tiling, or mapping observations. Note that for these *undersampled* observations (i.e. where the spatial resolution does not at least Nyquist sample the beam), the actual spatial resolution is worse than this, as it is defined by the beam of the (nominally 9.4″) spaxels; however, interpolated cubes are provided for these observations so that the user has a cube with a regular spatial grid (which the re-binned cubes do not), and because it is easier to "see" your target in these cubes than in the re-binned ones.
- *0.5″ for projected cubes from pointed or tiling observations*. This was also done so that the cubes for undersampled observations can have a regular spatial grid, but in this case in such a way that you can see the original spectral sampling – the 9.4″ spaxel shape – in the surface brightness pattern.
- 3″ *for projected cubes from mapping observations*, from the Nyquist-sampled and oversampled, and line- and range-scan, modes.
- *Less than or about 3″ for mapping observations* that contain both drizzled and projected cubes in Level 2/2.5. These will be line-scan observations with the Nyquist-sampled or oversampled mapping modes. For these observing modes, the drizzle task of the pipeline creates cubes with a spatial grid that is optimised to the FWHM of the beam at the wavelength of the data, so the spatial pixels are smaller for shorter wavelengths. That same grid is given to the projected cubes; this being done so that the users can directly compare these two types of cubes that are provided for these observations.

*For Nyquist- and oversampled-mapping observations, the size of the spatial pixel in the mosaic cube indicates the spatial resolution you can expect from those data. For tiling and pointed observations, the spatial resolution is the size of the native spaxels.*

Some examples of these different types of cubes are given in Fig. 2. Scripts have been provided in HIPE to allow the user to create any of the mosaic cubes with any spaxel size, starting from the Level 2/2.5 re-binned cubes (so there is no need to re-run a complete pipeline).

What is important to note is that different types of cubes (re-binned, interpolated, etc.) are suitable for different types of observation: line- or range-scan; pointed or one of the mapping types. Two cubes (plus the re-binned) are provided for most observations in the archive, and generally both are suitable for science: which you use depends on which you personally prefer. The PACS Products Explained should be consulted for more information.

*But more importantly than this is the fact that you have to be careful when interpreting your observation*. The spatial resolution is not the main limiting factor in the spatial appearance of your source. It is the *combination* of the spatial resolution with the shape of the detector (the spaxel grid, with its uneven illumination pattern, explained in Sec. 4) and with the mapping pattern of the observation (the grid created by the overlap of the individual grids in the raster). Please take this into account when working with your data. Looking at the image slices shown in Fig. 2, it is clear that (i) the 9.4″ native spaxel affects the appearance of *all* of the cubes, even the two taken from a mapping observation, and (ii) the appearance of the source is not exactly the same in the different mosaic cubes, and so it is a good idea to inspect all the mosaic cubes provided before deciding which to use and how far to interpret its appearance.

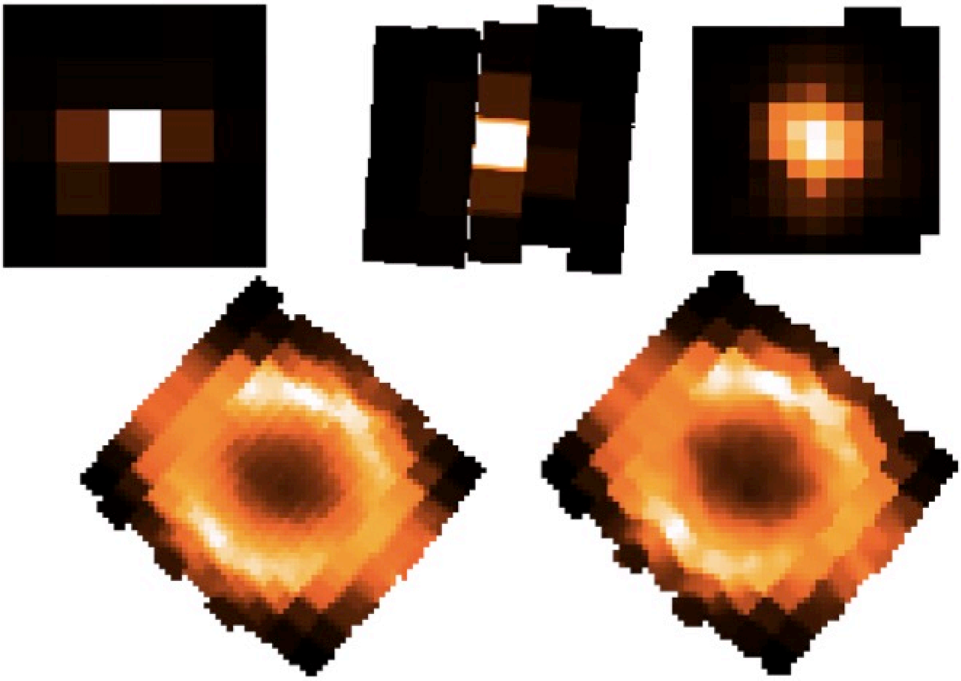

**Figure 2 Upper** (left to right): image slices from a re-binned, projected, and interpolated cube of a point source observed with a pointed observation. **Lower** (left to right): an image slice from a projected and drizzled cube of an extended source from a mapping observation. Cuts are always from the same wavelength.

# 4 FLUX CALIBRATION

## 4.1 What sources the standard flux calibration is good for

The flux calibration of PACS returns correct fluxes in all cubes for the following types of source:

- Fully extended sources – with a flat surface brightness distribution of the FoV of a single pointing (i.e. for a single re-binned cube of a pointed observation, or in each individual re-binned cube of a mapping observations), with a slope (change of surface brightness) of no more than about 20% in the FoV
- Point sources after the point-source calibrations are applied
- Small sources (<25—30″ full size) after the semi-extended source calibrations are applied

For any other sources – crowded fields or irregular/steep/blobby extended sources – the flux calibration is not fully correct.

The reason for this is to do with the spatial response of the PACS IFU to the in-falling light. This is explained in the spectroscopy calibration chapter of the PACS Handbook. If the PACS IFU were a perfect detector, each spaxel would have a 100% response across its entire face and the spaxels would be perfectly contiguous. In reality, the response of the spaxels was not 100% across their face, and this effectively means that they are not continuous: one can consider them to behave more like spaxels of 8″ with small gaps between them. This is illustrated in Fig. 3.

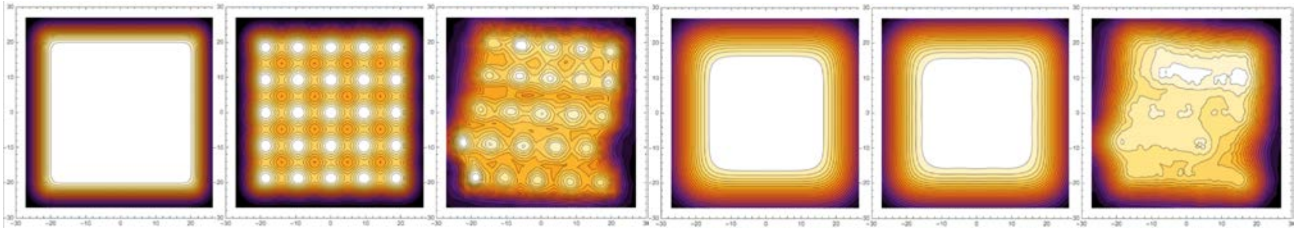

**Figure 3** Maps of the PACS FoV at 62 µm (left) and 145 µm (right). For each set of three, the **left** is the response pattern for "perfect" beams from each spaxel, i.e. spaxels of 9.4x9.4 arcsec with a flat response; **middle** is the response pattern for a 5x5 grid with 8x8 arcsec spaxels with a flat response; **right** are the measured beams.

The consequence of this non-perfect spatial response is that the *collected* flux from a source depends on where the source lies in the pattern, and hence to compute the true flux that *fell* on the detector, it is necessary to derive a correction that is based on the shape of the source and where it lies on the detector. This was possible to do in a generic way for a flat source (since its shape and location on the detector is uniform everywhere), and was possible to do for point and small sources (these corrections being among the first done by PACS, since point sources were used for the absolute calibration). Hence, fully flat sources are fully calibrated, and when subjected to the appropriate corrections, so are point and small sources. However, for any extended, randomly-shaped and randomly-located source, this was not possible to do as part of the data-reduction pipeline.

## 4.2 Flux calibrating irregular sources: the steps

So what can the average user of PACS spectroscopy data of an extended source do to estimate the true flux over the 3d field (1d spectral and 2d spatial) of their observation? How can you compute your own *unique* Extended Source Correction (uESC), to replace the Extended Source Correction used in the pipeline (pESC)? If you know (or can even estimate) the surface brightness distribution of your source – that is, its shape, sizes, position angle, and location – it is possible to estimate this uESC. A Forward Modelling [FM] Script has been written to do just this; it can be found on here on HELL (PACS; Level 2). The necessaries for this script are the following:

1. **A model of the source.** This model needs to be defined in as a class in python in HIPE. Examples of creating a model can be found in the user notes on semi-extended sources and point sources, and of course in the FM script itself. To learn how to modify the models, or find others, search the HIPE help documentation (e.g. in the Scripting Guide). There are two parts to the model: the spatial and the spectral. Note that these models are what you think your source *actually* looks like (*not* what PACS *shows* it to look like, since the PACS data are modified by the beams that are you actually trying to correct for).
   - **The spatial model**: What is the shape of the source, its radii, and its location in the field (in terms of the offset from the centre of the observed field)? Two models (Gaussian and flat) are included in the above-mentioned FM script. Developing a new model is a matter of inserting your own functions in the model class as written in the examples scripts, e.g. to turn a symmetrical Gaussian to an elliptical Gaussian is a matter of adding the position angle and radii as parameters in the modified Gaussian equation.
   - **The spectral model**. The total flux that the model source emits, over the wavelength covered by your observation, over the area of your input cube that your source subtends. Ideally this is the actual value of the flux of the source, but since it is possible that you don't know this, you can use a default spectrum – e.g. a continuum of 0 and a Gaussian of peak 1. The values of the fluxes – normally this would include the continuum flux level, slope, and the peak and width of the spectral line(s) – are input into along with the spatial details when the model is called. To learn more about how you can estimate these fluxes, see the further instructions below.
2. **The re-binned cube context from the observation you are modelling.** This needs to be a slicedRebinnedCube (HPS3DR[R|B: this is a context of one to several re-binned cubes) taken from the observation and of the wavelength range you are concerned with (do not use the mosaic cubes, even for mapping observations). The output from the main task of the script, pacsSpecFromModel, is also a slicedRebinnedCube context, with the same number of re-binned cubes therein as the input has. Each individual output cube has the same coordinates as its equivalent input cube. The centre of the observed field is stored in the metadata raNominal and decNominal of this input re-binned cube context, and this defines the location that the

spatial model offset values are measured from. (And although not stated in the on-line documentation for that task, it also is necessary that the cube be of the same wavelength range that you are modelling.)

With these data to hand, you can create a model of the source as it should look like, create a cube with simulated data that that model would show if it were observed with PACS, and by comparing the two, the spatial response of the detector – and thus the flux calibration – can be corrected for.

1. **The input.** The observed spectrum to which you eventually want to apply the correction should be extracted from the re-binned or mosaic cube taken from Level 2/2.5 in your observation, from which the pESC has first been removed. The spectral—spatial model is then created, on which the task pacsSpecFromModel is run.
2. **The simulated cubes.** The task creates re-binned cubes with the input model, where the data therein is the simulation of the input model and the beams taken from the calibration tree. If your observed spectrum was taken from a mosaic cube (which will be the case if you have a mapping observation, and may also be the case for pointed observations), you then create the same mosaic cubes from the simulated data.
3. **The correction.** A simulated spectrum is then extracted from this simulated mosaic cube, and compared to the input spectrum. The ratio between the two is the uESC and is applied to the observed spectrum to create a correctly calibrated source spectrum.

The FM script work for both mapping and pointed observations. The only difference is which type of mosaic cube to apply the correction to/extract the observed spectrum from. For this, you can to read the PACS Products Explained to see which cubes are recommended for which types of observation: we suggest that for pointed observations you can use the re-binned cubes, or the interpolated cubes with 3″ spaxels if it is easier to extract the spectrum of your source from that; for tiling observations you can use interpolated cube with 3″ spaxels; for mapping observations you can create projected cubes with 3″ spaxels; and for tiling observations with very large steps you can try projected cubes with 1.5″ spaxels.

*So to summarise*: you need to define a model of your source, showing what it looks like at the wavelength you are interested, spatially and spectrally. You then take the slicedRebinnedCubes (HPS3DR[R|B] in Level 2 or 2.5 of the observation) of the wavelength range you are interested in, and with this and the model, you create a model slicedRebinnedCubes and potentially a model mosaic cube from that. The data in these cubes represent your model *as modified by the PACS beams*. That is, what the model source would look like when observed with the PACS instrument, with the pointing pattern of your actual observation. The ratio the spectrum input (i.e. the model spectrum) to the spectrum output (as extracted from the cubes created by pacsSpecFromModel) is the modification of the data by the spatial response of the PACS detector (it will normally [but not always] be less than one), and by dividing the spectrum extracted from the observed cube by this, you are applying your uESC, i.e. creating a correctly-calibration source spectrum.

## 4.3 Flux calibrating irregular sources: what is the input model?

The model input to pacsSpecFromModel has to be a model, *not a model cube*. You have to extend the existing class ExtendedSpectralSource to add your source shape (e.g. a 2d symmetrical Gaussian) and spectrum (e.g. a continuum with a slope and Gaussian emission line with a certain peak and sigma width).

The class defines the model, and you call on it with the specific inputs necessary to create your model, e.g. how big is the source, where is it in the cube, how bright is the line? **However**, because the input to this process is a model, not a cube, it is *not* possible to compare a model input cube (what the source does look like) to a model output cube (what PACS shows the source to look like). You can only compare *information* taken from the input model and the output cube, e.g. the extracted spectrum.

## 4.4 Flux calibrating irregular sources: what can you compare at the end?

Since the input is not a cube, but the output is, it is not possible to compare the two directly. So how can you tell what the effect of the uneven illumination was on your observation? You can do three things:

1. Look at the output cube (after turning it into a mosaic cube), e.g. in the Spectrum Explorer or the Cube Viewer (or outside HIPE). You know what you defined the shape to be, so how different, qualitatively, is the source in your mosaic cube? Is the shape of the source the same in the observation and simulation?
2. If your input spectrum was an estimate of what the true spectrum of the source should be, then compare the simulated spectrum to the input spectrum they should be the same. If your input spectrum was a standard spectrum, this comparison is not useful.
3. If you are *sure* that your model is spatially and spectrally correct, and you used a correct spectrum as input, then the output model cube can also be compared to the observed cube.

All three comparisons are explored in the FM script.

What if you want to know how the shape of the source, i.e. the *spatial distribution* of the flux, is affected by the uneven illumination? Our suggestion is to input a model of a flat source (coding for which is included in the FM script) and to look at the spatial distribution of the flat source in the output (re-binned or mosaic) cube. The pattern you will get is effectively the response pattern of your observation. The absolute values are not useful, but the relative ones are. In this way you can at least see which parts of the FoV are most affected by the gaps between the spaxels.

The hardest parts of the script are defining the source model. Defining the spatial model is largely a matter of "do you know what the source shape is?" and then "can you code that into HIPE?" – this latter bit probably being the most technically challenging part. Defining the spectral model is technically easy (it not much more than a set of flux values), but may be scientifically problematic, although a standard spectrum can be used instead of a “real” one.

## 4.5 Forward modelling script: the observed and the input spectra

It is always necessary to extract the spectrum of your source from a cube in Level 2 or 2.5 (unchopped range-scan observations only) in the observation obtained from the Herschel Science Archive, from which you have first removed the pESC. This is the spectrum to which the correction – your computed uESC – is applied.

The spectral model is in fact a set of values: almost always these are a continuum flux and slope, and a Gaussian line peak flux and sigma width. For unresolved lines you can assume a Gaussian profile and the instrumental FWHM can be obtained from the data, from the PACS Handbook, or it can be computed in HIPE with the task getSpecResolution.

There are a number of sources from which that the spectral part of the model can be obtained, and this is demonstrated in the difference cases tackled in the script:

1. **Proof of concept**: takes the spectral model from the observation – the spectrum is taken from the point-source corrected result. This spectrum is definitely the correct spectrum of the source (i.e. what it really does look like), and this is used to demonstrate that the FM process works. The fluxes of this spectrum (the continuum flux and Gaussian line peak) are scaled by the area the source subtends on the sky, as the units have to be Jy/arcsec$^2$. However, most users will not know what the spectrum really is, so this is not a realistic user case.
2. **Testing on an offset source**: uses the same spectrum input but a different spatial model (an off-centred R Dor rather than a centred one). This demonstrates the differences in the results when the spatial model of the source (in this case, it’s exact location) is less certain.
3. **Using a standard spectrum**: here we demonstrated a case that simulates what most users will do: if you do not know what the spectrum of the source actually is, then you can use a “standard” spectrum to define the spectral model: a continuum flux of 0, slope 0, and a peak of 1 (or 10, or 13.5, or any value at all – the values taken from the observed spectrum can be used, for example). Again, the fluxes should be area scaled (as explained in case 1).

There are slight differences in how the uESC is then calculated for these cases:

1. **Proof of concept and Testing on an offset source**: the summed source spectrum is extracted from the cube created by pacsSpecFromModel, no uESC is necessary because the proof of concept is that the summed spectrum of the cube created by pacsSpecFromModel is the same as the summed spectrum taken from the observed cube (with the pESC removed).
2. **Using a standard spectrum**: the script uses a spectral model with a continuum of

0 and a Gaussian line with peak of 1. The input is therefore not actually a spectrum you can measure, just numbers: the only way to compute the intensity of this line is to use the equation of the area under a Gaussian. However, the output is a cube: that created by pacsSpecFromModel. The spectrum of the source is therefore extracted from this output cube (the script simply sums the entire cube spatially): the line intensity can then be computed with the Spectrum Toolbox tools (e.g. the Spectrum Fitter or the Cube Toolbox), but the script instead also uses the equation of a Gaussian area to (in this way one is comparing like to like). The ratio of the two is the uESC, as explained in Sec. 4.2.

## 4.6 Line shapes

Please note that one thing the FM script does *not do* is correctly model how the spectral lines may change in shape over a source – either because of the source's velocity structure, or because the source is point-like and has some consequent skew in the profiles due to the relative location of the various source's components with respect to the "slit" they're falling on (the IFU is re-imaged onto the equivalent of a stack of slits: see the PACS Handbook for more detail). This may make comparison of observed to modelled difficult, as a changing spectral shape will affect how the source looks on the sky at different wavelengths.

# 5 The FM Script

Here we show the plots and images that are produced by the FM script and explain what is happening at each stage. As the script is itself well annotated, the description here will be kept short.

The data used in this script is taken from a mapping observation done of the point source R Doradus. In this observation the telescope scanned around the source, and so in each raster position the source is located in a slightly different spaxel of the re-binned cube.

The amount of time and memory that the task pacsSpecFromModel requires increases with the number of re-binned cube slices it is being required to model. If you have a 3x3 or larger line-scan mapping observation, we recommend having at least 10GB assigned to HIPE.

## 5.1 Case 1: proof of concept

This is to demonstrate that the FM works. The model is kept simple by taking the re-binned cube from the observation where R Dor is as close as possible to the centre of the FoV. The spatial model is then a (almost fully) centred narrow Gaussian. A slight offset to the position (an offset between the star and the meta data raNominal and decNominal) is measured, and the script demonstrated how to check if you have measured the offset correctly. The input spectral model is the point-source corrected spectrum of the source, since this is in fact the true spectrum of the source. The model is then created and input into the task pacsSpecFromModel and the spectrum extracted from that. If the model is correct, then the simulated spectrum should look like the observed spectrum: here it agrees to a factor 0.99 ☺. The script also points out that the small-area comparisons (i.e. one spaxel rather than the total source) will usually be less correct since the input model has to be more correct (our corrected single spaxel agrees to a slightly worse factor of 0.95).

As the uESC correction in the blue is very small, we also checked that the FM works in the red using a calibration observation. Unfortunately, there is no useful standard observation of R Dor in the red to use in the script, so it will be difficult for the user to run the same checks in the red. We found that the correction in the red is also is good to a few percent.

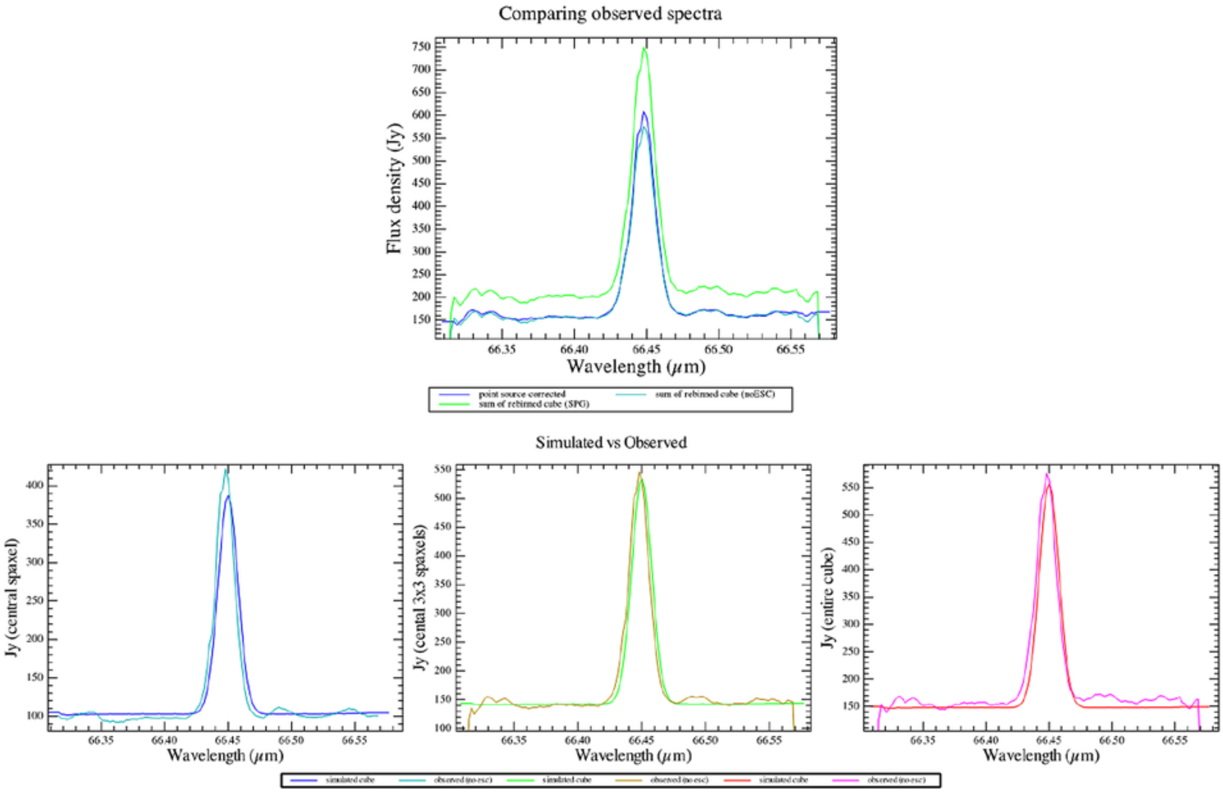

**Figure 4 Top** The spectrum produced in the first demonstration in the script. The spectrum from the re-binned cube taken directly from the observation is compared to the spectrum from that cube with the pESC removed, and both are compared to the point-source corrected spectrum. **Bottom** Spectra produced by the simulation compared to the spectra taken from the observation, for different parts of the cube.

## 5.2 Case 2: offset point source

The proof of concept was done on a centred source, where the correction is small because most of the source falls within the central spaxel, and the effect of the uneven illumination is limited. To show that this FM process works on off-centred sources, located in a worse spot in the illumination pattern, we ran the same test but starting with a cube where R Dor is located in another spot in the FoV. This requires one extra step: to change the meta data raNominal and decNominal so the test will work (this step will never be necessary for a real science case). The script again explains how to measure the offset of the source: by turning the re-binned cube into an interpolated cube, making an image of the cube by summing it along the spectral dimension, and then looking at the position of the star on that image. This is not easy, especially for this offset point source as the source has spectral lines with a skewed profile (this is a common consequence of spatial offsets, as is explained in the user note on point sources); the effect of this is to give the source a bit of an elongation at some wavelengths, as the images below demonstrate.

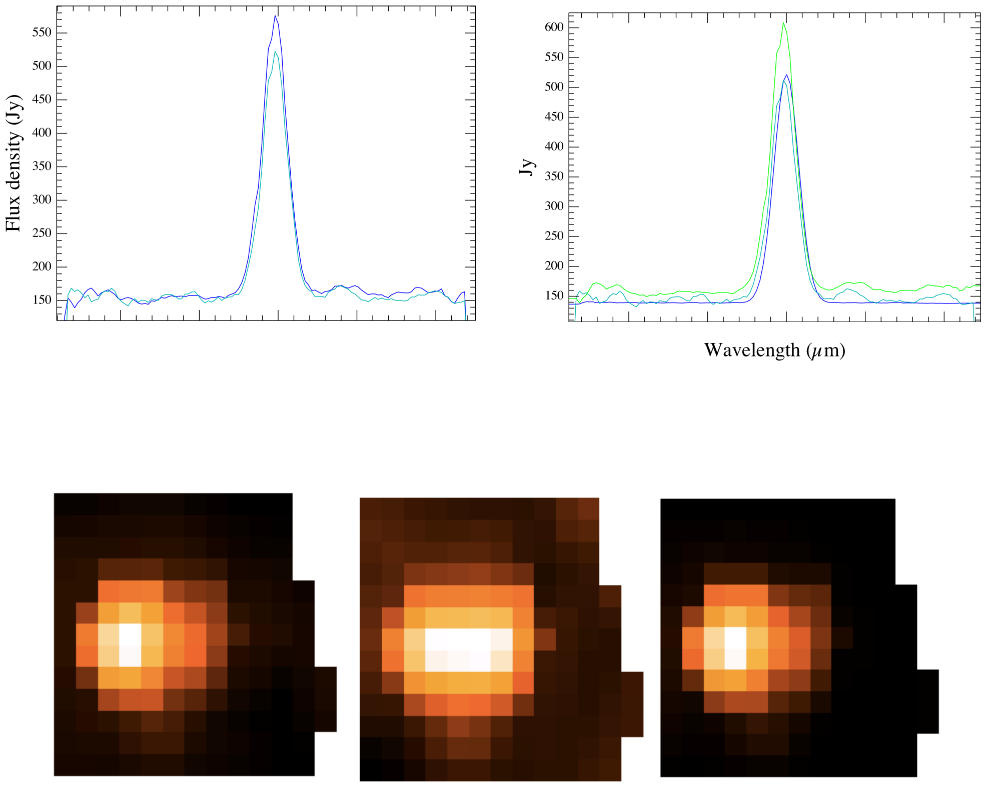

**Figure 5 Top left** is the spectra of the off-centred and centred cubes of R Dor, showing that, as expected, the off-centred spectrum is the fainter – the flux losses for an off-centred point source are greater than those of a centred point source. **Top right** is the simulated spectrum and the observed spectrum, showing that they are similar (i.e. we have simulated well the observed spectrum), and that both are fainter than the green curve, which is output of the forward modelling script: the observed spectrum with spatial response of the PACS IFU corrected for. The flux losses that result from this spatial response pattern are corrected in the pipeline by the point-source calibration curves, but here we instead modelled the point source, and the necessary corrections, with the forward modelling script. **Bottom** are screen grabs of interpolated cubes of **Left**, **Middle** the observed cube at a shorter and a longer wavelength within the spectral line, and **Right** the simulated

cube (which is the same at all wavelengths). If the appearance of the source changes with wavelength, this will add to the uncertainty in the final simulated results.

The final result of this case is a uESC that produces a corrected point-source line flux that is 0.93 of the true value. The correction is not quite as good as for the centred case, which is probably a combination of the greater difficulty in identifying the central location of the source and the fact that the very wings of the source will have fallen out of the FoV of the cube.

## 5.3 Case 3: using a standard input spectrum

If you do not know what the spectrum of the source should be (which will be true for most users of this script), you can use a standard spectrum, e.g. of continuum 0 flux and peak 1 Jy. The script demonstrates what to do here, using the off-centred pointing as demonstration. The spectrum extracted from the re-binned cube (with the pESC removed) is still the “observed” spectrum, to be corrected at the end. But since the spectral fluxes input to the model are not taken from an actual spectrum that you can measure, the integrated fluxes to be compared are computed from the equation of a Gaussian (which is the spectral line profile assumed here). The integrated flux of the input spectral line is computed with the peak value and sigma value (Gaussian width) defined for the input model, and the integrated flux from the simulated cube is measured in the same way. The uESC is the ratio of these two integrated fluxes; our correction is 0.94 of the true value, which is the same result as we obtained in case 2.

## 5.4 Case 4: steps for a mapping observation

The script next takes you through what to do for a mapping observation, using the centred pointing as demonstration. Rather than taking the full mapping observation of R Dor, we again use the pointing where the source is centred only: this is to see how well the result compares to case 1. The main difference from before is in the type of cube from which the observed spectrum is taken, and that you need to create that same type of cube from the simulated re-binned cubes created by pacsSpecFromModel. It is still necessary to remove the pESC from the slicedRebinnedCubes taken from Level 2/2.5, and it is from this you create a mosaic cube. Which type of cube to create depends on the type of observation you have, see the PACS Products Explained to learn more. The script creates both types that are possible here – projected and interpolated – in order to show that there are slight differences in the corrections and results derived: the results are dependent on the source, the model, the pointing pattern, *and* the type of mosaic cube used. The results are very good here (within 1% of the correct value), but we should point out that when repeating this exercise on the off-centred pointing, the results are worse: the corrected result is 0.90 (projected cube) and 0.93 (interpolated cube) of the correct result.

## 5.5 Case 5: spatial distribution of the correction

Finally, the script demonstrates how you can estimate the spatial distribution of the correction, or, in other words, how you can obtain the illumination pattern for your particular observation. This illumination pattern is not absolute – you can only tell where the illumination is lower and higher in your observed field. The script uses a flat flux spectral model as input to do this, and shows you how to produce a mean-scaled image at the end that is your illumination pattern. When inspecting this, bear in mind that your area-integrated uESC was taken from the part of the field where your source is only, while this image shows the entire observed field.

# 6 Uncertainties

The tests of this script were limited to the case of a point source, where the shape and flux of the source is fairly well known, and with the source centred and off-centred from the centre of the FoV. We showed that for these limited scenarios, an underestimate of 10% could be achieved in some cases, with the largest underestimates being when the source is off-centred. The uESC correction requires a good spatial model, and for smaller sources the model has to be more correct than for larger sources, as the uESC is more sensitive to where the source actually lies on the field compared to where your model places it. The correction depends on the mosaic cube used: if you have a mapping observation with sparse spatial sampling (e.g. tiling, or only a 2x2 raster), this effect will be more important. Finally, it is important that the source you model is the source alone, without contamination from any other sources in the field.

# 7 Glossary

FoV: field of view

HSA: Herschel Science Archive

HIPE: Herschel Interactive Processing Environment

Mapping observing mode: Two pointing modes were offered for observing with PACS. Mapping observations were performed as a raster of pointings.

Mosaic cubes (Interpolated, Projected, Drizzled): Mosaic cubes are those created by combining individual cubes taken in a mapping observation: separate and slightly-offset pointings combined to observe either a larger field or to improve the spatial sampling of the source.

Pointed observing mode: Pointed mode observations were on a single FoV, and the primary science products are re-binned or interpolated cube, and point-source tables.

Re-binned cubes: “Re-binned cubes” are the final fully-calibrated native cubes that are produced by the pipeline: one cube per requested wavelength range and, for mapping observations, per raster position. It is from the re-binned cubes that the mosaic cubes are produced – these mosaic cubes being the end product for most mapping observations and one possible end product for pointed observations. (The re-binned cubes have the 5x5 grid of 9.4″-sized spaxels that is the native sky footprint of the PACS IFU, while the spatial grid of the mosaic cubes depends on the mapping field requested by the observer and the size chosen for the mosaicked spatial pixels.)

SPG: Standard Product Generator, short-hand for the automatic pipeline run at the HSC for all Herschel observations. The SPG download/products are the “pipeline” products that a user downloads from the HSA after an archive search.

Spatial Pixel/Spaxel: Both are the name given to the pixels of a cube, each of which contains a full spectrum for a small patch of sky. The difference is that *spaxel* usually refers to the pixel of a native cube (for PACS that is the re-binned cube) with an area that is determined only by the instrument (i.e. 9.4″ for PACS); *spatial pixel* refers to the pixel of a mosaic cube, and it can have any size – the ideal size depends on the mosaicking algorithm and the pointing pattern of the observation (how many overlapping spaxels are there in each spatial pixel