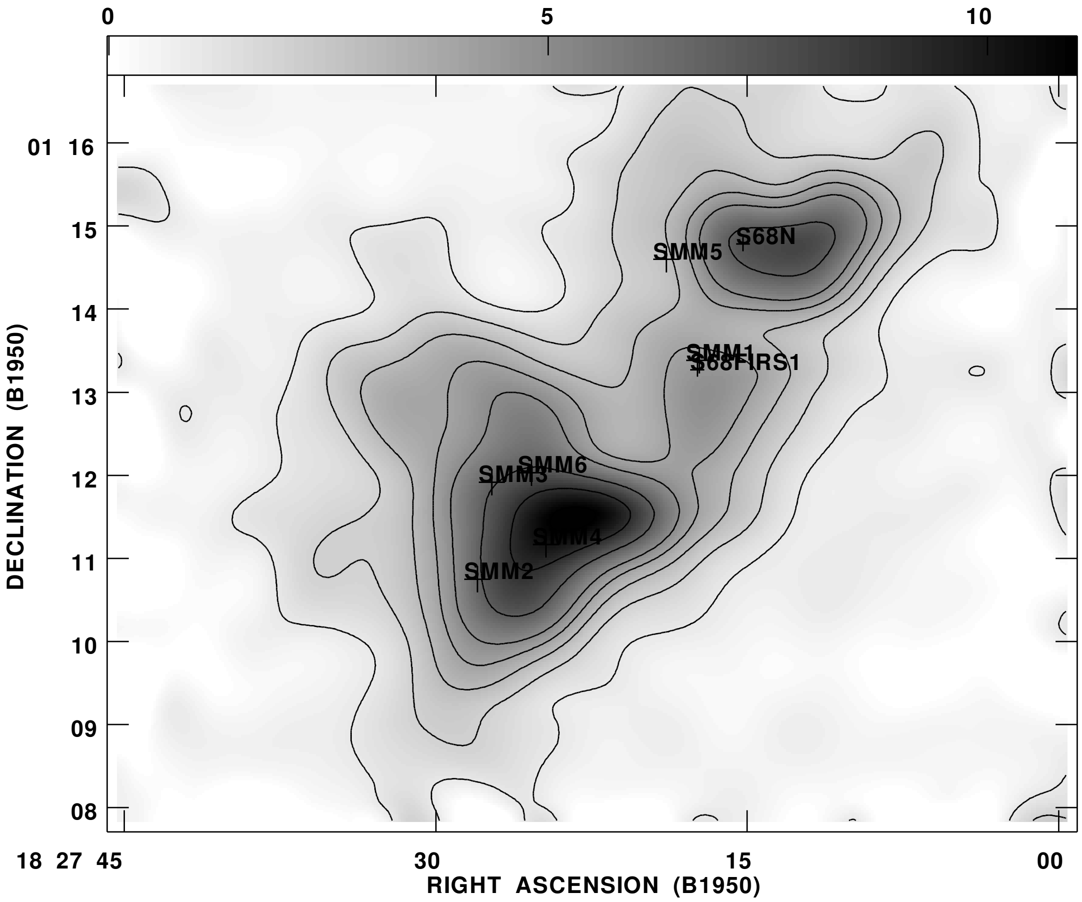

On The Fly Observing at the 12 Meter

Jeff Mangum
(jmangum@nrao.edu)

September 21, 1999

Contents

1 Introduction

On-the-fly (OTF) mapping is an observing technique in which the telescope is driven smoothly and rapidly across a field while data and antenna position information are recorded continuously. This technique is in contrast to traditional mapping of discrete positions on the sky, which is sometimes called “step-and-integrate” or “point-and-shoot” mapping. There are several advantages to OTF mapping. First, telescope overhead is reduced significantly since a specific position on the sky does not have to be acquired within a given tolerance. Second, since the entire field is covered rapidly, properties of the atmosphere and the system, including antenna pointing and calibration, change less in comparison to a conventional point-and-shoot map. Systematic changes may occur from map to map, but such effects average down rapidly and may be correctable by cross correlation techniques. In general, global changes from map to map are more benign and easier to correct than drifts across a single map field.

OTF mapping is available at the 12 Meter for spectral line and continuum measurements. OTF is not a new concept and has been used at a number of radio telescopes in varying forms for many years. Nevertheless, given the rigor of the position encoding that allows precise and accurate gridding of the data, the fast data recording rates that allow rapid scanning without beam smearing, and the analysis tools that are already available or are under development, we believe that the 12 Meter implementation is the most ambitious effort at OTF imaging yet.

1.1 When to OTF Map

In principle, the OTF technique should be a superior way to make almost any map. It is particularly efficient for large fields and relatively strong emission. In these cases, one gains considerable observing time from the low telescope overhead. We strongly recommend OTF mapping in these cases. OTF can, and has, been used to advantage for small fields and weaker lines. For small fields, however, the telescope must turn around at the end of a row more often. Weak lines may require many, many coverages of the field that can lead to a substantial data reduction chore requiring many hours of your time and a fast computer with gigabytes of disk space. This is a judgement the observer must make. Note that aside from the minimum hardware integration period imposed by the backends, all OTF integrations are done in post-processing analysis software rather than by the real-time systems.

Spectral line OTF guidelines are arbitrary, but we would recommend that for any field $> 6'$ in width, you should most definitely use OTF. OTF may be entirely appropriate for smaller fields, but the advantages may not be as marked. If the field is only a few grid points across, you should probably use conventional grid mapping. If the emission is weak, you may elect to use a slow scanning rate to reduce the number of field coverages required. Be advised that the OTF technique is constantly evolving. Observing options are being enhanced, computers are being upgraded, and analysis software is being developed. Accordingly, the criteria for choosing which observing algorithm to use is also evolving. Check with us before planning your observations.

1.2 How OTF Observing Works

Before starting an OTF map, you configure the telescope control system for the width of the field, the scanning rate, the row spacing, and a few other things. For spectral line OTF, the map is taken in a total power observing mode in the sense that you take a calibration spectrum (a vane CAL) and a total power OFF, followed by one or more total power scanning rows, typically made up of hundreds of individual spectra (the ONs). For continuum OTF, the map is acquired using the continuum beam-switched observing mode. In this mode, the subreflector is switched between two positions (the "+" and "-" beam) in azimuth while the telescope scans. Each of the individual spectra or continuum "+" and "-" beam total power measurements is tagged with the *actual* antenna encoder positions. As a result, antenna tracking errors caused by wind gusts, for example, are actually recorded and taken into account in the data analysis stage.

For spectral line OTF, the same CAL and OFF are used to calibrate all of the ONs in the scanning rows until another OFF is taken. Each total power OFF is given its own scan number. All of the spectra or continuum "+" and "-" beam total power measurements in each scanning row are concatenated along with arrays of time and position information and stored on disk under a single scan number with a single header. The header information for each scanning row contains the scan number of the previous OFF (for spectral line OTF) and CAL.

Given the scanning parameters, the positioning system is configured for tracking rates, row offsets, and the duration of a scanning row. The spectral line data-taking backend tells the tracking system to move to position and begin scanning. Using some handshaking bits on the telescope's status and monitor (SAM) bus, the two systems are synchronized at the start of each row. In addition, both the tracking and backend computers have IRIG clock cards that are driven by the observatory rubidium standard clock. The data backend reads out the spectrometers system every 100 milliseconds for spectral line OTF. The continuum data system is read out every 250 milliseconds. For both spectral line and continuum OTF, the data backend tags each data parcel with the UT time stamp. Since the Millimeter Autocorrelator (MAC) is a digital autocorrelation device, before the data can be read out Fourier transforms of the 100 millisecond data samples are required. To make data analysis as fast as possible, the FFT's are performed in real-time by the MAC control computers. Every 10 milliseconds, the tracking computer records its Az/El position with respect to the field center. At the end of the scanning row, the position information from the tracking system is merged with the data. An interpolation of the position information is done to align slight differences between the time stamps of the two data sets.

1.3 An Important Note About Spatial Sampling

The following is due to Darrel Emerson

When setting up your map observations, it is important to keep in mind the following facts about sampling and aliasing in radio astronomical mapping data. If you want to represent the full resolution of the telescope, you have to sample the data often enough to represent all the spatial frequencies detected by the dish. You can think of the extreme edges of the dish of diameter D as part of an interferometer of spacing D, which has to be sampled at $\frac{\lambda}{2D}$. Depending on the illumination taper, this corresponds typically to sampling at about 2.4 or 2.5 points per FWHM. Of course, if there is zero, or by some definition "negligible", illumination at the edge of the dish, you won't be sensitive to such high spatial frequencies. It will be just the same as a smaller dish of diameter d equal to the diameter of that part of the dish that has significant illumination, and the sampling will be calculated from $\frac{\lambda}{2X}$, where X is defined as the diameter of the illuminated part of the surface.

It's useful to consider what happens if you undersample data. Assume that the undersampling happens on the sky, rather than in later in the data processing. Suppose you have a 10m dish, but you only sample at $\frac{\lambda}{2\times 8m}$ rather than the $\frac{\lambda}{2\times 10m}$ that you should. That means that the spatial frequencies present from the dish baselines of 8m to 10m get reflected back into the spatial frequencies of 8m down to 6m. Not only have spatial frequencies from the 8m to 10m baselines been lost, but valid spatial frequencies from baselines of 6m to 8m have been corrupted. You can't tell if structure in your map with a spatial wavelength of $\frac{\lambda}{7m}$ is genuine, or was really structure at $\frac{\lambda}{9m}$ which has been aliased on top of any genuine $\frac{\lambda}{7m}$ spatial wavelength signal. In this sense, undersampling the sky is really twice as bad as you might have thought.

How important this undersampling is depends on exactly what the illumination taper is, how important it is that you retain the maximum possible resolution of the telescope, how good a dynamic range you want in the observations, and at some level how much fine scale structure there is in the source itself. If you only sample at 0.8 Nyquist (*e.g.* $\frac{FWHM}{2}$ rather than $\frac{FWHM}{2.5}$), what matters is the energy in the data at spatial wavelengths shorter than $\frac{\lambda}{2\times 0.8\times D}$. So in a sense you need to ask what the illumination taper is at a radius of $0.4 \times D$ on the dish surface. The spatial frequency response of a single dish is the autocorrelation function of the voltage illumination pattern. So, you need to calculate how much area there is under the 2-D autocorrelation function beyond spatial frequencies of $0.8 \times D$, compared to the area within $0.8 \times D$. This ratio is some measure of the dynamic range. A better definition of dynamic range might take into account the spatial frequency structure of the source. If the source has no structure on scales smaller than $\frac{\lambda}{2\times 0.8\times D}$, then you don't need to sample at the full $\frac{\lambda}{2\times D}$ anyway.

This is one circumstance where it is perfectly rigorous to undersample the data. If, say, you have a 10m dish, and you are taking data to compare with other observations using a 1m dish at the same wavelength (or the equivalent number of wavelengths at some other frequency) then you only need to sample the data at $\frac{\lambda}{2\times5.5m}$ or $\frac{\lambda}{11m}$. This is so because, when sampling a 10m dish as if it were a 5.5m dish, the spatial frequency components from baselines of 5.5m out to 10m will be reflected back into the data as if corresponding to baselines of 5.5m down to 1m. So, the spatial frequency terms of 1m baseline and below will not have been corrupted. The data analysis of this undersampled data would apply a spatial frequency cutoff at 1m, and there will have been no corruption in this smoothed data caused by the undersampling. Putting it in more general terms, if you are going to be smoothing a dish of diameter D to simulate observations made with a smaller dish of diameter d, then the sampling interval only needs to be $\frac{\lambda}{d+D}$ rather than $\frac{\lambda}{2\times D}$.

There are other aspects that make it desirable to sample **more** often than the Nyquist rate, as we recommend for OTF observing. These are practical points like how well gridding or interpolation works with a finite sized gridding or interpolation function. A little oversampling may enable you to reduce the convolution (interpolation) function by a factor of a few, saving a huge amount of computational overhead at the expense of a few per cent more data.

1.4 Other Sources of Information

- D. T. Emerson, "Increasing the Yield of our Telescopes", in Proceedings to IAU 170, *CO: 25 Years of Millimeter-Wavelength Spectroscopy.*
- E. Greisen, "Single-Dish Data in Aips", Chapter 10, *AIPS Cookbook.*
- J. G. Mangum, D. T. Emerson, & E. Greisen, "The On-The-Fly Observing Technique", in Proceedings to *Imaging at Radio through Submillimeter Wavelengths.*

2 Observing Setup

2.1 OTF Observing Checklist

The following list gives some step-by-step instructions which you can use as a guide to acquiring and processing OTF data. In each item, I indicate the appropriate section which you should refer to for further information.

- For Spectral Line OTF...
  - Have the operator configure the filter bank, MAC, and position-switch parameters for your measurements.
  - Test the system by having the operator make a short position-switched measurement of a strong source in the field you are going to map.

- Use your test measurement to identify and flag any bad filter bank channels (§2.2.4).

For Both Spectral Line and Continuum OTF...

- Tell the telescope operator that you will be doing OTF mapping. You will need to specify the map setup parameters (§2.3).
- Start a new data file for each new map coverage.
- Once the map has begun, adjust the Dataserv parameters to get the best on-line display (§3.1).
- Use the AIPS procedures OTFSET and OTFRUN to read and process your completed map (§6.1.1).
- Run the Unix shell script *datacomp* to compress your raw data files (§2.4).
- Check the raw and AIPS disk status with the disk space monitor (select "Diskspace" in the obs "Observer Tools" menu). If you reach the point where you need to write data to tape, use the *tar* utility (§2.4).

2.2 OTF Map Parameters

In the following I describe some of the OTF map parameters which you must calculate in order to setup your OTF observing.

2.2.1 Row Sampling Rate

It is very important that you set up your map to be properly sampled perpendicular to the scanning direction. If you under sample, you will miss information in the image field, you will be unable to accurately combine your image with that from interferometers or other single-dish telescopes, and you may introduce artifacts from the analysis algorithms. Keep in mind that you can always smooth the map after it is taken to degrade the resolution and improve signal-to-noise.

The scanning rows must be spaced at no more than the Nyquist spacing, which is also known as "critical sampling" and is given by

$$\begin{aligned} \theta_N &\equiv \frac{\lambda_{obs}}{2D} \\ &\simeq \frac{2576.53}{\nu_{obs}(GHz)} \end{aligned}$$

for a 12m aperture. In practice, though, a small amount of oversampling is recommended. If the rows are critically spaced, small scanning errors can result in the map being under sampled. In addition, the tail of the gridding function used by the analysis procedure extends slightly beyond the information cutoff of the telescope, which will result in some noise being aliased into each grid cell. A little oversampling will reduce this effect. We recommend that the rows be spaced by

$$\begin{aligned} \theta_{row} &= \left(\frac{\lambda_{obs}}{2D}\right) \times 0.9 - 2'' \\ &\simeq \left(\frac{2318.87}{\nu_{obs}(GHz)}\right) - 2\ arcseconds \end{aligned} \tag{1}$$

where the factor of 0.9 is an oversampling factor and the factor of 2″ is a guard band to accommodate any scanning errors. Figure 1 is a plot of this relationship. This parameter will be set by the on-line system using the above equation and the current observing frequency.

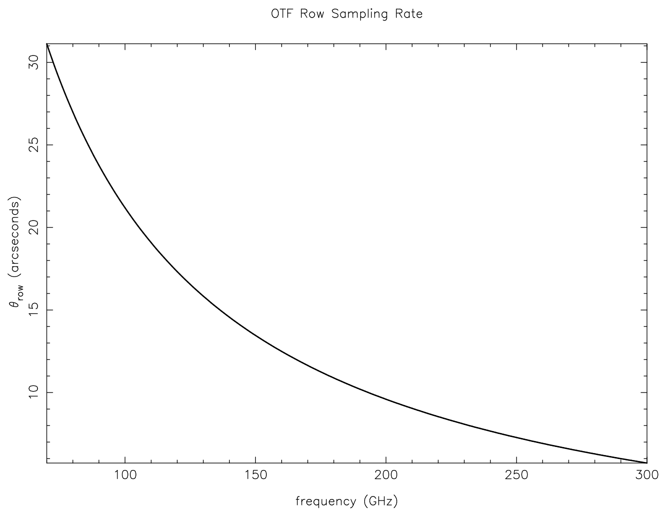

Figure 1: θ_{row} as a function of observing frequency.

2.2.2 Scanning Rate

Your map must also be sampled properly in the scanning direction. Since the spectrometers must integrate for a finite interval before being dumped, the data are "square-box" binned. To avoid noise aliasing and beam-smearing problems, one needs to oversample in the scanning direction. The scan rate may be written as

$$R(arcsec/sec) = \frac{\theta_N}{n_{os} t_{dump}} \tag{2}$$

where θ_N is the Nyquist spacing defined above, n_{os} is an oversampling factor, and t_{dump} is the spectrometer data dumping interval. At the 12 Meter, $t_{dump} = 0.1$ seconds for spectral line OTF, 0.25 seconds for continuum OTF. We recommend a minimum value for n_{os} of 2, which will result in $< 3\%$ increased noise as a result of aliasing and minimal beam broadening. Note that it is quite acceptable to oversample by much larger factors than the minimum value suggested above, particularly in the scanning direction. There are some tradeoffs to be considered, however. Large factors of oversampling will produce better signal-to-noise on a single coverage of the field and may result in simpler data processing, *i.e.* fewer maps to average together. On the other hand, the longer a single coverage takes, the more susceptible your image will be to drifts in the system, such as pointing and atmosphere. This eliminates some of the advantages of fast mapping mentioned above. As a compromise, we recommend that the row spacing be computed from Equation 1, and that the scan rate follow from Equation 2, with $n_{os} > 2$. Other circumstances may come into play, of course. For example, if the field size is very small, you may choose to slow the scan rate (increase n_{os}) substantially.

2.2.3 Row Length Limitations

Since the telescope acquires a data sample every 0.1 (spectral line) or 0.25 (continuum) seconds, the combination of the telescope slew rate and the scan length in RA determines how much data you get per row. Due to software and practical restrictions, the maximum number of samples one can acquire in a given scan is 3600. This corresponds to a slew rate of 10 arcseconds/second over a scan which is 1° in length. You should set your observing parameters to stay below the 3600 spectra per row limit.

2.2.4 Bad Channel Flagging

It is a good idea to flag bad filter bank channels *before* processing the data in AIPS. Therefore, before you begin your OTF map, identify any bad filter bank channels by making a normal position-switched test measurement of your source. Use the *badch* procedure in unipops to identify the bad channels and tell the operator that you want these channels to be flagged. If you forget to flag some bad channels, you will have to use the chansel adverb in *OTFUV* (see §6.1.2).

2.2.5 Map RMS

When planning an OTF observing run you must calculate how much integration time you need for each sampling cell to reach the desired signal-to-noise level. First determine the integration time required using the standard Radiometer Equation

$$^{spec}\sigma_{cell} = \frac{T^*_{sys}}{\sqrt{\Delta\nu\, ^{spec}t_{cell}}}\left[1 + \frac{^{spec}t_{cell}}{t_{off}}\right]^{1/2} \quad \text{(for spectral line)}$$

$$^{cont}\sigma_{cell} \quad = \quad something \quad \text{(for continuum)} \tag{3}$$

where T^*_{sys} is the effective system temperature, $\Delta\nu$ is the spectral resolution, t_{cell} is the time spent integrating on a cell, and t_{off} is the off-source integration time. For spectral line OTF, the time spent integrating on a given cell is typically one second or less, while the off-source integration time is typically 10 seconds (the default value). Therefore, $t_{cell} \ll t_{off}$ and we can rewrite the spectral line Radiometer Equation as

$$^{spec}t_{cell} = \frac{1}{\Delta\nu}\left(\frac{T^*_{sys}}{\sigma_{cell}}\right)^2 \tag{4}$$

The integration time in a map cell is a function of the scanning rate and the number of coverages of the map field, and should be calculated with respect to the time accumulated in a Nyquist sampling cell. Figure 2 shows some of the terms I will define below.

Referring to Figure 2, if we define θ_x to be the angular distance along the scanning row, θ_{ramp} to be a "ramp-up" distance at the beginning and end of each row, and R to be the scanning rate (see §2.2.2), then the time to scan one row is

$$t_{row} = \frac{\theta_x + 2\theta_{ramp}}{R} \tag{5}$$

If we also define θ_y to be the angular height of the map and θ_{row} to be the angular sampling interval between scanning rows, then the number of rows in the map is given by

$$N_{row} = \frac{\theta_y}{\theta_{row}} \tag{6}$$

The number of independent sampling cells in the map is

$$N_{cell} = \frac{(\theta_x + 2\theta_{ramp}) \times \theta_y}{\theta_N^2} \tag{7}$$

where we are approximating the cells as rectangular. The integration time per cell, t_{cell}, is related to the above three quantities as follows

$$t_{cell} = \frac{t_{row}N_{row}}{N_{cell}} \times \eta \tag{8}$$

where η is a convolution function correction factor which takes account of the fact that we are using a tapered function (a Gaussian-tapered Bessel function) instead of a pure sinc. This correction factor is given by

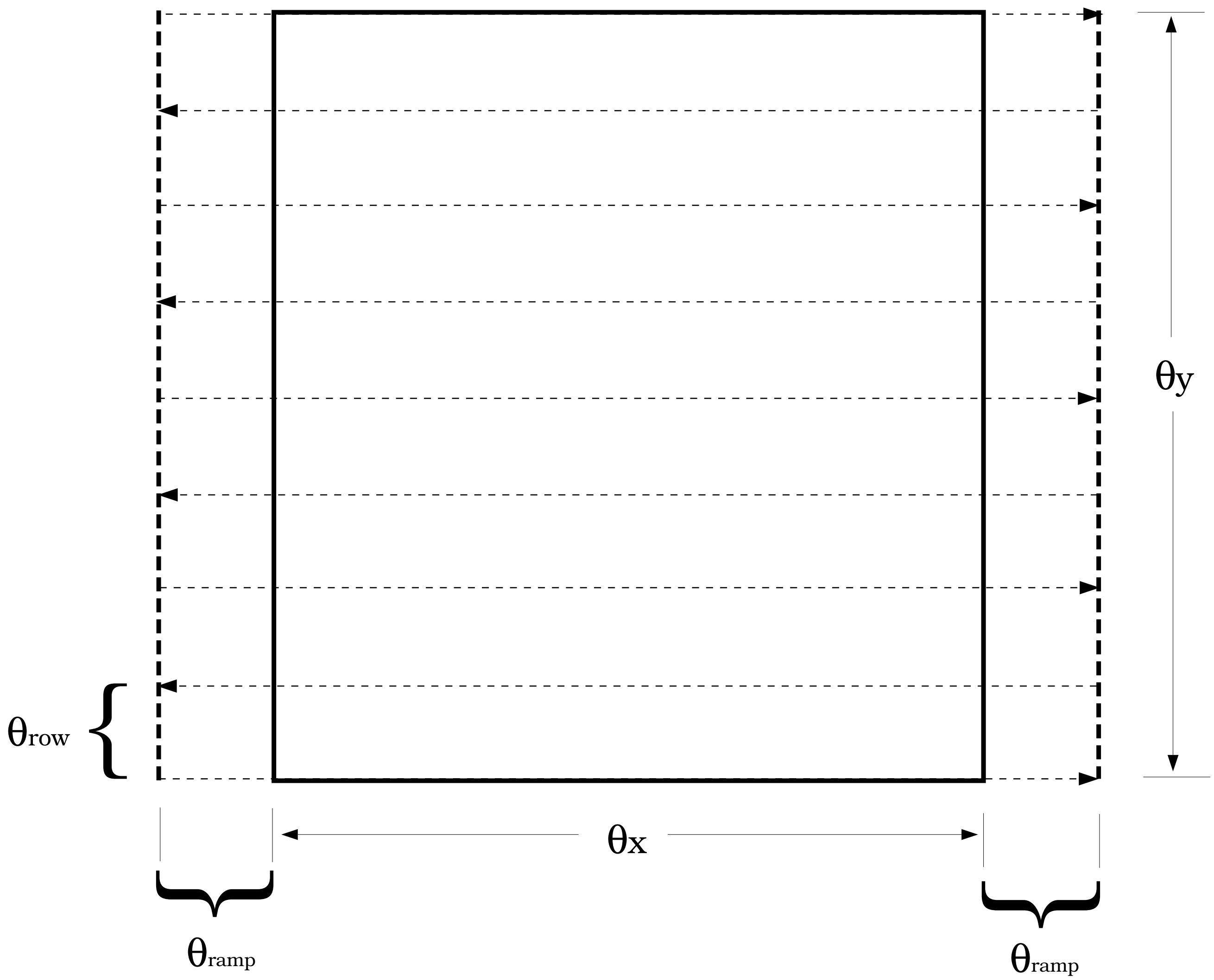

Figure 2: Map parameter definitions.

$$\begin{aligned} \eta &= \frac{\int FFT\,(Gauss \times Bessel)\,dxdy}{\int FFT\,(sinc)\,dxdy} \\ &\simeq 2.9135 \end{aligned} \tag{9}$$

for the truncated functions we use in the AIPS task SDGRD (see §6.1.2). Combining Equations 5, 6, 7, 8, and 9, we find that

$$t_{cell} = \frac{2.9135\theta_N^2}{R\theta_{row}} \tag{10}$$

Combining this relation for t_{cell} with our Radiometer Equation (Equation 4), we find that the RMS noise per cell in an OTF map is given by

$$\sigma_{cell} = \frac{T^*_{sys}}{\theta_N}\left(\frac{R\theta_{row}}{2.9135\Delta\nu}\right)^{\frac{1}{2}} \tag{11}$$

Note that a given RMS noise per cell obtainable with multiple coverages and/or polarizations is given by

$$\sigma = \frac{\sigma_{cell}}{\sqrt{N_c}} \tag{12}$$

where N_c is the number of coverages and/or polarizations you combine to produce your final image. Note that you can also improve your map RMS by spatially smoothing it after gridding (see §6.1.4).

2.2.6 Total Map Time

The total time required to acquire a map must include not only the integration time scanning across the field but also the time for calibration and OFF integrations. This can be written as

$$\begin{aligned} {}^{spec}t_{tot} &= N_{row}\left[t_{row} + \frac{t_{off}}{N_{rpo}} + \frac{t_{cal}}{N_{rpo}N_{opc}} + \frac{\epsilon}{N_{rpo}}\right] \quad \text{(for spectral line)} \\ {}^{cont}t_{tot} &= N_{row}t_{row} \quad \text{(for continuum)} \end{aligned} \tag{13}$$

where N_{row} is the number of rows in the map, t_{row} is given by Equation 5 above, t_{off} is the OFF integration time, t_{cal} is the calibration integration time (VANE plus SKY sample times), N_{rpo} is the number of rows per OFF measurement you do, N_{opc} is the number of OFF measurements per calibration measurement you make, and ϵ is the "overhead" time, which is the time that the telescope spends doing things other than integrating (like moving from OFF positions to the map field). The value for ϵ depends mainly on how far away the OFF position is from the map, but 10 seconds is probably a good average estimate. The CAL is taken at the same position and just before the OFF, and so doesn't involve additional overhead. There may be other small overhead losses in moving from row to row and starting scans, but these losses can usually be neglected.

2.2.7 Doppler Tracking

The default reference position for doppler tracking of OTF maps is the map center position. It is possible to change these settings to doppler track relative to any RA and Dec.

2.2.8 Examples

Spectral Line:

In early September of 1995 I made a map of the J=1 $\rightarrow$ 0 $C^{18}O$ transition toward the Serpens star formation region. The rest frequency for this transition is 109.782160 GHz, which means that the Nyquist sampling rate was 23″ (Equation 1) and the row separation was 20″ (Equation 1). I mapped a field $30' \times 30'$ in size, which meant that my map had 90 rows. I scanned at a rate of 50 arcsec/sec, which gave me an oversampling factor of 4.6 (Equation 2). I used the default ramp-up distance of 1′, which means that each row took 38.4 seconds to acquire (Equation 5). Given these map parameters, the number of independent sampling cells in the map was 6,533 (Equation 7). The system temperature was about 300 K while the filter resolution was 100 kHz. Therefore, from Equation 11 we calculate that $\sigma_{cell} \simeq 0.76K$ for a single-polarization map. I actually did this map twice, both times with dual polarization, so that the resultant rms in my map will be about 0.38 K (Equation 12). This estimate compares quite favorably to the measured rms of $\sim$ 0.37 K.

Continuum:

Coming soon...

2.3 OTF Map Setup

When you give your map setup parameters to the operator, he will fill-in a menu like the one shown in Figures 3 and 4. In the following, I describe each of these parameters and indicate how they should be set for both types of OTF map. I have also written a program called *MAPCALC* which will calculate your OTF map RMS given your input map parameters. This program is available on any of the 12 Meter Telescope computers and is also available by request from Jeff Mangum.

OFF Integration Time: *(spectral line only)* This is the time spent acquiring OFF position measurements. The default value of 10 seconds is good for almost all situations.

CAL Integration Time: This is the time spent acquiring a vane calibration measurement. The default value of 5 seconds is good for almost all situations.

OFF Type: *(spectral line only)* You can choose between relative (PS) or absolute (APS) OFF position measurements. If you use APS, you must have an OFF source position accessible from your source catalog.

Rows per OFF: *(spectral line only)* The default value is 1, but under almost all circumstances it will be more efficient to use 2. If you use 2, you should make sure that you use the OFF-interpolation feature of the *OTFUV* task in AIPS to assign the closest OFF measurement in time to your ON scans. Larger numbers of rows per off should

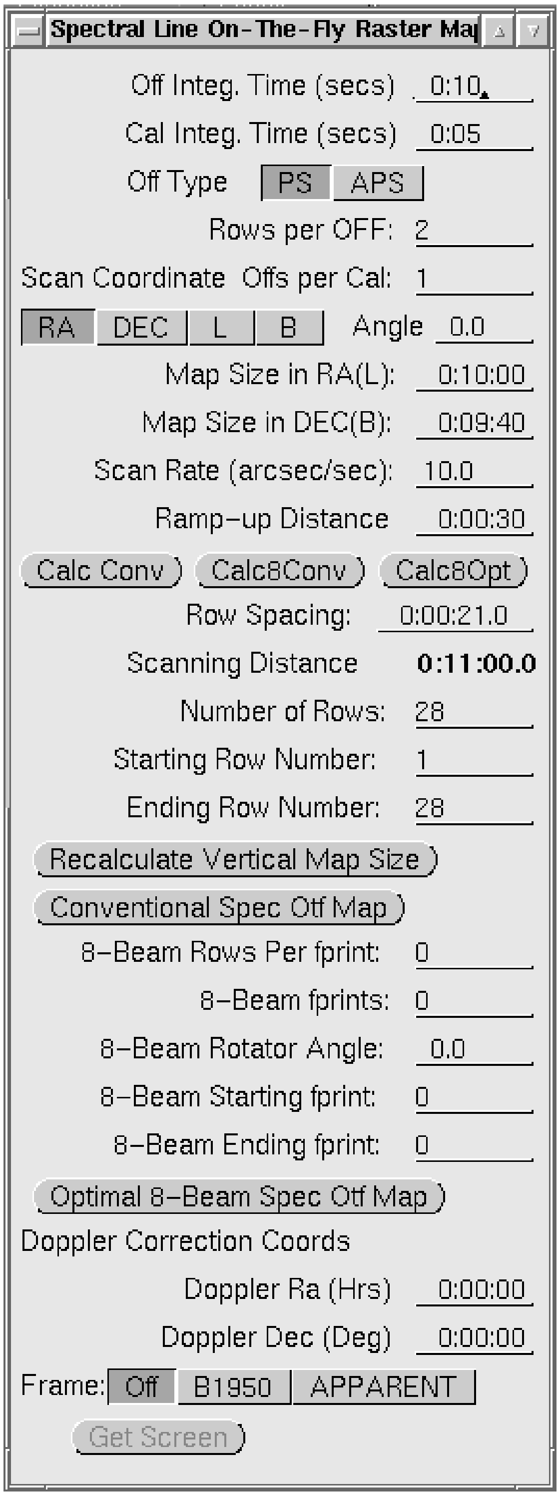

Figure 3: Spectral line OTF observing setup screen.

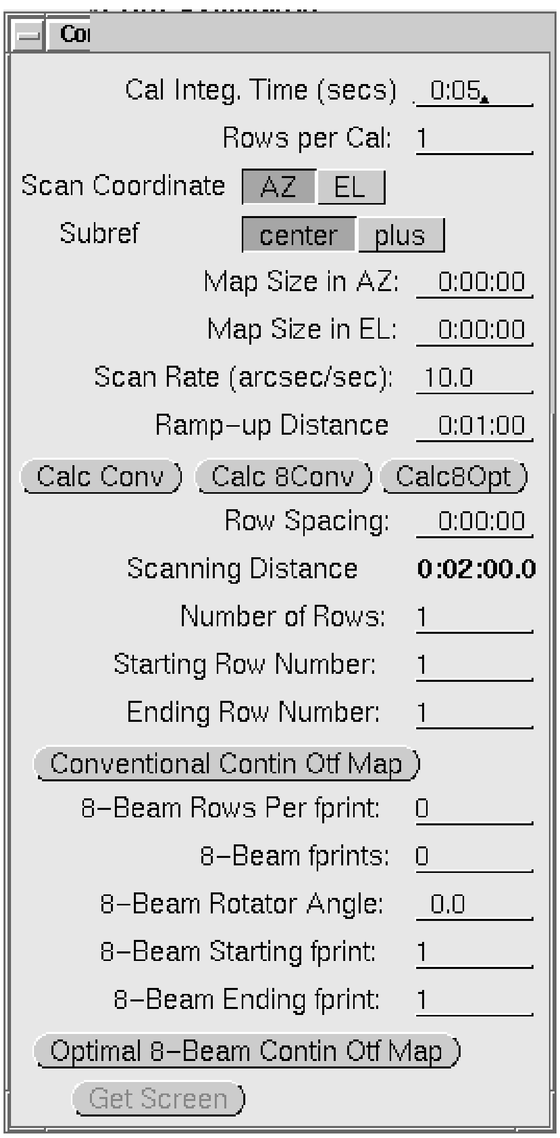

Figure 4: Continuum OTF observing setup screen.

only be done under the very best of weather conditions at 3mm, and rarely if ever at 1mm.

Rows per Cal: *(continuum only)* The default value is 1, which is probably the best value to use under all conditions.

OFF's per CAL: *(spectral line only)* This should be left at the default value of 1 since calibration measurements involve a minimal amount of overhead.

Scan Coordinate: This is the coordinate in which the map rows will be acquired. For spectral line OTF you can scan in either RA, Dec, lII, or bII, while in continuum OTF the scan directions are Az and El.

Angle: *(spectral line only)* This is the angle relative to the scanning coordinate entered above that you want to have the telescope scan.

Map Size in RA(L): *(spectral line only)* This is your map dimension in RA or lII *excluding* the rampup distance.

Map Size in DEC(B): *(spectral line only)* This is your map dimension in DEC or bII *excluding* the rampup distance.

Map Size in AZ: *(continuum only)* This is your map dimension in Az *excluding* the rampup distance.

Map Size in EL: *(continuum only)* This is your map dimension in El *excluding* the rampup distance.

Scan Rate (arcsec/sec): Since the resultant map RMS noise level at a given observing frequency and spectral resolution is given by

$$\sigma \propto \sqrt{\frac{R}{N_c}} \tag{14}$$

(see Equations 11 and 12), your target map RMS noise level can be achieved by slow scanning rates and fewer coverages or faster scanning rates and a larger number of coverages. When setting R, you should:

1. Measure at least 2 data samples per Nyquist interval (see §2.2.2). Note that at 230 GHz the maximum scanning rate is 56″/sec for spectral line OTF, and **??**″/sec for continuum OTF.
2. Calibrate often enough so that the sky doesn't vary significantly between your ON and associated OFF source measurements (for spectral line OTF) or during a row (for continuum OTF). This time should be less than 2 minutes at 3mm and less than 1 minute at 1mm, with shorter intervals necessary under poorer weather conditions.

3. *(spectral line only)* Spend the majority of the total map time ON source, which would be the case if t_{row} (Equation 5) is much larger than t_{off} plus t_{cal}.
4. Be able to complete your map in less than 1 hour to minimize changes in pointing and calibration.

Smaller map fields are generally done at slower scanning rates ($10''$/sec is typical), while larger fields are mapped using faster scanning rates ($60''$/sec is common, but too fast for observing frequencies $\geq$ 230 GHz).

Ramp-up Distance: This is the distance at the end of each row that the telescope uses to get up to speed and on track. The default value of $1'$ is a good value for most map fields, but if your row length is smaller than about $10'$, you should use a smaller value (*i.e.* $30''$).

Calc Conv, Calc8 Conv, and Calc8 Opt: These three buttons calculate the default row spacing and number of rows for conventional OTF (*Calc Conv*) and conventional 1mm array (8-Beam) OTF (*Calc8 Conv*) observing. For optimal 1mm array OTF, the *Calc8 Opt* button calculates the number of rows per footprint, number of footprints, and the rotator angle.

Row Spacing: See §2.2.1. This value is set by the control system.

Scanning Distance: This is an informational field which tells you what your total scanning distance, which is the row length plus ramp-up distances, along a row is.

Number of Rows: This is set by the control program based on your vertical field size and row spacing. Unless you are going to measure only a portion of your map field, this parameter should be left at the value set by the control system.

Starting Row Number: The first row to measure in your input map field. As with the number of rows, this is set by the control program and is left at the set value unless you are measuring a sub-map of your input field.

Ending Row Number: The last row to measure in your input map field. As with the number of rows, this is set by the control program and is left at the set value unless you are measuring a sub-map of your input field.

Recalculate Vertical Map Size: If you decide that you want to choose a finer row spacing and/or a different number of rows, this button will recalculate the vertical map size (map size in Dec or bII).

Conventional Spec OTF Map: Starts your map observation.

The following set of parameters in this setup screen are for 1mm array (or 8-Beam) observations (see "Observing with the NRAO 1mm SIS Array Receiver" for details).

8-Beam Rows Per fprint: The number of 8-Beam rows which need to be acquired in order to fill-in one 1mm array footprint. This parameter is calculated by the control program.

8-Beam fprints: The number of 8-Beam footprints which need to be acquired in order to cover the designated field. This parameter is calculated by the control program.

8-Beam Rotator Angle: The current total offset angle (which includes any user offset angle specified) of the 1mm array. This parameter is set by the control program.

8-Beam Starting fprint: The first 1mm array footprint to measure in your input map field. This parameter is set by the control program and is left at the default value unless you are measuring a sub-map of your input field.

8-Beam Ending fprint: The last 1mm array footprint to measure in your input map field. This parameter is set by the control program and is left at the default value unless you are measuring a sub-map of your input field.

Optimal 8-Beam Spec OTF Map: Starts an optimal 1mm array OTF map.

The following set of parameters in this setup screen are for changing the way that the doppler tracking is done for spectral line OTF (see §2.2.7 for details).

Doppler RA (Hrs): This is the RA in in hh:mm:ss toward which the doppler tracking calculations will be made.

Doppler Dec (Deg): This is the Dec in in dd:mm:ss toward which the doppler tracking calculations will be made.

Frame: The coordinate reference frame for the RA and Dec toward which the doppler tracking calculations will be made.

2.4 Data File Management

Spectral line OTF generates a great deal of data. When in OTF mode, the data rate for the filter banks is 3.1 MBytes/minute, while for the MAC the rate is 11.7 MBytes/minute when observing in 2048 channel mode, and 159.1 MBytes/minute when observing in 32768 channel mode. Therefore, including a bit for header information, the total data rate is between 14.8 and 162.2 MBytes/minute, or between 0.9 and 9.7 GBytes/hour. We currently have 330 GBytes of raw data storage space spread across 10 partitions and an additional 90 GBytes of AIPS storage space spread across 10 partitions.

Your OTF data are recorded in files named `sdd.ini_nnn`, `sdd_hc.ini_nnn`, `gsdd.ini_nnn`, and `gsdd_hc.ini_nnn` where `sdd` stands for "single dish data" and are the standard data files containing all filter bank and continuum data, `gsdd` files contain the calibration "gains" data, `hc` indicates data from the hybrid correlator, `ini` are your 3 initials entered by the telescope operator to designate your data directories and files, and `nnn` is a 3-digit number that is incremented for each data file. Your first data file is numbered 001.

Given the high data rates of OTF observing, the raw data disks will fill up quite quickly. In order to manage this large volume of data, we recommend that you open a new set of data files for each map. When an OTF map is finished, you should read the data into AIPS (see §6). Once the raw data has been converted to "UV"[1] data, you can compress the files using the datacomp script from the Unix prompt in the obs directory. Simply type

```
datacomp nnn
```

and the data for version nnn will be compressed. The compression results in about a 50% reduction in the size of OTF filter bank data files, but leads to no compression of the OTF hybrid correlator files. If your OTF observing run is particularly long, you will also need to write your data to tape. Use the Unix *tar* command to write your data to an 8 mm Exabyte or 4 mm DAT tape as follows:

```
tar cvf /dev/rst?  /home/data/ini
```

where the device name rst? can take on the values `rst0` (a 4 mm DAT drive) or `rst1` (an 8 mm Exabyte). `ini` are your 3 observing initials set up by the operator.

3 On-Line Data Monitoring

3.1 Dataserve

The full reduction of a map using AIPS tasks can be done in about a 10 minutes by an experienced user. More elaborate massaging of the data can take much longer, of course. Because it is difficult for one observer to keep up with both the data analysis and keep an eye on the progress of a current OTF map, we have developed a tool which will allow you to take a "quick look" at your OTF data.

Dataserve is a general utility program for automatically displaying all types of 12 Meter data on an X Window system. During conventional switched power spectral line observing, Dataserve automatically displays the filter bank and MAC spectra in a series of panels on the workstation console in the southeast corner of the control room (Dataserve can display on any X workstation, including remote stations on the Internet). Dataserve also displays continuum observations and displays and analyzes five-point pointing checks, focus checks,

[1] These data are called "UV" data because, in form and in data processing procedures, they are analogous to UV data from an aperture synthesis instrument. One difference between single-dish and aperture synthesis UV data is that single-dish UV data are actually in the image plane.

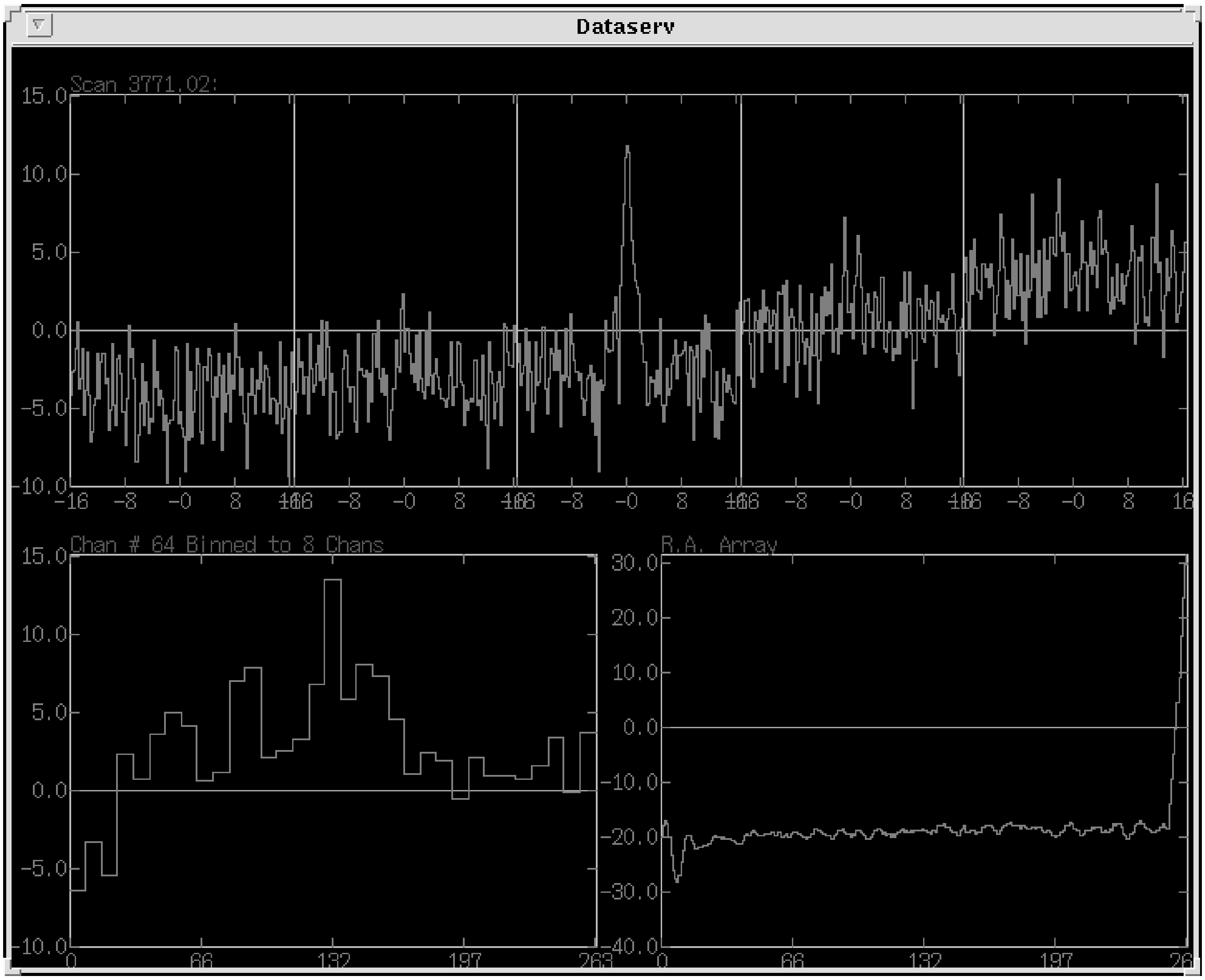

Figure 5: Spectral line OTF Dataserve display.

When spectral line OTF is in progress Dataserve displays a series of panels as indicated in Figure 5. The top set of five spectra are composed from five positions along the scanning row – the beginning, one-quarter of the distance across the field, halfway across, three-quarters, and finally at the end of the row. Each spectrum is actually an average of ten 100 ms spectra. Use these spectra to monitor the quality of spectral baselines and to look for bad channels or other flaws or anomalies. The bottom-left panel of the display is the value of a user-selectable channel (in this case, channel 64) of the spectrometer as scanned across the row. For stronger sources, this allows you to see a cross-section of source emission. You can also use this to monitor total power changes across the field. The bottom-right trace shows the difference between a linear fit to the right ascension coordinate as the telescope scans across the row and a straight line based on the starting point of the row. This trace should be a smooth flat line from start to finish with the exception of a slight sinusoidal wiggle at the start of the row owing to servo oscillations during telescope acceleration. This display allows you to ensure that the scanning is proceeding smoothly.

You can change the display parameters of Dataserve with the “Observer Tools” pull-down menu within your obs session.

3.2 Databrowse

Databrowse will let you get a quick look at OTF data. You can start Databrowse from the “Observer Tools” pull-down menu of your obs session. Databrowse has a variety of options for displaying your data. Information and help on its use can be found within its internal help facility.

4 UniPops

UniPops is the default data reduction program for single-point observations and traditional, point-and-shoot maps made at the 12 Meter. UniPops was never designed to cope with the large volume of data generated by spectral line OTF or the demands of large-scale image analysis. Nevertheless, UniPops does have some useful quick-inspection capabilities for spectral line and continuum OTF. Full image analysis must be done with AIPS.

4.1 Spectral Line

UniPops can be used to display individual spectra in the scanning row and to plot the time and position arrays. Each row of the map, usually scanned in right ascension, is designated by an integer scan number with a fraction appended for the particular filter bank block. If you are using the parallel filter bank mode, four data scans will be recorded for each mapping row; series mode gives two data scans. For example, for scan 30, in parallel mode, the data scans recorded would be

30.01 → Filter Bank 1, Polarization 1

30.02 → Filter Bank 1, Polarization 2

30.03 → Filter Bank 2, Polarization 1

30.04 → Filter Bank 2, Polarization 2

In series mode, the data scans would be

30.01 → Filter Bank 1, Polarization 1

```
30.03 → Filter Bank 2, Polarization 2
```

For the MAC, you will get between 1 and 8 data scans, dependent upon whether you observe with 1, 2, 4, or 8 IF's. For example, if you configure the MAC for 2 IF's, your hybrid correlator data scans would be

```
30.11 → MAC Polarization 1

30.12 → MAC Polarization 2
```

OTF data are recorded in the regular sdd data file but in a special format: all the spectra in a scanning row are written in succession followed by the time and position arrays. A typical scanning row may contain of order 1000 complete spectra stored under a single scan number. You cannot get access to an OTF spectrum using the standard UniPops GET verb. Instead, a special verb, GETOTF has been created for this job. The syntax for this verb is

```
getotf(scan#,spectrum#)
```

where scan# is the usual filter bank or hybrid correlator scan number as explained above, and spectrum# is the number of the spectrum along the scanning row. For example, if a row takes 60 seconds of scanning time, there will be 600 spectra in the row. If you wanted to display the middle spectrum (number 300) of scan 40, filter bank 1, polarization 2, you would type

```
getotf(40.02,300); page show
```

You can also display the time and position arrays as follows:

Time →	getotf(scan#, -1)
RA →	getotf(scan#, -2)
DEC →	getotf(scan#, -3)
AZ →	getotf(scan#, -4)
EL →	getotf(scan#, -5)

We have written a set of UniPops procedures to make quick looks more convenient. To load these procedures into the Line program, type

```
batch fbotf.prc
```

You will then have access to the following procedures:

`ora(scan#)`	Plots the RA offset versus scanning sample (time)

`odec(scan#)`	Plots the DEC offset versus sample
`oaz(scan#)`	Plots the AZ offset versus sample
`oel(scan#)`	Plots the EL offset versus sample
`ot(scan#)`	Plots the UT time array versus sample
`os(scan#,channel#,#bins)`	Plots channel# as scanned across the field averaging over #bins spectra
`os1(scan#,spectrum#)`	Plots a calibrated (on-off/off) spectrum

4.2 Continuum

Coming soon...

5 OFF Scan Editing Utilities

Tom Folkers has written a utility which allows one to modify the OFF measurement information in an OTF data set. The utility *otfmakeoffs* will synthesize an OFF measurement from the (presumably line-free) beginning and end of an OTF map. This utility writes its output files to the directory /home/data4/fixed, which has been assigned the logical name FIXDAT. These output files have the same name as the unedited input files. Below I describe *otfmakeoffs*...

otfmakeoffs: Reads an sdd file and discards any OFF scans. A new OFF scan is synthesized from the begining and end (presumably signal-free) spectra. OFF scan numbers are derived from the original OTF OFF-scan number + 1000...

```
% otfmakeoffs [-e num1 [num2] &| -w num3 [num4]] datafile
```

...where...

-e → Select east end spectra

-w → Select west end spectra

num1 → Number of spectra to include from east end; default 20

[num2] → Optional argument: last east end spectrum to include (so that num1 becomes the first east end spectrum to include)

num3 → Number of spectra to include from west end; default 20

[num4] → Optional argument: last west end spectrum to include (so that num4 becomes the first west end spectrum to include)

For example, if one wanted to use the 5th through 15th spectra from the east end of a map and the 10th through 20th spectra from the west end of a map to synthesize an OFF spectrum for sdd file number 12...

```
% otfmakeoffs -e 5 15 -w 10 20 sdd.jgm_012
```

...while if one wanted to use the 10 spectra from the east end of a map and the 20 spectra from the west end of a map to synthesize an OFF spectrum for sdd file number 12...

```
% otfmakeoffs -e 10 -w 20 sdd.jgm_012
```

6 AIPS Processing

In the following, I describe general techniques for processing spectral line and continuum OTF data taken at the 12 Meter. As a result of a great deal of effort by Eric Greisen, the reduction of OTF data at the 12 Meter is done completely within the AIPS program.

In the following, I assume that you are familiar with the AIPS program syntax and structure. If you are a novice AIPS user, it would be a good idea to review the AIPS Cookbook. The AIPS Cookbook can be browsed using Netscape, which can be run using the Netscape selection in your "Programs" pull down menu on any of the 12 Meter Sun workstations. **Before you begin processing your data in AIPS, you should make sure that UniPOPS is not reading the sdd file you wish to process.** UniPOPS will lock any file it is reading and not allow AIPS to read the file for processing.

For best efficiency, we have set up our AIPS installation at the 12 Meter Telescope to be run on the main analysis workstation (modelo). Therefore, one must first rlogin to modelo before starting the AIPS program...

```
<sonora:jgm> rlogin modelo

<modelo:jgm> aips
```

You will be asked a number of questions by the AIPS startup. When you are asked for a user number, you can either enter your own personal AIPS user number (if you have one) or you can enter either of the general user numbers 2067 (Observer-12m#1) or 2068 (Observer-12m#2). Note that any AIPS data left on disk will be deleted shortly after the end of your observing run.

6.1 Spectral Line

The diagram shown in Figure 6 gives a pictoral description of the processing steps one must follow to reduce spectral line OTF data.

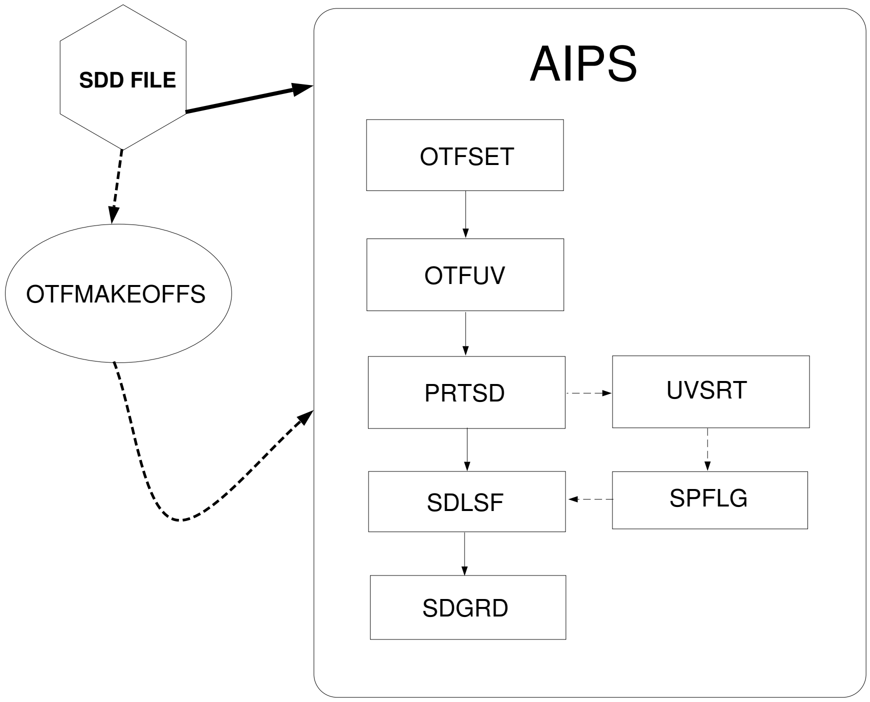

Figure 6: Spectral line OTF data analysis flow chart.

6.1.1 Shotgun Analysis

Now that you are in the AIPS program, you can setup many of the basic parameters and process a map using the AIPS procedures *otfset* and *otfrun*. *otfset* will ask you a number of questions and will use your responses to these questions to set many of the more permanent task parameters that you will use to reasonable values. *otfrun* will ask you a few more questions and use your responses to process a single map. Both *otfset* and *otfrun* will set task parameters for both the "old" and "new" processing routes (see Figure 6). To run *otfset* and *otfrun*, issue the following commands within AIPS...

```
> run otfproc
```

```
> otfset
```

```
> otfrun
```

The RUN command will read the file *OTFPROC.001* in the $RUNSYS area of AIPS and place it into the AIPS core memory. **You should only need to issue the RUN command this one time**. Thereafter, *otfset* and *otfrun* are available for use (even if you exit and re-enter AIPS).

In the following I have reproduced a sample AIPS session where I have used *otfset* and *otfrun* to process a map.

```
>otfset
AIPS 1: ' '
AIPS 1: 'ENTER OBSERVER INITIALS (IN SINGLE QUOTES):'
#'jgm'
AIPS 1: ' '
AIPS 1: '...YOUR OBSERVER INITIALS ARE '     'JGM '
AIPS 1: ' '
```

Your observer initials are the ones assigned to you at the beginning of your observing run.

```
AIPS 1: 'FIRST SET PARAMETERS FOR OTFUV...'
AIPS 1: '(NO USER INPUT NECESSARY)'
AIPS 1: ' '
AIPS 1: ' '
AIPS 1: '...NOW SET PARAMETERS FOR SPFLG...'
AIPS 1: ' '
AIPS 1: ' '
AIPS 1: 'SPFLG AVERAGE TIME SET TO 0.5 SECONDS...'
AIPS 1: ' '
```

The tasks *OTFUV* and *SPFLG* require no user input. You will rarely need to use the task *SDLSF*, but we set the parameters here just in case.

```
AIPS 1: ' '
AIPS 1: '...NOW SET PARAMETERS FOR SDGRD...'
AIPS 1: ' '
AIPS 1: ' '
AIPS 1: 'ENTER HORIZONTAL DIMENSION OF MAP (IN ARCSECONDS):'
#1800
AIPS 1: ' '
AIPS 1: 'ENTER VERTICAL DIMENSION OF MAP (IN ARCSECONDS):'
#1800
AIPS 1: ' '
```

```
AIPS 1: 'ENTER ROW SPACING IN ARCSEC:'
#20
AIPS 1: 'YOUR CELL SIZE WILL BE '              20        ' ARCSECONDS'
AIPS 1: ' '
AIPS 1: 'ENTER FINAL HORIZONTAL AND VERTICAL IMAGE SIZES:'
AIPS 1: '(THESE CAN BE ANY MULTIPLE OF 2)'
AIPS 1: 'GIVEN YOUR CELLSIZE OF '              20        ' ARCSECONDS'
AIPS 1: 'AND IMAGE SIZE OF '      1800        ' X '       1800
AIPS 1: 'I WOULD CHOSE SOMETHING LARGER THAN '             90
AIPS 1: ' X '           90
AIPS 1: 'NOTE: IF THESE ARE LESS THAN 32, CHOOSE 32'
#100
#100
```

For setting the *SDGRD* parameters, I have entered the horizontal and vertical map dimensions ($30' = 1800''$) and observed row spacing ($20''$). I was told by *otfset* that I could use an image size of as small as 90 pixels, but it is a good idea to choose something larger than the minimum suggested just to make sure that your image encompasses all of the data acquired.

```
AIPS 1: ' '
AIPS 1: '...NOW SET PARAMETERS FOR PRTSD...'
AIPS 1: ' '
AIPS 1: 'ENTER CENTRAL CHANNEL IN MAP:'
#64
AIPS 1: ' '
AIPS 1: '...NOW SET PARAMETERS FOR SDLSF...'
AIPS 1: ' '
AIPS 1: '...WILL FIT FIRST-ORDER SPECTRAL BASELINES...'
AIPS 1: ' '
AIPS 1: 'ENTER NUMBER OF BASELINE REGIONS:'
#2
AIPS 1: 'ENTER BEGIN, END, AND INCREMENT CHANNELS FOR BASELINE REGIONS:'
AIPS 1: '(ENTER ONE AT A TIME)'
#5
#25
#1
#100
#125
#1
```

The baseline regions you set for *SDLSF* should be line-free regions which can be used to take-out a spectral baseline from your data. You can use your test position switched measurement to determine the line-free regions.

```
AIPS 1: 'ALL DONE SETTING OTF REDUCTION PARAMETERS'
```

Now that many of the more permanent OTF AIPS task parameters are set, we can run *otfrun*, which will ask a few more questions and actually process your map.

```
>otfrun
AIPS 1: 'SETTING OTF PROCEDURE PARAMETERS'
AIPS 1: ' '
AIPS 1: 'ENTER STARTING SEQUENCE NUMBER TO BE USED FOR ALL OUTPUT FILES:
        '
AIPS 1: '(NOTE...SEQUENCE NUMBERS GREATER THAN OR EQUAL TO THIS VALUE'
AIPS 1: ' MUST BE UNIQUE)'
#1
```

This "sequence number" functions somewhat like the old VMS version numbers did and is a way of distinguishing AIPS data files with the same name.

```
AIPS 1: ' '
AIPS 1: 'ENTER A UNIQUE NAME FOR AIPS OUTPUT FILES (IN SINGLE QUOTES):'
AIPS 1: '(SOME VARIATION ON THE OBJECT NAME WOULD BE A GOOD CHOICE)'
#'serpc18o'
AIPS 1: ' '
AIPS 1: 'YOUR DEFAULT AIPS OUTNAME WILL BE '     'SERPC18O'
AIPS 1: ' '
AIPS 1: 'ENTER AIPS DISK NUMBER TO STORE DATA:'
#2
```

For the AIPS disk to store data, you should choose a number between 1 and 6. All of the output files from the OTF AIPS tasks will reside on this disk, so it is a good idea to choose an AIPS disk with an ample amount of free space.

```
AIPS 1: ' '
AIPS 1: 'HOW MANY SDD FILES ARE YOU GOING TO READ?'
#2
```

I made two maps of this source and would like to process them both.

```
AIPS 1: ' '
AIPS 1: 'ENTER RAW DATA FILE NUMBER (IN SINGLE QUOTES):'
AIPS 1: '(NOTE...MUST BE A NUMBER LIKE 001)'
#'081'
```

```
AIPS 1: 'ARE YOU PROCESSING FILTERBANK OR MAC DATA?:'
AIPS 1: '(ENTER FB OR MAC IN SINGLE QUOTES)'
#'fb'
```

The filter bank data for these maps is in data files sdd.jgm_081 and sdd.jgm_082. If I were to process the millimeter autocorrelator data, it would look for the map data in sdd_hc.jgm_081 and sdd_hc.jgm_082.

```
AIPS 1: 'IS THIS OFF-MADE DATA?:'
AIPS 1: '(I.E. HAVE YOU USED OTFMAKEOFFS?)'
AIPS 1: '(ENTER Y OR N IN SINGLE QUOTES)'
#'n'
```

OFF-made data is data that has been run through *otfmakeoffs* (see §5).

```
AIPS 1: ' '
AIPS 1: '...YOUR SDD FILE IS '
AIPS 1: 'OBSDAT:SDD.JGM_081                                   '
AIPS 1: '...YOUR GSDD FILE IS '
AIPS 1: 'OBSDAT:GSDD.JGM_081                                  '
AIPS 1: ' '
AIPS 1: 'HOW MANY SEPARATE BACKENDS ARE YOU GOING TO LOAD FROM THIS FILE?'
#1
```

I acquired this map in parallel mode, so I have two backends to process which I will need to put into two separate *OTFUV* files. Each file will have both polarizations concatenated for each backend. At this point, I should explain the distinction between IF's and polarizations. An OTF measurement can consist of between 1 and 8 IF's. Each IF can be either an independent frequency, the second polarization of a dual-polarization measurement, or an independent beam element (for the 8-Beam receiver). Within AIPS, data samples are tagged with the IF number by using the "beam" random axis parameter. For example, an OTF map acquired in the first filter bank IF will be IF number 01 and beam number 1, while an OTF map acquired in the first MAC IF will be IF number 11 and beam number 11. This distinction between IF's within AIPS allows one to combine dual-polarization or multiple-beam (8-Beam receiver) measurements into a single UV data file. Tasks such as SPFLG distinguish between IF's within the same data file by reading each IF as an "autocorrelation pair". For example, data samples from IF number 18 are tagged as autocorrelation pair 18-18.

So for our parallel-mode filter bank measurements, OTFUV will read IF's 01 and 02 in one pass, and if you enter 03 for the polarization to load, will actually load IF's 03 and 04 in one pass. In the following, I only loaded the first filter bank set (IF's 01 and 02).

```
AIPS 1: ' '
AIPS 1: 'ENTER BACKEND NUMBER TO LOAD:'
AIPS 1: '(NOTE: ALL POLARIZATIONS ASSOCIATED WITH THIS BACKEND'
AIPS 1: 'WILL BE CONCATENATED)'
#1
AIPS 1: Resumes
AIPS 1: Task OTFUV   has finished
```

Next, we load data from the second data file in the same way as we did for the first data file.

```
AIPS 1: ' '
AIPS 1: 'ENTER RAW DATA FILE NUMBER (IN SINGLE QUOTES):'
AIPS 1: '(NOTE...MUST BE A NUMBER LIKE 001)'
#'082'
AIPS 1: 'ARE YOU PROCESSING FILTERBANK OR MAC DATA?:'
AIPS 1: '(ENTER FB OR MAC IN SINGLE QUOTES)'
#'fb'
AIPS 1: 'IS THIS OFF-MADE DATA?:'
AIPS 1: '(ENTER Y OR N)'
#'n'
AIPS 1: ' '
AIPS 1: '...YOUR SDD FILE IS '
AIPS 1: 'OBSDAT:SDD.JGM_082                              '
AIPS 1: '...YOUR GSDD FILE IS '
AIPS 1: 'OBSDAT:GSDD.JGM_082                             '
AIPS 1: ' '
AIPS 1: 'HOW MANY SEPARATE BACKENDS ARE YOU GOING TO LOAD FROM THIS FILE?'
#1
AIPS 1: ' '
AIPS 1: 'ENTER BACKEND NUMBER TO LOAD:'
AIPS 1: '(NOTE: ALL POLARIZATIONS ASSOCIATED WITH THIS BACKEND'
AIPS 1: 'WILL BE CONCATENATED)'
#1
AIPS 1: Resumes
AIPS 1: Task OTFUV   has finished
```

At this point we have finished loading the raw data into the *OTFUV* data files. We can now begin gridding the data.

```
AIPS 1: '...NOW RUNNING SDLSF ON SEQUENCE 1 ...'
AIPS 1: Resumes
AIPS 1: Task SDGRD   has finished
```

```
AIPS 1: '...NOW RUNNING SDGRD ON SEQUENCE 1 ...'
AIPS 1: Resumes
AIPS 1: Task SDGRD   has finished
AIPS 1: '...NOW RUNNING SDGRD TO CREATE A WEIGHTS IMAGE...'
AIPS 1: Resumes
AIPS 1: Task SDGRD   has finished
```

Note that we have run the *SDGRD* task twice. The first pass through *SDGRD* was to grid the data, while the second pass was to produce a weights image for this data cube which can be used in the *WTSUM* task (see §6.1.3).

```
AIPS 1: ' '
AIPS 1: 'ALL DONE PROCESSING MAP(S)'
AIPS 1: 'YOUR BASELINED CUBE(S) IS (ARE) '     'SERPC180    '
AIPS 1: '.BASE;'          1         '-'            1
```

At this point, you have a processed map cube!

6.1.2 Step-By-Step Analysis

If after running *otfset* and *otfrun* you would like to run each AIPS task independently, you can do so by using the *tget* verb in AIPS. Below I give a description of how to run each task.

OTFUV: Load your raw OTF data into AIPS for further processing using the task *OTFUV*. Since *otfset* has already set many of the basic parameters for this procedure, including any data averaging selections you have made, you need only recall them from AIPS memory...

> `tget otfuv`
>
> `bcount` *first scan to be included in map*
>
> `ecount` *last scan to be included in map*
>
> `inputs`

You need only specify bcount and ecount if you need to exclude bad scans from your map (for example, you might have seen some bad data in your map while looking at it with Dataserve or Databrowse). If you are satisfied with the input values, you can run *OTFUV* with the "go" command...

> `go`

This will produce a UV data file of your OTF map on the AIPS disk you specified in *otfset*. **NOTE:** It is possible to average your 100 ms OTF data to the point where your sampling is greater than or equal to $\frac{\lambda}{3D}$ (1.5 times Nyquist) without sacrificing signal-to-noise or spatial resolution (Emerson, Jewell, & Mangum 1998). This is done by specifying the *yinc* parameter in *OTFUV*. Please see the help and explain files on *OTFUV* before using this method of data smoothing. Note also that we have not sufficiently tested the effects of data smoothing on OTF data, so you should use this feature with caution.

PRTSD: The task *PRTSD* will print out useful information about your data which you can use as a reference for further processing...

> `tget prtsd`

> `getn` *uvdatafilenumber*

> `inputs`

> `go`

SPFLG: The task *SPFLG* will allow you to edit your data using the AIPS TV display. Since *SPFLG* requires that your data be in "TB" order, you need to first run *UVSRT* on your UV data set...

> `tget uvsrt`

> `getn` *uvdatafilenumber*

> `sort 'TB'`

> `inputs`

> `go`

By default, the output UV data file from *UVSRT* will have the class name 'TBSRT'. It is this sorted UV data file that one will edit in *SPFLG* and, if any editing is indeed done, use in the subsequent analysis. The *SPFLG* parameters have been set such that your data will be time-smoothed within *SPFLG* to 0.5 seconds for editing and display. Note that this is not a permanent smoothing of your data, but a necessity due to the limited number of data points that *SPFLG* can process...

> `tget spflg`

> `getn` *sorteduvdatafilenumber*

> `inputs`

> `go`

The default settings will cause *SPFLG* to load the first IF (which is identified internally as autocorrelation baseline 01-01 in your UV data set) of your data set into the TV display with channel number along the horizontal axis and time along the vertical axis. *SPFLG* has many options, which you might like to read about in the EXPLAIN file for the task. Basic processing can be done by using any of the "clip" or "flag" selections. If you have a dual-polarization data set, you can select and edit the second IF by using the "enter baseline" option (since there is only two "baselines" in your data set in this case, *SPFLG* will automatically choose the second one with the "enter baseline" command). **Note, though, that for strong (i.e. maser) sources, it is possible to mistake data samples with signal for bad samples. Be careful!**

SDLSF: The task *SDLSF* will remove a least-squares-fit baseline from your raw (OTFUV) data. The output from *SDLSF* is a baseline-subtracted raw data file. Since *otfset* has already setup the default parameters for you, you need only recall these preset *SDLSF* parameters, tell *SDLSF* what UV data file to process, review the inputs, and go...

> `tget sdlsf`

> `getn` *uvdatafilenumber*

> `inputs`

> `go`

...where *uvdatafilenumber* is the number of the UV data file that you produced in the *OTFUV* step.

SDGRD: The task *SDGRD* will do a number of things to your OTF UV data set. Since your UV data came into AIPS with no projection information, *SDGRD* will apply this information to your data set. The default projection (set by *otfset*) is the TAN (tangent) projection, which is the same projection in which the OTF data was acquired (for small fields). *SDGRD* will then sort your UV data, followed by the actual gridding of the data. Since a Bessel function of the first kind of order 1 ($J_1(r)$), divided by r, and multiplied by a Gaussian, is an appropriate gridding function for OTF data (see Mangum, Emerson, & Greisen 1999), *otfset* has preset the *SDGRD* gridding parameters so that the following gridding function is used...

$$\frac{J_1\left(\frac{|r|}{a}\right)}{\frac{|r|}{a}} \times \exp\left[-\left(\frac{|r|}{b}\right)^2\right]$$

...where...

a = Nyquist sampling rate in pixels = 1.55

b $\simeq$ half of the full width zero intensity for the primary beam = 2.52

The choice of the above function as the default convolution function is due to the fact that it gives the best representation of the response of the telescope beam to the sky distribution. Figures 7 and 8 show this and several other common convolution functions with their associated Fourier transforms.

The output from *SDGRD* is a gridded/interpolated image cube. Since *otfset* has already setup the default parameters for you, you need only recall these preset *SDGRD* parameters, tell *SDGRD* what UV data file to process, review the inputs, and go...

> `tget sdgrd`

> `getn` *uvdatafilenumber*

> `inputs`

> `go`

...where *uvdatafilenumber* is either the number of the UV data file that you produced in the *OTFUV* or the output UV data file from your *SDLSF* step.

TVMOV: You can look at a "movie" of your baseline-subtracted cube channel image planes with the verb TVMOV. Do the following to look at a movie of your baseline-subtracted cube...

> `getn` *baselinedcubenumber*

> `tvmov`

Directions for changing the transfer function, color, frame rate, etc. will appear on your screen.

6.1.3 Combining Multiple Maps

A common observing mode involves making many maps of a given field which are later combined to build-up signal-to-noise. As described above, you can use *OTFUV* to load each polarization from each map data set into one UV data file. If you have read and processed your maps and/or polarizations into AIPS individually and wish to combine them as maps, use the task *WTSUM*. *WTSUM* requires four input maps/cubes; the two data maps/cubes which are to be combined and their respective weights images. One can produce a weights image for a given data cube by using *SDGRD* with REWEIGHT(1) = 3 (the 1/sigma**2 option) and BCHAN = ECHAN = 1 (the data weights are channel independent). *WTSUM* will do a weighted-average of the two input maps/cubes. The procedures *otfset* and *otfrun* produce by default the necessary weights image you will need for an analysis with *WTSUM*. You simply need to set the input parameters...

> `task 'wtsum'`

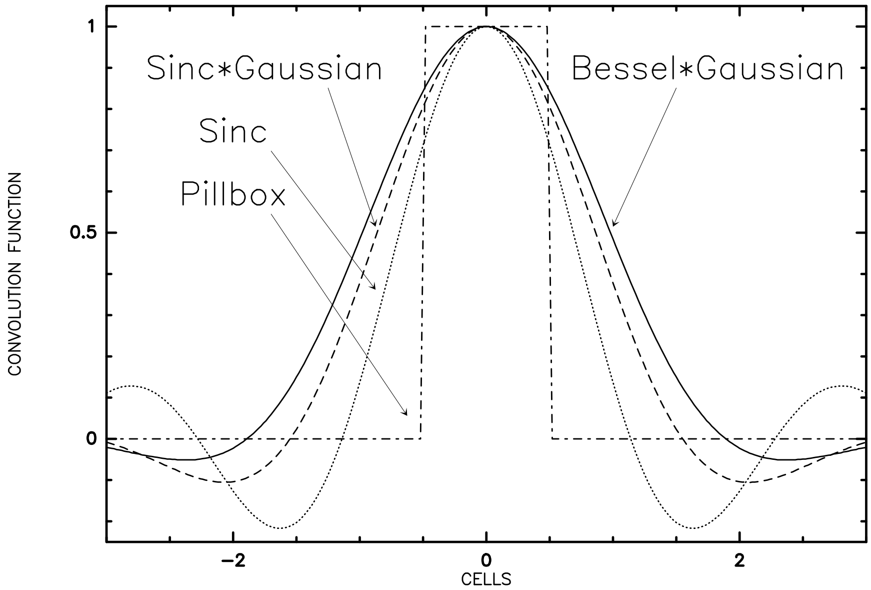

Figure 7: Representative convolution functions.

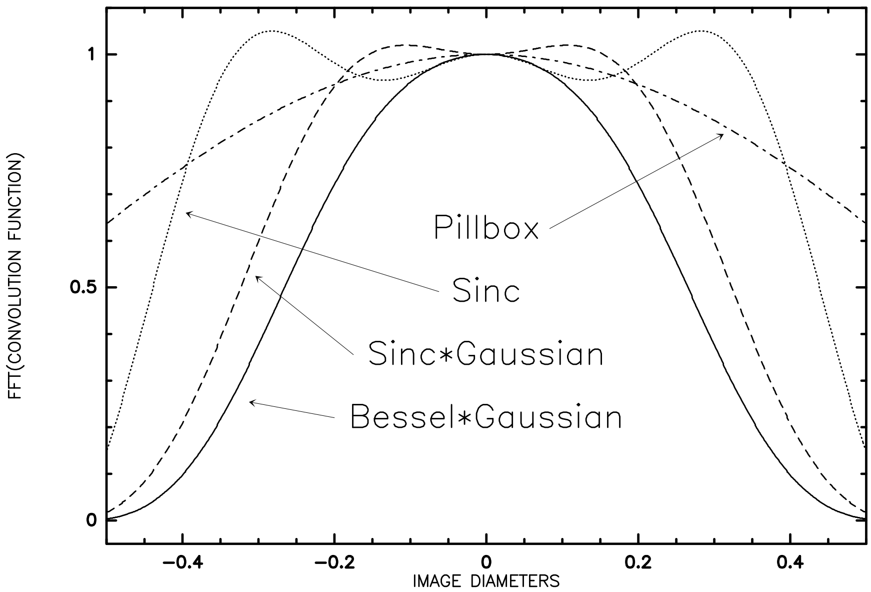

Figure 8: Representative Fourier transforms of convolution functions.

> `indisk` *maplocation*

> `in2disk` *maplocation*

> `in3disk` *maplocation*

> `in4disk` *maplocation*

> `getn` *firstgriddedcube*

> `get2n` *secondgriddedcube*

> `get3n` *firstweightsimage*

> `get4n` *secondweightsimage*

> `doinver -1`

> `go`

...where the *doinver* parameter will assure that we get the correct treatment of the weights images (see the AIPS help or explain files for more information).

6.1.4 Spatial Smoothing

The AIPS task *CONVL* can be used to spatially smooth your gridded data. You should read the AIPS help or explain file for this task for further information.

6.2 Continuum

In the following, I describe a general technique for processing spectral line OTF data taken at the 12 Meter. As a result of a great deal of effort by Eric Greisen, the reduction of OTF data at the 12 Meter is done completely within the AIPS program. The diagram shown in Figure 9 gives a pictoral description of the processing steps one must follow to reduce OTF data.

Coming soon...

Figure 9: Continuum OTF data analysis flow chart.

6.3 More Extensive Processing

6.3.1 Integrated Intensity Images

XSUM: Integrated intensity images can be made using the task *XSUM*. Say you want to make an integrated intensity image of channels 49 through 82 of your processed cube. You need to first make sure that the cube you want to make an integrated intensity image of has velocity as its first axis. You can make such a cube using the *TRANS* task on your baseline-subtracted cube...

TRANS: This step is only necessary if you use the "old" processing route. You now have a channel map cube. You must now correct for any nonlinearities in your spectral baselines. These nonlinearities will show up as declination stripes in your maps. Baseline removal in cubes is done using the *IMLIN* task. But, since *IMLIN* expects your data cube to have axes in an (velocity,RA,Dec) order, you need to first TRANSpose your cube to this axis order (transcode = '312'). Since *otfset* has already set the *TRANS* task parameters for you, you need only recall these preset *TRANS* parameters, tell *TRANS* what file to transpose, review the inputs, and go...

> `tget trans`

> `getn` *baselinedcubenumber*

> `inputs`

> `go`

...where *baselinedcubenumber* is the number of the image cube file that you produced in the *SDGRD* step.

...then run *XSUM* on this cube...

> `task 'xsum'`

> `getn` *transposedcubenumber*

> `outn` *integratedintensityimagename*

> `outdisk indisk`

> `blc 49 0`

> `trc 82 0`

> `opco 'sum'`

> `go`

...where *baselinedcubenumber* is the cube which resulted from the *SDGRD* task and *integratedintensityimagename* is a name of your choosing. An example of an integrated intensity map taken from our sample Serpens data set is shown in the examples for the plotting tasks *CNTR* and *GREYS* given below.

NOTE: The units of this map should really be K*km/s. One can use the AIPS verb *RESCALE* to multiply this integrated intensity image by the channel width (0.845 km/s in this case) and the verb *PUTHEAD* to insert the proper unit name (K*km/s) into the map header. I have done this for the example plots shown below.

6.3.2 Gaussian Fits

JMFIT: Gaussian fits to individual channels can be made using the task JMFIT. Say your integrated intensity image has one main blob of emission to which you would like to fit a Gaussian.

> `task 'jmfit'`
>
> `getn` *integratedintensityimagenumber*
>
> `tvall`
>
> `tvwin`
>
> `ngauss 1`
>
> `ctype 0`
>
> `domax 1`
>
> `dopos 1`
>
> `dowidth 1`
>
> `go`

The TVALL and TVWIN steps will allow you to load the integrated intensity image and determine the box (window) coordinates of the blob you want to Gaussian fit.

For information on other AIPS tasks that you might find usefull for processing your OTF images, see the AIPS Cookbook.

6.3.3 Displaying Your Data

CNTR: Contour plots of individual maps can be made using the task *CNTR*. To make a contour plot of the integrated intensity map made above, I used the following inputs...

> `task 'cntr'`
>
> `getn` *integratedintensityimagenumber*
>
> `clev 1`
>
> `levs 2,3,4,6,8`
>
> `go`

Your plot will be attached to the integrated intensity image as a PL (plot) extension. To display it (either as hardcopy or on your workstation) you will need to use any of the tasks *LWPLA* (for hardcopy output), *TKPL* (for a Tektronix display on your AIPS Tek server), or *TVPL* (for a plot on your AIPS TV display). Figure 10 was produced using *CNTR* and *LWPLA*.

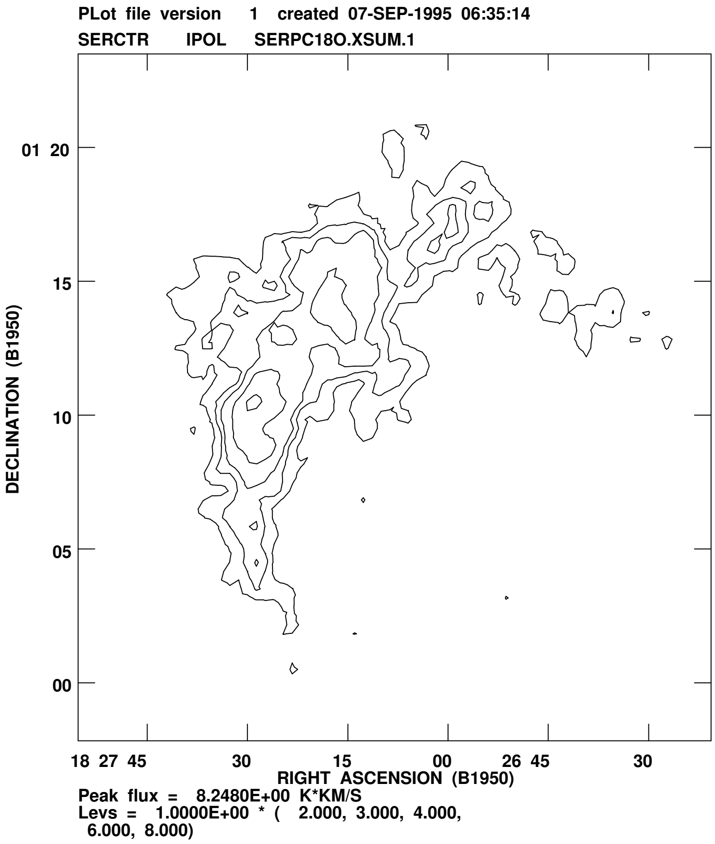

Figure 10: Example output from the *CNTR* task.

GREYS: Greyscale and contour plots of individual maps can be made using the task *GREYS*. To make a *GREYS* plot of the integrated intensity map made above, I used the following inputs...

$>$ `task 'greys'`

$>$ `getn` *integratedintensityimagenumber*

$>$ `get2n` *integratedintensityimagenumber*

$>$ `clev 1`

$>$ `levs 2,3,4,5,6,8`

$>$ `pixra 0 8`

$>$ `dowedge 1`

$>$ `go`

As with *CNTR*, your plot will be attached to the integrated intensity image as a PL (plot) extension. To display it (either as hardcopy or on your workstation) you will need to use any of the tasks *LWPLA* (for hardcopy output), *TKPL* (for a Tektronix display on your AIPS Tek server), or *TVPL* (for a plot on your AIPS TV display). Figure 11 was produced using *GREYS* and *LWPLA*.

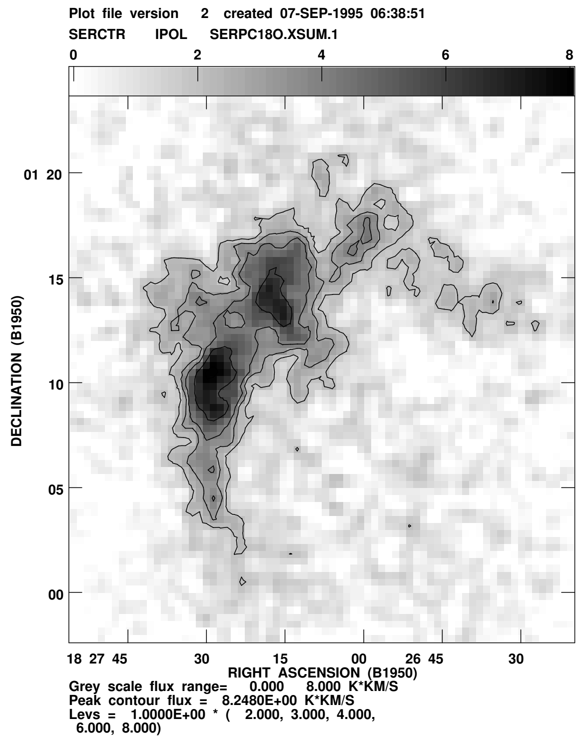

Figure 11: Example output from the *GREYS* task.

KNTR: Contour plots of multichannel data can be made using the task *KNTR*. To make a *KNTR* plot of emission channels 60 through 63 from our processed Serpens data set, I used the following inputs...

> `task 'kntr'`

> `getn` *retransposedcubenumber*

> `clev 0.4`

> `levs -3,3,6,9,12`

> `blc 0 0 60`

> `trc 0 0 63`

> `go`

As with *CNTR*, your plot will be attached to the integrated intensity image as a PL (plot) extension. To display it (either as hardcopy or on your workstation) you will need to use any of the tasks *LWPLA* (for hardcopy output), *TKPL* (for a Tektronix display on your AIPS Tek server), or *TVPL* (for a plot on your AIPS TV display). Figure 12 was produced using *KNTR* and *LWPLA*.

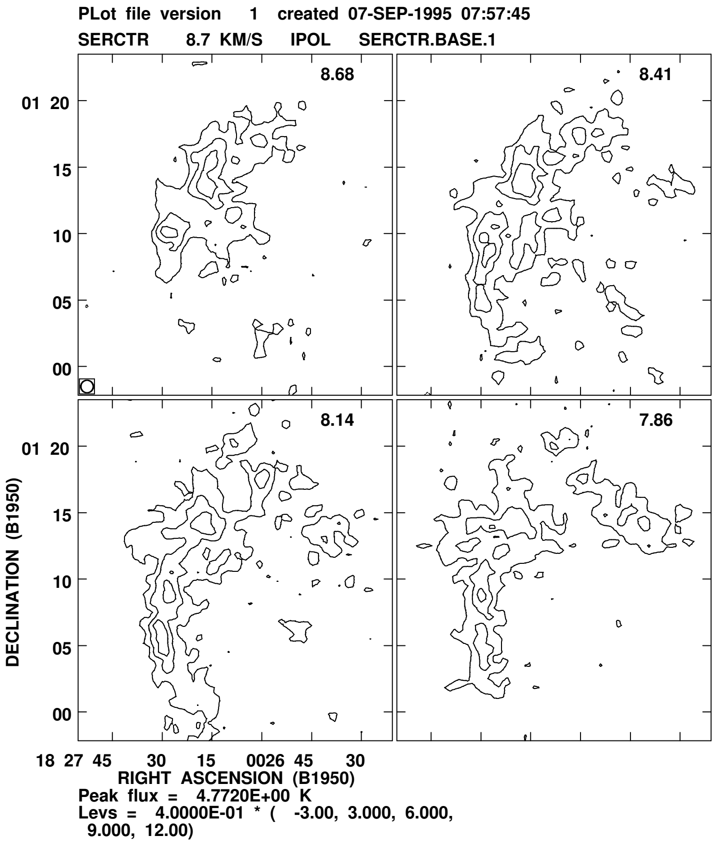

Figure 12: Example output from the *KNTR* task.

PLCUB: Spectral plots of multichannel data can be made using the task *PLCUB*. To make a *PLCUB* plot of the spectra at the central 6 pixels processed (axis order must be (velocity,RA,Dec) Serpens data set, I used the following inputs...

> `task 'plcub'`

> `getn` *baselinedcubenumber*

> `pixra -0.5,3.5`

> `blc 0 31 31`

> `trc 0 33 33`

> `go`

As with *CNTR*, your plot will be attached to the cube file as a PL (plot) extension. To display it (either as hardcopy or on your workstation) you will need to use any of the tasks *LWPLA* (for hardcopy output), *TKPL* (for a Tektronix display on your AIPS Tek server), or *TVPL* (for a plot on your AIPS TV display). Figure 13 was produced using *PLCUB* and *LWPLA*.

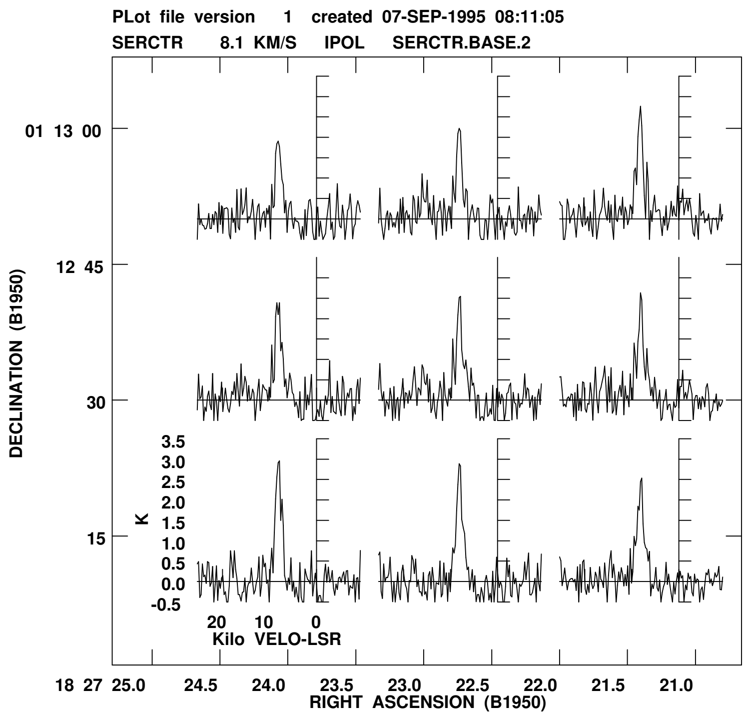

Figure 13: Example output from the *PLCUB* task.