T_{sys} and Weights Within MIR Calibration

Mark Gurwell

14 Feb 2012

T_{sys} calibration serves several purposes.

a) Transfer cross correlation coefficients to approximate physical units (Jy)
b) Correct for atmospheric absorption
c) To properly weight data

The correlator output in it's 'raw' format is essentially a fractional correlation measure; it is a measure of the correlation (over time) of the signal at antenna *i* with that of antenna *j,* as a fraction of the total input, e.g. it is the fraction of the measured geometric mean system temperature that is correlated. Thus, for each baseline pair, the correlated antenna temperature can be determined by multiplying the correlator output by the system temperature for that baseline. This is analogous to the antenna temperature of a source in a single dish measurement. To place onto an approximate physical scale, the antenna temperature is multiplied by the raw antenna gain: for the SMA this is 130 Jy/K. (Here, a digression, feel free to move right along if you want: the SMA correlator units are not quite the fractional power that is correlated. Many moons ago it was determined that there was a scaling factor offset, probably due to an incomplete normalization during one of the FFT. The nomalization factor of $1/\sqrt{2\pi}$ is accounted for within the ***apply_tsys*** routine. See SMA Log Entry 7457 and related poem "Noiserwocky" for all the delirious details).

Note that when we typically talk about 'T_{sys}' what we are really referring to is the system temperature *corrected to outside the earth's atmosphere* (oft denoted as T_{sys}*). It is an estimate of the equivalent 'temperature' of a source at the top of the atmosphere needed to create a T_{sys} defined by the power measured by the system (e.g. $T_{sys}^* = T_{sys}\, e^{+\tau A}$ with τ = zenith opacity and A = airmass). The standard chopper wheel calibration that Eric discussed last week provides an estimate of (DSB, in our case) T_{sys}* rather than T_{sys}, but that's ok, because we want to remove the atmospheric transmission effects as much as possible.

I'm going to use some simplified equations for T_{sys} and T_{sys}* so that the process can be written a bit more clearly and hopefully illuminate how it works. First, T_{sys} is a measure of the power received, and (in the simplification here) is made up of the receiver noise power T_{rx}, the emission from the earth's atmosphere (which is related to the opacity τ and it's temperature T_{atm}), and the equivalent source antenna temperature T_S (we're going to assume it is a point source for the moment), reduced by the atmosphere:

$$T_{sys} = T_{rx} + (1-e^{-\tau A})T_{atm} + e^{-\tau A}\, T_S \qquad [1]$$

[We're ignoring many things here, like certain gain terms, hot spillover, cold spillover, exact form of the source coupling, etc. even how we treat DSB vs SSB measurements]

Now, the 'corrected' T_{sys}* can be written as

$$T_{sys}^* = T_{sys}\, e^{+\tau A} = (T_{rx} + T_{atm})\, e^{+\tau A} - T_{atm} + T_S \qquad [2]$$

Recall that the correlator produces correlation coefficents, that are essentially the fraction of the system temperature that is correlated between any two antennas, or simply $\varepsilon = T_S / T_{sys}^*$; thus to get the (correlated) antenna temperature for a source you simply multiply the correlator output by the corrected system temperature. Multiplying by the standard gain for the antennas provides the visibility amplitude (in approximate Jy units, and modified by the spatial filtering of the baseline for a resolved source):

$$V_{ij} = \varepsilon \times T_{sys}^* \times G \qquad [3]$$

where G is 130 Jy/K for the SMA. Super simple!

<u>Expected Noise</u> From Eric's presentation last week, we also know that the rms noise level, in Jy, on a given visibility point is directly related to T_{sys}*, along with the bandwidth, the integration time, the antenna diameter and efficiency, etc. This is also referred to as the point source sensitivity, and for a single baseline (two element interferometer) with a single polarization the standard formulation is:

$$\sigma_{ij} = \frac{\sqrt{2}\, k\, Tsys*}{\sqrt{BW\, t}\;\; \eta_c Aeff} \qquad [4]$$

where k is Boltzmann's constant, BW is the bandwidth, t is the integration time, and $Aeff$ is the effective collecting area of one antenna (geometric area times aperture efficiency). For an N-element interferometer, there are $N(N-1)/2$ baselines. As with most applications of gaussian random noise, the rms noise reduces by the square root of the number of samples (baselines) and thus for the full interferometer (assuming equal T_{sys}*, $Aeff$, etc) is given by the typical interferometer point source sensitivity equation:

$$\sigma = \frac{2\, k\, Tsys*}{\sqrt{BW\, t}\;\; \sqrt{N(N-1)}\; Aeff} \qquad [5]$$

<u>Weights</u> In MIR we calculate for each visibility point (continuum and spectral bands) the expected rms noise level according to Equation [4]. As with most other applications of error estimation (particularly with gaussian noise), the optimal way to combine data is to weight each data point by the inverse of the variance, and thus each visibility point is given an initial weight

$$w_{ij} = \frac{1}{\sigma_{ij}^2} \qquad [6]$$

The MIR routine ***apply_tsys*** handles all of these tasks:

- multiplies the correlator output by the single sideband system temperature (which in turn is determined to be simply twice the double sideband system temperature, assuming balanced receiver performance and atmospheric transmission)
- multiplies that result with the standard gain of 130 Jy/K appropriate for the SMA antennas, placing the visibility data on an approximate Jansky scale
- calculates weights based upon the bandwidth, system temperature, integration time, etc. with the weights calculated as the inverse of the expected variance for each point.

Further Handling of Weights Within MIR

There are several calibration steps which impact the flux scale, and thus affect the weights. In addition, in certain cases the continuum channel bandwidth can change (for example, exclusion of a strong broad line when generating the continuum from the spectral band data will result in a smaller continuum bandwidth), which will also change the weights of the continuum channel visibility data.

The primary routines that affect the flux density calibration are ***gain_cal*** and ***sma_flux_cal***. These routines seek to improve the calibration of the data by removing time and/or elevation dependent amplitude variations, based upon looking at (one hope's) strong point source calibrators which traverse a similar path through the sky as the target (***gain_cal***) or seek to improve the overall flux scale calibration by observation of a standard source of 'known' flux density, often a moon or planet (both ***gain_cal*** and ***sma_flux_cal***). In either case the visibility data amplitudes are multiplied by a correction factor:

$$V_{ij}' = V_{ij} \times f \qquad [7]$$

where f can be a function of time, elevation, baseline, etc.

For any visibility point, any scaling factors affecting amplitude should also be applied to the expected rms noise (and thus maintain SNR). For example, if ***sma_flux_cal*** suggests that visibility data on a baseline should be scaled by f, then to maintain the SNR of that visibility point the expected rms noise should be scaled by the same factor. Thus, the weight (which is the inverse of the variance) becomes:

$$w_{ij}' = \frac{1}{\sigma'^2_{ij}} = \frac{w_{ij}}{f^2} \qquad [8]$$

Hopefully this makes some intuitive sense: if you have to scale up a visibility point then you should scale down the weight. Both ***gain_cal*** and ***sma_flux_cal*** handle weights in this way.

As mentioned above, if you change the bandwidth of the continuum channel, this will also affect the expected rms noise (as shown in Eq. 4). This in turn will affect the weights:

$$w_{ij}' = w_{ij} \quad \frac{BW'}{BW} \qquad [9]$$

As you increase the bandwidth, you decrease the expected rms noise and thus increase the weight. The MIR tool ***uti_avgband*** is used to generate the continuum from a selected part of the full spectral data, and makes this weight correction.

Finally, a useful little tool in MIR is ***uti_rms*** which can calculate the expected rms noise in a map based upon the weights in the data. To use, simply select the source, sideband and band (can be used for either continuum or spectral band), and then type ***uti_rms***; it will return the expected rms noise you should get when mapping the data. I find that it generally gives a pretty accurate estimate.