GALFA Technical Memo 2005-3

OBTAINING THE I.F. BANDPASS USING LEAST-SQUARES FREQUENCY SWITCHING (LSFS)

December 12, 2005

Carl Heiles

ABSTRACT

In digital heterodyne spectroscopy, the intermediate-frequency (IF) gain has severe frequency dependence. Measurements must be corrected by dividing by this gain. This is usually accomplished by taking a reference spectrum, obtained with either frequency or position switching. However, such switching introduces additional artifacts into the data. Here we describe a way to obtain the IF gain by taking measurements at three or more local oscillator frequencies and using least-squares to extract it, and also the radio-frequency spectrum. We call this technique *Least-Squares Frequency Switching*, or *LSFS*. Here discuss some important mathematical details of the technique and illustrate it with a very successful example in which a large contiguous bandwidth was created by aligning three smaller bandwidths end-to-end.

The Appendix contains documentation of the LSFS software used for GALFA at Arecibo.

Contents

1. INTRODUCTION

Obtaining accurate spectral profiles for Galactic HI is difficult because of the difficulty in obtaining a reference spectrum (the OFF spectrum), which is needed for calibrating the system gain versus frequency (the system bandpass shape). There is no place in the sky where Galactic HI does not exist, so one cannot use an OFF position. Instead, the OFF spectrum is commonly obtained by taking an off-frequency spectrum, moving far enough in frequency so that the HI line is zero. This technique is known as "frequency switching".

At many telescopes, frequency switching produces inaccurate OFF spectra. This occurs because the RF gain and/or the RF power have frequency structure. One contributor is reflections on the telescope structure, for example between the feed and the reflector. If their separation distance is D, then the reflected signal returns with a time delay $\tau = \frac{2D}{c}$. This produces a peak in the autocorrelation function of the received signal with delay τ, which in turn produces a sinusoidal ripple in the frequency spectrum with period $f_\tau = \frac{1}{\tau}$. At Arecibo, $f_\tau \sim 1.0$ MHz, equivalent to about 200 km s^{-1}, which is comparable to the velocity ranges of interest for many HI studies. Telescopes typically have many reflecting paths with different delays, so the received signal has a superposition of ripples with somewhat different periods. These ripples cannot be removed by frequency switching, and in fact are sometimes amplified by an unfortunate choice for the frequency-switching interval.

Here we describe a new approach. Instead of switching the local oscillator (LO) frequency and hoping for good cancellation of the ripple, we set the LO frequency to a number N of different values so that we can *evaluate* the ripple and the bandpass as distinct entities using a unique least-squares technique.

2. REVIEW OF FUNDAMENTALS: POSITION AND FREQUENCY SWITCHING

In heterodyne spectroscopy, we convert the radio-frequency (RF) spectrum to a low intermediate frequency (IF). This conversion is done by multiplying RF power by the local oscillator (LO) in the mixer. Symbolically, denoting frequency by f, we multiply f_{RF} by f_{LO}. This multiplication generates the sum and difference frequencies $(f_{RF} + f_{LO})$ and $f_{IF} = (f_{RF} - f_{LO})$; we remove the sum with a suitable low-pass filter, leaving the desired near-baseband f_{IF}.

The mixer is a transition point between RF and IF frequencies. We can meaningfully discuss the RF and IF sections as separate entities. Thus, the RF section receives from the sky the antenna temperature $T_A(f_{RF})$ and also has the receiver contribution $T_R(f_{RF})$, which is often much larger. These are multiplied by the RF transfer function, known as the gain $G_{RF}(f_{RF})$. Most of the frequency dependence of $G_{RF}(f_{RF})$ occurs in the feed and electronics, which operates on both T_A and T_R, at least to a first approximation. However, often the antenna temperature suffers an additional frequency-dependent gain, which occurs because the incoming power reflects from various portions of the telescope structure and interferes with itself. For simplicity, we neglect this difference and assume that the RF gain affects T_A and T_R equally. Thus, symbolically, the RF power into the mixer $S_{RF}(f_{RF})$ is equal to

$$S_{RF}(f_{RF}) = G_{RF}(f_{RF}) \; [T_A(f_{RF}) + T_R(f_{RF})] \; . \tag{1}$$

The IF section has a transfer function, or gain, $G_{IF}(f_{IF})$. A well-designed system has no additional power contributed at IF. The spectral power measured by the digital spectrometer is provided as a function of the IF frequency, so the appropriate symbol is $P_{IF}(f_{IF})$, which is given by

$$P_{IF}(f_{IF}) = G_{IF}(f_{IF}) \; S_{RF}(f_{RF}) \; . \tag{2}$$

The relationship between the spectral channels and IF frequency is fixed. We access different portions of the RF spectrum by changing the LO frequency.

We now break the $T_A(f_{RF})$ and $T_R(f_{RF})$ into frequency-independent ("continuum") and frequency-dependent ("spectral") portions to simplify further development. With these decompositions, the measured spectral power $P_{IF}(f_{IF})$ depends on the following quantities:

1. $T_A(f_{RF})$. The subscript A means "antenna temperature", so this is the spectral line contribution from the *sky*. The explicit presence of the dependence (f_{RF}) means that there is a spectral dependence, as befits a spectral line.

2. T_A, the antenna-temperature continuum contribution. The absence of the dependence (f_{RF}) means that there is no spectral dependence, as befits continuum radiation.

3. $T_R(f_{RF})$. The subscript R means "receiver temperature", and includes all non-antenna contributions. As with $T_A(f_{RF})$, the explicit dependence on (f_{RF}) denotes only the frequency-dependent portion. For many systems, spectral variations in $T_R(f_{RF})$ are fractionally small.

4. T_R, the receiver-temperature continuum (frequency-independent) contribution.

5. $G_{RF}(f_{RF})$, the RF gain (dependent only on RF frequency). For many systems, $G_{RF}(f_{RF})$ varies slowly with frequency.

6. $G_{IF}(f_{IF})$, the IF gain (dependent on IF frequency). With digital spectroscopy, we must limit the bandwidth by an appropriate IF bandpass filter. This means that G_{IF} varies severely across the band, varying from 0 to full gain.

With these definitions equation 1 becomes a bit more elaborate. When we look at a source in the sky we measure the on-source (ON) spectrum

$$P(f_{IF}) = G_{IF}(f_{IF})\ G_{RF}(f_{RF})\ \ [(T_A(f_{RF}) + T_A) + (T_R(f_{RF}) + T_R)]\ \ . \tag{3}$$

Our goal is to disentangle the sky contribution from everything else, i.e. to obtain $(T_A(f_{RF}) + T_A)$. Being primarily interested in spectroscopy, a modified goal is to obtain only the spectral portion $T_A(f_{RF})$.

We cannot do either of these without dealing with the two gains and the contributions from T_R. It is traditional to deal with these extraneous quantities by taking a reference spectrum, usually denoted the off-source (OFF) spectrum, and arithmetically combining it with the ON spectrum by taking $\left(\frac{ON-OFF}{OFF}\right)$. This process is commonly known as "switching". It works well if the frequency dependencies of the extraneous quantities are benign, but this is not always the case. Let us examine the results of this switching process.

Below we consider two ways of obtaining the OFF, one by moving off in position and one by moving off in frequency. Let *primed* quantities indicate the OFF measurements and *unprimed* the ON. Further, let us simplify the problem by assuming the spectral dependence of the receiver temperature to be small, i.e.

$$T_R(f_{RF}) \ll T_R\ . \tag{4}$$

This allows us to make s Taylor expansions of the expression $\left(\frac{ON-OFF}{OFF}\right)$ for the two types of switching by dropping terms higher than first order. (We make these expansions to minimize complication; if higher-order terms are included, the expressions become more complicated but the techniques can be still applied).

2.1. Position Switching

When the astronomical source is limited in angular extent we can obtain the OFF spectrum by pointing the telescope away from the source. For simplicity, we further assume that the OFF position has no line. Remembering that $T_R = T_R'$, this gives for $\frac{ON-OFF}{OFF}$

$$\frac{P(f_{IF}) - P'(f_{IF})}{P'(f_{IF})} = [T_A(f_{RF}) + (T_A - T_A')] \left[\frac{1 - \frac{T_R(f_{RF})}{(T_A'+T_R)}}{T_A' + T_R} \right] \ . \tag{5}$$

This gives the desired quantity $T_A(f_{RF})$ plus the additive constant $(T_A - T_A')$, which is difference between the antenna temperatures of the two positions. The result is further contaminated by the right-hand multiplicative factor. In effect, this is a frequency-dependent gain. However, its effect on the line shape is small because of our assumption of equation 4. These small effects mean that position is usually the technique of choice.

However, this does not mean that position switching always provides good results. If the difference in continuum temperatures $(T_A - T_A')$ is large, then its multiplication by the right-hand multiplicative factor produces a large effect, and this can make it impossible to distinguish the astronomical spectral line $T_A(f_{RF})$. Thus, position switching can fail for weak lines with strong continuum sources.

2.2. Frequency Switching

If the spectral line is sufficiently spatially extended then we cannot position switch. The prime example is Galactic HI. Here one normally moves off in frequency. This means that the ON and OFF RF frequencies differ, i.e. $f_{RF} \neq f_{RF}'$. In particular, the RF gains differ between the ON and OFF measurements; also, the continuum antenna and receiver temperatures subtract out, i.e. $T_A = T_A'$ and $T_R = T_R'$. Again, to simplify, we assume the ON (unprimed) and OFF (primed) gains don't differ much, i.e. we define

$$\frac{\Delta G}{G} = 1 - \frac{G_{RF}'(f_{RF})}{G_{RF}(f_{RF})} \tag{6a}$$

and assume

$$\frac{\Delta G}{G} \ll 1 \ . \tag{6b}$$

The differing gains introduce a further complication into the Taylor expansion:

$$\frac{P(f_{IF}) - P'(f_{IF})}{P'(f_{IF})} = \left[T_A(f_{RF}) + (T_R(f_{RF}) - T'_R(f_{RF})) + \frac{\Delta G}{G}(T_A + T_R + T_A(f_{RF}))\right] \left[\frac{1 - \frac{T_R(f_{RF})}{(T_A+T_R)}}{T_A + T_R}\right] . \tag{7}$$

This is similar to equation 5 except for the additive term $\frac{\Delta G}{G}(T_A + T_R + T_A(f_{RF}))$ in the first factor on the right-hand side. Even though $\frac{\Delta G}{G} \ll 1$, this term is disastrous because it operates on T_R, which is large. Unless $\frac{\Delta G}{G} \lll 1$, this combination produces serious baseline contamination in frequency switching. Nevertheless, frequency switching works well when, as is often the case, $\frac{\Delta G}{G}$ varies smoothly and slowly with f_{RF} so that it is well-represented by a low-order polynomial fit.

3. DETERMINATION OF $G_{IF}(f_{IF})$ BY LEAST-SQUARES FREQUENCY SWITCHING (LSFS)

The classical approaches of position and frequency switching work only under good conditions. The quantity having the most severe frequency variations is $G_{IF}(f_{IF})$. If we could determine this quantity explicitly we could forgo the switching and, instead, simply divide all measured spectra $P(f_{IF})$ by $G_{IF}(f_{IF})$. Least-squares frequency switching (LSFS) does this explicit determination.

3.1. The Basic Equations and their Iterative Solution

We begin by rewriting equation 3 in a much simpler form. Its right-hand side is the product of the IF gain and several RF quantities. We lump these RF quantities into a single one, $S_{RF}(f_{RF})$:

$$S_{RF}(f_{RF}) = G_{RF}(f_{RF}) \ \ [(T_A(f_{RF}) + T_A) + (T_R(f_{RF}) + T_R)] \tag{8}$$

so we rewrite equation 3 as

$$P(f_{IF}) = G_{IF}(f_{IF}) \ S_{RF}(f_{RF}) \ . \tag{9}$$

Our technique extracts $G_{IF}(f_{IF})$ and $S_{RF}(f_{RF})$ as separate entities.

To proceed, we first express frequencies as channel offsets. The digital spectrometer produces a spectrum having I channels, with channel number i ranging from $i = 0$ to $i = I - 1$. The frequency separation between adjacent channels is Δf. Thus, for the IF frequency of channel i we can write

$$f_{IF,i} = f_0 + i\Delta f \ , \tag{10}$$

where f_0 is a constant. The separation Δf also applies to f_{RF}, so apart from a possible additive constant we have for the RF frequency of channel i

$$f_{RF,i} = f_0 + f_{LO} + i\Delta f \ , \tag{11}$$

where f_{LO} is the LO frequency.

In our scheme, we make measurements at N different LO frequencies, each designated by n. We increment these frequencies in units of Δf, and we write

$$f_{LO,n} = f_{LO,n=0} + \Delta i_n \Delta f \ , \tag{12}$$

where n ranges from 0 to $N-1$. Δi_n is the number of channels that $f_{LO,n}$ is offset from $f_{LO,n=0}$; clearly, $\Delta i_{n=0} = 0$. For convenience we assume Δi_n to increase monotonically with n, so the maximum LO excursion is Δi_{N-1}. We can write all frequencies in units of the channel separation Δf, so the RF frequencies become expressed as digital indices $i + \Delta i_n$, where i is the IF frequency offset from spectral channel zero and Δi_n is the LO frequency offset from the lowest LO frequency (at $n = 0$), both in units of the channel width Δf. Equation 9 becomes

$$P_{i,\Delta i_n} = G_i S_{i+\Delta i_n} \ . \tag{13}$$

There are NI of these equations. We could use them as our equations of condition for the least-squares fit. However, for reasons discussed below, we normalize the variables for computational efficiency. To normalize, we require the means $\langle P_{i,\Delta i_n} \rangle$ and $\langle S_{i+\Delta i_n} \rangle$ to equal unity; of course, this implies that the typical $G_i \sim 1$. Henceforth we assume P and S to be so normalized.

In addition, we express $S_{i+\Delta i_n}$ (whose mean is unity) as an offset $s_{i+\Delta i_n}$ from unity, i.e. we write

$$S_{i+\Delta i_n} = 1 + s_{i+\Delta i_n} \ . \tag{14}$$

Clearly, the mean $\langle s_{i+\Delta i_n} \rangle = 0$. Below we will assume $s \ll 1$, which should be valid as long as the total fractional bandwidth $\frac{[(I-1)+\Delta i_{N-1}]\Delta f}{\langle f_{RF} \rangle}$ is not too large and, also, there are no strong spectral lines. This assumption will be made only for reasons of computational efficiency and does not affect the final solution. For $P_{i,\Delta i_n}$, we can replace the index Δi_n by the index n to reduce clutter. Equation 13 becomes

$$P_{i,n} = G_i + G_i \ s_{i+\Delta i_n} \ . \tag{15}$$

The $P_{i,n}$ are measured quantities and the G_i and $s_{i+\Delta i_n}$ are unknowns too be determined by a least-squares fit. With I spectral channels, N LO frequencies, and a maximum LO excursion of Δi_{N-1} frequency channels, the total number of unknowns is $a = (2I + \Delta i_{N-1})$: there are I unknown values of G and $(I + \Delta i_{N-1})$ values of s. For example, for the calibration spectrum of our Arecibo spectrometer, we have 512 channels so $I = 512$ and we use $\Delta i_{N-1} = 31$, so there are 1055 unknowns. This is a substantial, but hardly impossible, least squares problem. Its solution requires, in essence, the inversion of a 1055×1055 matrix.

We must solve this set of equations using nonlinear least-squares techniques, which are required because both G and s are unknown. Nonlinear least-squares is an iterative process, involving making a guess for the parameters and solving for the difference between the guesses and the true values. Let the guessed values of the parameters be denoted by the superscript g. Then, taking the difference between the true values and the guessed values, and using equation 15, we have

$$P_{i,n} - P^g_{i,n} = (G_i + G_i \ s_{i+\Delta i_n}) - (G^g_i + G^g_i \ s^g_{i+\Delta i_n}) \tag{16}$$

We express the differences between quantities with the symbol δ; thus the unknowns become $\delta G_i = G_i - G^g_i$ and $\delta s_{i+\Delta i_n} = s_{i+\Delta i_n} - s^g_{i+\Delta i_n}$. As usual in iterative schemes, we assume these differences are small and drop second order terms. Also, we divide through by G^g_i, an act which implicitly assumes that its fractional error is small, but as we shall see in fact it doesn't matter if its fractional error isn't small. This gives

$$\frac{\delta P_{i,n}}{G^g_i} = \frac{\delta G_i}{G^g_i}(1 + s^g_{i+\Delta i_n}) + \delta s_{i+\Delta i_n} \tag{17}$$

We have turned the nonlinear least-squares problem into an iterative linear one.

However, the presence of the term $s^g_{i+\Delta i_n}$ means that the equation-of-condition matrix changes from one iteration to the next. The number of unknowns is large, and the need to evaluate the inverse matrix for each iteration requires significant computational time. We can eliminate this burden by dropping the term $s^g_{i+\Delta i_n}$ in the above equation. This yields our final equation, which is used in an iterative solution in which each step is a linear least-squares fit for the two sets of unknowns—and has the highly desirable property of an unchanging equation-of-condition matrix. The two unknowns for each channel (i) and each LO setting (Δi_n) are now $\frac{\delta G_i}{G^g_i}$ and $\delta s_{i+\Delta i_n}$:

$$\frac{\delta P_{i,n}}{G^g_i} = \frac{\delta G_i}{G^g_i} + \delta s_{i+\Delta i_n} \tag{18}$$

Now, it might seem that the elimination of the term $s^g_{i+\Delta i_n}$ is an arbitrary action that produces erroneous results. However, this is not a problem. As $\frac{\delta G_i}{G_i} \rightarrow 0$—i.e., as we attain convergence—this term goes to zero so the final solution is unaffected. And, miraculously, it *does* converge—usually rapidly.

The set of equations 18 does not sufficiently constrain the solution. One more equation is needed to keep the mean RF power $\langle S \rangle$ constant (i.e., equal to unity), which is the additional constraint

$$\sum_{i,n} \delta s_{i+\Delta i_n} = 0 \tag{19}$$

To solve this set of equations iteratively, we begin with initial guesses $G_i^g = 1$ and $s_{i+\Delta i_n}^g = 0$. This provides the initial $P_{i,n}^g = 1$. We least-squares solve equation 18 for the $\frac{\delta G_i}{G_i^g}$ and $\delta s_{i+\Delta i_n}$, which provides the new guesses G_i^g and $s_{i+\Delta i_n}^g$. There are, of course, NI such equations, I for each of the N LO frequencies. We usually obtain convergence in ~ 10 iterations, which takes a fraction of a second on a modern computer for $NI = 1055$.

3.2. Number of Equations of Condition; Number of Unknowns

In least-squares fitting we develop equations of condition, one for each observed quantity. Least-squares fitting requires the unknowns to be overdetermined, i.e. the number of unknowns to be smaller than the number of equations of condition. As mentioned above just after equation 15, the number of unknowns is $a = (2I + \Delta i_{N-1})$. The number of equations of condition is $M = NI + 1$: I channels for each of the N LO frequencies, plus equation 19. For the least-squares technique we require $M > a$, i.e.

$$NI > 2I + \Delta i_{N-1} - 1\ . \tag{20}$$

Clearly, it makes no sense to have $\Delta i_{N-1} > I$, because that generates additional values of $s_{i+\Delta i_n}$ while providing no information on G_i. Thus we require $\Delta i_{N-1} = hI$, where $h < 1$. This yields

$$N > 2 + h - \frac{1}{I} \tag{21}$$

With $h < 1$, we generally require $N \geq 3$.

3.3. An Illustrative Textbook Example

We present an illustrative example with the goal of clarifying the procedure. The simplest example has the smallest numbers. Since we need $N \geq 3$ and $I > \Delta i_{N-1}$, we choose $I = 4$ and $N = 3$, with $\Delta i_n = [0, 1, 3]$. This provides an arithmetic progression for the successive frequency differences $\Delta i_{nn'}$, which we define as $(\Delta i_n - \Delta i_{n'})$ (where we consider all combinations of n and n').

The values of $\Delta i_{nn'} = [0, 1, 2, 3]$. Note that $\max(\Delta i_{nn'}) = \delta i_{N-1}$. In matrix form, the equations of condition (equations 18) are

$$\mathbf{X} \cdot \mathbf{a} = \mathbf{p} \ . \tag{22}$$

Our notational convention is that boldface small letters are vectors and boldface large letters are matrices. In the vector of unknowns $\mathbf{a}$, to avoid clutter we write g_i in place of $\frac{\delta G_i}{G_i^g}$. This vector is

$$\mathbf{a} = \begin{bmatrix} g_0 \\ g_1 \\ g_2 \\ g_3 \\ \delta s_0 \\ \delta s_1 \\ \delta s_2 \\ \delta s_3 \\ \delta s_4 \\ \delta s_5 \\ \delta s_6 \end{bmatrix} \ . \tag{23}$$

In the the vector of measured quantities $\mathbf{p}$, we write $p_{i,n}$ in place of $\frac{\delta P_{i,n}}{G_i^g}$. This vector is

$$\mathbf{p} = \begin{bmatrix} p_{0,0} \\ p_{1,0} \\ p_{2,0} \\ p_{3,0} \\ p_{0,1} \\ p_{1,1} \\ p_{2,1} \\ p_{3,1} \\ p_{0,2} \\ p_{1,2} \\ p_{2,2} \\ p_{3,2} \\ 0 \end{bmatrix} \ , \tag{24}$$

and the equation-of-condition matrix is

$$
\mathbf{X} = \begin{bmatrix}
1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\
1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\
1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\
0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\
0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 1 & 1
\end{bmatrix} . \tag{25}
$$

In this matrix,

1. The first four rows are for the lowest LO frequency with $\Delta i_0 = 0$
2. The next two pairs of four rows have $\Delta i_1 = 1$ and $\Delta i_2 = 3$.
3. The last row is the power conservation equation 19.
4. The first four columns are for the four IF gains.
5. The last seven columns are for the RF powers.

The usual least-squares process of solving these equations of condition (see Press et al 1992) involves multiplying $\mathbf{X}$ by its transpose to obtain the curvature matrix (the matrix of normal equations) and then taking the inverse of that matrix product to obtain the covariance matrix $\boldsymbol{\alpha}$:

$$
\boldsymbol{\alpha} = (\mathbf{X}^{\mathbf{T}} \cdot \mathbf{X})^{-\mathbf{1}} , \tag{26}
$$

and the solution for the coefficient vector is

$$
\mathbf{a} = \left(\boldsymbol{\alpha} \cdot \mathbf{X}^{\mathbf{T}}\right) \cdot \mathbf{p} . \tag{27}
$$

For this illustrative problem, the inverse is well defined with no numerical problems. The biggest normalized covariance (or correlation), obtained by converting the covariance matrix to a correlation matrix, is -0.51, whose absolute value is not unreasonably large.

3.4. Larger I, Larger N

We have not investigated how the quality of the solution changes with N. Our initial experiments used $I = 736$, $N = 4$, and a seemingly arbitrary set of Δi_n (see below); this worked well. Nevertheless, common sense would seem to suggest the following criteria:

1. An arithmetical progression for $\Delta i_{nn'}$, beginning with unity and incrementing by unity up to some up to some maximum value. It seems to us, intuitively, that incrementing by unity is akin to sampling uniformly when doing Fourier transforms, which is always the desirable situation—but this is only an intuitive hunch not based on any objective investigation.

2. A reasonably large value for h. Our current practice, which applies to $I = 512$, is to adopt $N = 7$ with an arithmetical progression for $\Delta i_{nn'}$; this leads to $\Delta i_{N-1} = 31$, making $h = 0.0605$. Intuitively this seems small, but nevertheless it gives good results.

The problem of generating an arithmetic progression with N placings of the LO frequency is akin, in radioastronomical interferometry, to the well-studied problem of generating an arithmetic progression of baselines with a linear array of telescopes. For N LO frequencies there are $\frac{N(N-1)}{2}$ frequency spacings, some of which are redundant; similarly, for N telescopes on the ground there are $\frac{N(N-1)}{2}$ distance spacings, some of which are redundant. The classic discussion by Moffett (1968) considers these minimum-redundancy telescope arrays, which use N antennas to generate a minimally redundant arithmetical progressive series for $\Delta i_{nn'}$. Zero redundancy is possible only for $N \leq 4$; for $N = 4$ the spacings range up to 6. For $N > 4$ there must be some redundancy. Moffett presents two types of minimum-redundancy arrays, *restricted* and *general*.

Restricted minimum-redundancy arrays provide all spacings $\Delta i_{nn'}$ up to a maximum, with no gaps; these are useful for radio interferometry when the available real estate is limited. General ones provided all spacings up to a particular limit, and in addition provide larger spacings. In particular, the $N = 7$ restricted array provides all spacings $\Delta i_{nn'} \leq 17$, while the $N = 7$ general array provides all spacings $\Delta i_{nn'} \leq 18$ and, in addition, $\Delta i_{nn'} = [24, 26, 31]$. For our purposes the general array is a better choice: apart from the fact that we are not limited by available real estate, it provides more different values for $\Delta i_{nn'}$ and, also, a larger value of h.

For the general minimum-redundancy arrays, roughly $\Delta i_{N-1} \approx 31 \left(\frac{N}{7}\right)^{2.66}$. This is a steep dependence, so it doesn't take an unreasonably large number of LO frequencies to get a desired h even if N is large.

It isn't strictly necessary for $\min(\Delta i_{nn'})$ to equal unity. If one has prior knowledge that the IF gain varies slowly over some range R of channels throughout the bandpass region of interest, then one can bin the data into groups of R channels. This both increases h for a given N and also reduces I, thus speeding up the computational time. We discuss this case in §3.5 and §4.2 below.

3.5. An Illustrative Real-World Example

Our initial experiments with LSFS were performed with three independent banks of Arecibo's interim corrrelator. Each bank had 2048 channels and covered a bandwidth of 25 MHz; we overlapped the three banks by cutting off 64 channels on each end, i.e. we spaced the centers by $\frac{31}{32} \times 25$ MHz, stitching together a single spectrometer with 5888 channels covering a total contiguous bandwidth of 71.875 MHz. We binned the channels by a factor of 8, making 736 channels of width 0.0977 MHz. We used four different LO frequencies with spacings $\Delta i_n = [0, 35.50, 43.69, 72.36]$, providing the nonuniform set of six spacings $\Delta i_{nn'} = [8.19, 28.67, 35.50, 36.86, 43.69, 72.36]$. Figure 1 shows the results.

LSFS works well in this initial-experiment example. This is a somewhat extreme case, with very large variations of $G_{IF}(f_{IF})$ within the band because of the three different IF filters and, because of the absence of a uniformly-progressing series of frequency separations, a seemingly (and, we presume, actually) unintelligent source of the four LO frequencies[1]. This particular case illustrates how one can reliably generate a contiguous large band from narrower ones. LSFS also works very well with the GALFA spectrometer, for which the calibration spectrum is 100 MHz wide with 512 channels.

Obtaining the RF power $S_{RF}(f_{RF})$ is only the first step in obtaining the desired antenna temperature $T_A(f_{RF})$. The extraneous quantities $T_A(f_{RF})$ and $G_{RF}(f_{RF})$ need to be removed. Over a modest RF bandwidth this is normally done by polynomial fitting and subtraction; with the GALFA spectrometer we determine polynomial coefficients for the 100 MHz wide "calibration band", interpolating over the 7 MHz "science band", and use those coefficients to correct the science band. For wider bandwidths the quantities $T_A(f_{RF})$ and $G_{RF}(f_{RF})$ should be determined independently, which can be done by taking measurements on and off of a continuum source.

4. DETAILS OF MATRIX ALGEBRA FOR LARGER I, LARGER N

Under some conditions we find empirically that some of the equations of condition become degenerate. This degeneracy is best understood by considering the $\mathbf{X}$ matrix in equation 22 as a series of a column vectors, where a is the number of unknowns; $a = 2I + \Delta i_{N-1}$. Suppose that two columns are degenerate, meaning that they are linear combinations of each other. In the matrix product $\mathbf{X} \cdot \mathbf{a}$, there is a one-to-one relationship between each column in $\mathbf{X}$ and its corresponding unknown coefficient in $\mathbf{a}$. A degeneracy between two columns means that the matrix product cannot distinguish between the two corresponding coefficients.

When one applies the usual technique of generating the normal equations and inverting the curvature matrix, as in equation 26, it doesn't work: the inverse matrix doesn't exist because of

[1]We cannot recall why we chose this particular set of Δi_n.

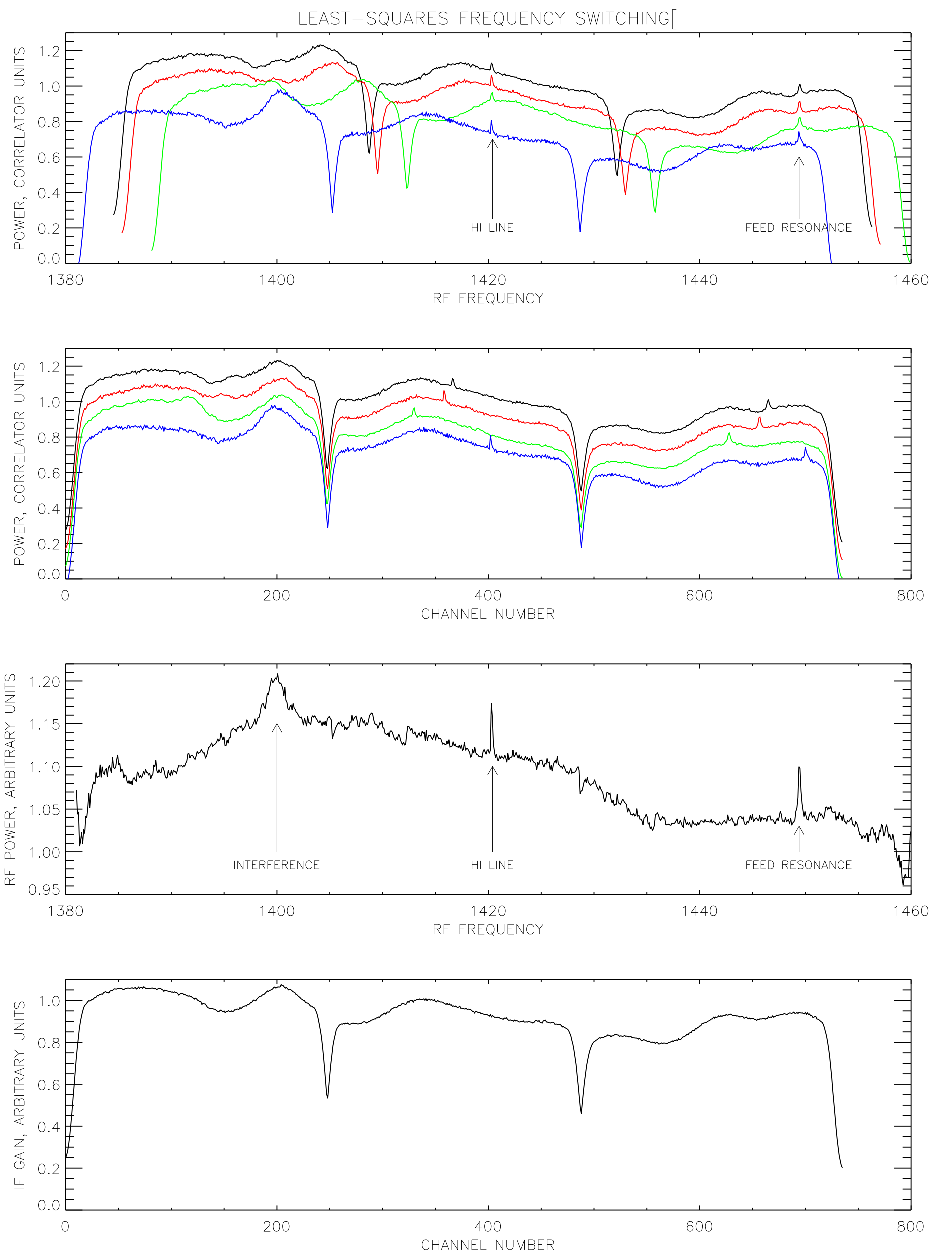

Fig. 1.— Real-world example of LSFS. The top panel shows the total measured power $P_{IF}(f_{RF})$ versus RF frequency for the four LO frequencies; note that the HI line remains fixed near 1420 MHz and the IF bandpass shapes move with the LO. The second panel shows the measured power $P_{IF}(f_{IF})$ versus IF frequency for the four LO frequencies; the IF bandpass shapes remain fixed and the HI line moves with the LO. The third panel shows the derived RF power, $S_{RF}(f_{RF})$ in equation 9, with the known spectral peaks identified. The bottom panel shows the IF gain $G_{IF}(f_{IF})$.

the degeneracy. In cases like this one has two choices: think hard and find the root cause of the degeneracies; or use Singular Value Decomposition (SVD) to empirically remove them. The number of unknowns is large, so the first option—picking through a huge matrix looking for degeneracies—is difficult, probably even for a mathematical expert. Therefore, we choose the latter one.

4.1. SVD: Theory

Numerical Recipes (Press 1992) provides a useful discussion of the SVD technique as applied to least squares. The SVD technique forgoes the usual generation of the normal equations for calculating the matrix $\left(\boldsymbol{\alpha} \cdot \mathbf{X^T}\right)$ in equation 27. Rather, it expresses this matrix in terms of three matrices that are derived from the SVD decomposition of the $\mathbf{X}$ matrix.

The cornerstone of SVD is that any $M \times a$ matrix, where the number of rows M and the number of columns a satisfy $M \geq a$, can be decomposed as the product of three matrices. In particular, our matrix $\mathbf{X}$ in equation 22 satisfies this criterion, so we can write

$$\mathbf{X} = \mathbf{U} \cdot [\mathbf{W}] \cdot \mathbf{V^T} \ , \tag{28}$$

where

1. $\mathbf{U}$ is $M \times a$, $[\mathbf{W}]$ is $a \times a$ and diagonal, and $\mathbf{V^T}$ is $a \times a$. The square brackets around the matrix $\mathbf{W}$ indicate that it is diagonal.

2. The columns of $\mathbf{U}$ and $\mathbf{V}$ are unit vectors that are orthonormal. Because $\mathbf{V}$ is square, its rows are also orthonormal so that $\mathbf{V} \cdot \mathbf{V^T} = 1$. Recall that, for square orthonormal vectors, the transpose equals the inverse so $\mathbf{V^T} = \mathbf{V^{-1}}$.

Degeneracies are directly reflected in the $\mathbf{V}$ and $[\mathbf{W}]$ matrices. The square matrix $\mathbf{V}$ consists of a set of a orthonormal column vectors. These are normalized orthogonal versions of the *non*orthogonal column vectors in $\mathbf{X}$. Suppose that L columns of $\mathbf{X}$ are degenerate. Then the number of independent orthonormal vectors represented in $\mathbf{X}$ is decremented by L. Nevertheless, the $\mathbf{V}$ matrix still contains a orthonormal vectors. The decrement by L is represented not by the orthonormal column vectors in $\mathbf{V}$, but rather by their *weights* in the diagonal matrix $[\mathbf{W}]$. Each column vector in $\mathbf{V}$ has an associated weight in $\mathbf{W}$, and if there is degeneracy, then the corresponding value of $\mathbf{W}$ is zero. This means the corresponding orthonormal vector in $\mathbf{V}$ cannot be represented by the column vectors in $\mathbf{X}$.

Having derived the SVD components of $\mathbf{X}$, we can write

$$(\boldsymbol{\alpha} \cdot \mathbf{X^T}) = \mathbf{V} \cdot \left[\frac{\mathbf{1}}{\mathbf{W}}\right] \cdot \mathbf{U^T} \ . \tag{29}$$

Now suppose there is degeneracy; then the corresponding values of $[\mathbf{W}]$ are zero, so the corresponding values of $\left[\frac{\mathbf{1}}{\mathbf{W}}\right]$ become infinite. This is an attempt by the matrix algebra to represent the space defined by the corresponding columns of $\mathbf{V}$ with data that were taken with inappropriate values of $\mathbf{X}$.

SVD, as applied to least squares, handles these infinities by setting the corresponding values of $[\frac{\mathbf{1}}{\mathbf{W}}]$ (which are formally equal to ∞) to zero. This provides stable, realistic solutions in which the offending degenerate coefficient values are close to being correct—or, at least, not being totally unreasonable. By following this procedure, one can handle degeneracies without understanding their cause simply by zeroing out the relevant components of the $\left[\frac{\mathbf{1}}{\mathbf{W}}\right]$ vector.

This zeroing process can—and should—be applied in cases of *near* degeneracy. Just exactly what "near" means depends on the noise in the data, because the data values are amplified by $\left[\frac{\mathbf{1}}{\mathbf{W}}\right]$ in calculating the coefficient values. For sufficiently noisy data one might best zero out the offending elements of $\left[\frac{\mathbf{1}}{\mathbf{W}}\right]$ even if they are not too terribly large.

We use IDL for our data analysis. IDL has two routines for SVD, `svdc` and `la_svd`. The latter is more robust in performing the SVD decomposition and is preferred; it was introduced in IDL version 5.6.

4.2. SVD: Practice

The $\mathbf{X}$ matrix depends only on the values Δi_n, not on the data values S. Moreover, calculating the inverse of the $\mathbf{X}$ matrix is computationally expensive. Thus, when using a particular set of LO frequency offsets for multiple observations, it behooves one to do the SVD calculation of $(\boldsymbol{\alpha} \cdot \mathbf{X}^{\mathbf{T}})$ once, store the results on disk, and read them back when necessary. This has the further advantage that one can examine the weight vector $[\mathbf{W}]$ once for each set of LO frequency offsets, decide which particular values of $\left[\frac{\mathbf{1}}{\mathbf{W}}\right]$ to set to zero, and forget about dealing with this on a case-by-case basis.

Our experience is as follows. For our initial experiments with four LO frequencies, there was no degeneracy. Neither is there degeneracy for our current practice of of 512 channels and 7 LO frequencies. Here the 7 LO frequencies are the general minimum-redundancy set, with increments $\Delta i_{nn'}$ ranging from 0 to 31 in steps of unity (with $\Delta f = \frac{100 \text{ MHz}}{512} \approx 0.195$ MHz).

We use the same set of 7 LO frequencies for the narrowband science spectrum on the Galfa spectrometer. This spectrum covers a bandwidth of $\frac{100}{14} \approx 7.143$ MHz and has 8192 channels, which makes $\Delta f \approx 872$ Hz. Thus, the LO increment Δf is about 224 times the channel spacing. Of the 8192 channels, 7679 are recorded. We invent an extra, making the total 7680. Before applying LSFS we rebin these, lumping successive bins of 16 together so that the total number of binned channels is 480. The frequency offset of the seventh LO frequency is almost as large as the total bandwidth, so we use only the first six LO frequencies. This gives increments $\Delta i_{nn'}$ ranging from 14 to 283, so $h = \frac{283}{480} \approx 0.59$. We assume smoothness and use an interpolation procedure to recover the 7679 values of $G_{IF}(f_{IF})$.

This narrowband case exhibits thirteen degeneracies. We have 1198 coefficients to derive, so we have 1198 orthonormal column vectors in the $\mathbf{V}$ matrix. The 1198 weights $[\mathbf{W}]$ range from about 8.9×10^{-7} to 27. Thirteen weights are nearly degenerate, being smaller than 1.8×10^{-6}; we set their corresponding inverses to zero. The lowest nonzeroed weight is ≈ 0.69, so there is a huge gap between the range of accepted weights (~ 0.69 to 27) and the zeroed inverse weights (the largest is $\sim 1.8 \times 10^{-6}$). Zeroing the inverse weights provides a very nice solution.

5. POLARIZED STOKES PARAMETERS OBTAINED FROM CROSSCORRELATION

LSFS also applies to polarized Stokes parameters obtained from crosscorrelation. For example, if the native polarization is crossed linears, the propertly Mueller-matrix calibrated crosscorrelation outputs provide Stokes U and V. The uncalibrated crosscorrelation products are not Stokes parameters but instead simply products. For crossed linears, it is convenient to call these XY and YX. The treatment of these is identical so below we discuss only one, XY.

Our preceding discussion applies to a single native polarization; for native linears, it is convenient to call these XX and YY. The application of the LSFS procedure to XX and YY provides their associated r.f. powers and, more importantly for now, the gains G_{XX} and G_{YY}. These are power gains; the associated voltage gains are their square roots. The cross products XY and YX are the products of the voltages. Thus, as we wrote for XX [equivalent to $P(f_{IF})$] equation 3, for XY we write the analogous

$$\begin{aligned} XY(f_{IF}) = [G_{XX,IF}(f_{IF})G_{YY,IF}(f_{IF})]^{1/2}[G_{XX,RF}(f_{RF})G_{YY,RF}(f_{RF})]^{1/2} \\ [(XX_A(f_{RF}) + XX_A) + (XX_R(f_{RF}) + XX_R)] \ , \end{aligned} \tag{30}$$

where the subscripts A and R on XX have the same meanings as in equation 3, with A meaning a contribution from the sky and R from the r.f. portion of the receiver. Note that we assume that there is no contribution from crosscorrelation in the i.f.; this assumes perfect isolation of the two polarization channels, both in the electronics and in the spectrometer. Deviations from this assumption can be revealed by inspecting the data, and the appropriate modifications of the current treatment invoked.

It is straightforward to duplicate the steps leading to equation 13. There is one simplification here: the i.f. gains are *known* from the previous LSFS analysis of XX and YY. It is convenient to include these known quantities as modfying the data, i.e. to rewrite the eqivalent of equation 13 as

$$\frac{XY_{i,\Delta i_n}}{G_i} = S_{XY,i+\Delta i_n} \ . \tag{31}$$

The left-hand side contains the measured quantities and the right the desired unknown ones. Because there are no multiplying factors, the lease-squares solution for the unknowns is just the appropriate average of the left-hand quantities. Alternatively, we can follow the line of development pursued in §3 and express the solution as a least-squares problem using matrices. This is more time-consuming computationally (i.e., perhaps a few seconds at most) but much simpler programatically and offers more flexibility. In contrast to the the situation in §3, the present problem in a linear one so it is not necessary to do an interative solution. In particular, we do not need to add the additional constraint embodied in equation 19.

Referring to §3.2, here the number of unknowns is $M = NI$ (I channels for each of the N l.o. frequencies) and because we don't have to determine the gains (δG_i) the number of unknowns is $a = (I + \Delta i_{N-1})$. This gives $N > (1 + h)$, which more easily satisfied than the corresponding equation 21 for XX. For textbook example of §3.3, the current $\mathbf{X}$ matrix that corresponds to equation 25 looks like

$$\mathbf{X} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} . \tag{32}$$

This work was supported in part by NSF grant 95-30590 to CH and, also, by the NAIC. Josh Goldston and Snezana Stanimirovic read an early version of the manuscript and made several important suggestions.

REFERENCES

Moffett, A. 1968, IEEE Trans Ant Prop AP-16, 72

Press, W.H., Teukolsky, S.A., Vetterling, W.T., & Flannery, B.P. 1992, Numerical Recipes in C, Second Edition, section 15.4

---

6. APPENDIX: GALFA'S IDL PROCEDURES FOR LSFS

This appendix documents the LSFS software for GALFA at Arecibo. It describes the nesting and functions of each procedure. There is additional documentation in each procedure, which can be accessed from the usual `doc_library` call in IDL.

The tree of GALFA LSFS IDL procedures includes procedures that (1) prepare the GALFA data for LSFS analysis, and (2) procedures that actually do that analysis. As usual, the former dominate. The latter procedures can be used generally for non-GALFA data; they include `xmatrixeval`, `xmatrixgen`, and `carl9`. The procedure tree looks like this:

- `find_smartf` . Given a list of fits datafiles, this looks at the `mh` files and selects the particular ones that contain SMARTF data.
- `smartf_groups` . Takes the output of `find_smartf`, which is a list of which files contain SMARTF data, and groups them into arrays containing the beginning and ending indx numbers in the array of filenames. For example if the SMARTF data were taken at the beginning and end of the run, it will find two groups. Then the LSFS reduction can be done, and saved, separately for each group. This grouping is why we added the UTC seconds in the LSFS file name (see below).
- `lsfs_shell` . This is a wrapper whose input is a list of contiguous FITS files containing SMARTF data. It reads these files, organizes the data, and writes the derived information into an IDL save file. The file name contains date, time, and project number info. An example:

  `galfa.20050713.1121257541.a2004.0005.sav`

  Here, 20050724 is the date on the original fits datafile, 1121257541 is the UTC seconds since 1971, a2004 is the project number, 0005 is the sequence number for the SMARTF data in the original fits file. You don't have to know the UTC to read this file; it can be read with

  `restore, 'galfa.20050713.*.a2004.0005.sav'`

  Note: possible problem with `lsfs_shell`*!* This program reads all fits files in the group before selecting the particular records to process. If there are too many fits files in one SMARTF group, then the resulting array will become too large for the machine's memory and IDL will issue the corresponding message and stop. This is on my list to fix, but it is low priority.

  Note: the UTC seconds field is being added in mid-July 2005. All software will need to be rewritten, and all data will need to be rerun or filenames changed, to accommodate this addition.

  – `lsfs` The input is the list of contiguous fits filenames containing SMARTF data. The output is the various arrays that are solutions of LSFS.

- ∗ `m1_to_m1s_s0` . The input is the list of contiguous fits filenames containing SMARTF data. Of course, not all spectra in any file are actually SMARTF. The output is the array of `m1` structures in those files that are actually labeled SMARTF. This array is called `m1s` .
- ∗ `lsfs1` . The input is the array `m1s` produced by `m1_to_m1s_s0` . This program is good at doing the horrendous job of deciding which data are really SMARTF, which data are simply so labeled but are not SMARTF, which data to discard because it might contain an LO or CAL transition, etc.

  The most important outputs are arrays that are dimensioned [2,7,7,2,4]. The indices mean: `[pol,rx,LO,cal,cycle]` The `cycle` means: each combination of 7 LOs and 2 cals is one cycle. The procedure will return up to four contiguous cycles. These arrays are averaged over all spectra that are valid, true members of each state, typically about 10 spectra.

  The most important outputs are:

  1. `PWRWB_AVG, PWRNB_AVG [2,7,7,2,4]` . The wb and nb powers (averaged over all channels) for each combination of LO frequency, CAL, and cycle.
  2. `G_WIDE_AVG, G_NB_AVG [nchan,2,7,7,2,4]` . These are the wb and nb spectra for each combination of LO frequency, CAL, and cycle.
  3. `LOFRQS_LSFS_OUT`, the seven LO frequencies used for the LSFS calibration.
- ∗ `carl_preliminary` . Inputs are the spectral powers like `G_WIDE_AVG` and the seven LO frequencies from the output of `lsfs1` . The outputs are `FNEW`, the RF frequencies of the channels of the derived RF power (there are `INDXTOT` $= 2I + \Delta i_{N-1}$) such channels; see section 3.2); `STAMOD`, the interpolated spectral powers (in case the LO frequency offsets are not exactly integral channel offsets), `FRQOAMOD`, the corresponding interpolated frequencies.
- ∗ `xmatrixeval` . Input is `YESNO`, whose value is used to evaluate parameters telling the software to either evaluate the $\mathbf{X}$ matrix, do its SVD decomposition, evauate the matrix $(\boldsymbol{\alpha} \cdot \mathbf{X^T})$ in equation 29, or read the results from disk. See §4.1.
- ∗ `xmatrixgen` . This generates the matrix $\mathbf{X}$ of equation 22. This is only called if the output parameters of `xmatrixeval` are set.
- ∗ `carl9` . Performs the iterative solution described in §3.1. *Note:* `carl9` uses the IDL routine `svdc`, which is not as good as the `la_svd`, which was introduced in IDL version 5.6. This should be changed.

- `ggnb_recon` . Inputs are the RF freqs and gains for both wb and nb. For the LSFS fit the nb spectra are binned down from 7680 to 480 channels. `ggnb_recon` does the interpolation and reconstitution and returns 7679 channels of $G_{IF}(f_{RF})$. This interpolation process is complicated and I'm not sure I'm doing it correctly: it needs attention. And it may be that we should take a special set of measurements for the NB spectra, using a different set of LO frequencies.