MEMVIS: Maximum Entropy Method using Visibilities

Andrew Conway

October 25, 2000

1 Introduction

Before describing MEMVIS, I will outline the basics of the Maximum Entropy Method (MEM) as it is applied to image reconstruction for HESSI (MEM SATO).

Observations are made in the form of counts in time-bins, each time-bin corresponding to a particular roll angle. There are nine sets of counts, one set from each detector. The aim is to find an image which is consistent with these counts to within the photon counting noise. Since there can be many images consistent with the observations, MEM is designed to find smoothest such image. Traditionally, consistency is measured using a χ^2 measure on the counts:

$$\chi^2 = \sum_i \frac{(N_i - E_i)^2}{\sigma_i^2}$$

where N_i are the expected counts and E_i are the counts from the reconstructed image, and σ_i is the estimated error of count N_i. If only photon counting noise is important then $\sigma_i = E_i$, but in practice systematic errors and uncertainties in the background may also be important and so should be included in σ_i. It is important to note that E_i should be used to estimate the error on the counts rather than the observed value N_i. E_i need not be an integer, and will rarely (if ever) be zero. These properties are crucial for low counts. When there is less than one expected photon per bin, then the C statistic should be used instead of the χ^2 statistic. The smoothness of the image is quantified using an entropy measure:

$$S = -\sum_j F_{ij} \log F_{ij}$$

where j labels the pixels in the image. A smooth image will have a larger S. For example, two images of P pixels, one with only one 'lit' pixel, the other with constant pixel brightness, that both have the same total flux F will have S values of $-F \log F$ and $-F \log(F/P)$ respectively. The goal of MEM is to find the smoothest image that fits the data. In practice, this means maximising $S - \frac{\lambda}{2}\chi^2$ for the smallest possible λ. The idea is that λ is set to a small value to begin with and an image is sought iteratively which has $\chi^2 = 1$, i.e. it is consistent with observations. If such an image cannot be found, the smoothness constraint is probably too strong, and so λ is increased to allow a less smooth image.

The main steps in the implementation of MEM for HESSI using counts (MEM SATO) are as follows:

1. The iteration starts from an image where every pixel has the same intensity (the grey map). The total flux of the image is estimated by summing all the counts, and multiplying by a factor (about 4 for HESSI) accounting for the grid transmission.

2. This image is used to calculate an expected set of counts.

3. The expected counts from the image are compared to the observed counts, and the differences are used to adjust the image so as to increase $S - \frac{\lambda}{2}\chi^2$.

4. If the difference in counts from the new image has $\chi^2 \sim 1$, then the process is ended.

5. If the change in χ^2 is very small, then λ is increased, effectively relaxing the smoothness constraint slightly, which might then allow χ^2 to be brought closer to 1.

6. Return to step 3

The MEMVIS algorithm uses the same basic steps as outlined above, but works in polar co-ordinates, and uses visibilities rather then counts. The advantages of using visibilities are:

1. The aspect correction can easily be applied whilst constructing visibilities from the observed counts.

2. Due to 1, visibility modulation patterns do not need to be corrected for the aspect solution for each time bin.

3. Due to 2, operations involving modulation patterns can be performed with a convolution based on Fast Fourier Transform methods, rather than a matrix multiplication. This allows a speed up of order M^2 as compared to $M \log M$, where M is the number of time bins.

4. Due to 2, integration of visibility values over several rotations simply involves adding together visibility series from those rotations.

5. In constructing visibilities from observed counts, there is effectively a smoothing, which relaxes problems arising from the presence of zero counts. The price is the introduction of a small systematic error and a correlation of noise on visibilities in neighbouring time bins.

2 Visibilities

The notion of visibilities is borrowed from radio astronomy, though here we will only consider visibilities in the context of HESSI, making clear their relation to observed counts. In the present context, 'counts' and 'visibilities' can simply be thought of as two time series for presenting the same information: the response of HESSI to an image. For this reason, in the MEMVIS application, if there are M timebins of counts in one rotation then there will be M visibilities. Counts are non-negative integer values, whereas visibilities are complex numbers, though the real and imaginary parts of the visibilities do not contain independent information.

It is easiest to understand the relation between visibilities V_{mk} and counts N_{mk}, where m labels the timebin and k the detector, if the spin axis is fixed, i.e. not pointing to different points on the Sun at different times as it will in practice. The real parts of the visibilities are just equal to the mean-subtracted counts:

$$Re(V_{mk}) = N_{mk} - \bar{N_k} \tag{1}$$

where $\bar{N_k}$ is the mean over all timebins for detector k. Each member of the time series for a given k corresponds to different azimuth in the transform of a 2D image in polar co-ordinates. Each detector on HESSI responds to a spatial frequency corresponding to the pitch of the detector. In this sense, the visibilities obtained from HESSI's counts are analogous to samples from a 2D Fourier transform of the source in polar co-ordinates. The samples are on concentric circles at evenly spaced points on each circle. The points on the same circle have the same k, and the points at the same azimuth have the same m. The largest circles, corresponding to the shortest spatial frequencies from the finest detectors naturally need the most points. The counts, which correspond to the real part of the transform, may therefore be thought of as a cosine transform. The word 'analogous' is used above because the transform is not based on sinusoids, as is a Fourier transform, but is based on the profile shown in Fig. 1. This is the visibility time series, or profile, that would obtained from a point source. It has a period equal to the rotation period (only one period is shown in the figure), but contain cycles of varying length within each rotation period.

The visibilities from detector k, time bin m, of an image F_{ij} are defined mathematically as

$$V_{mk} = \sum_{ij} P_{ij,mk} B_{ij} \tag{2}$$

The matrices P are visibility modulation patterns which are defined (for the first harmonic):

$$P_{ij,mk} = e^{-i\mathbf{K}_{mk}.\mathbf{r}_{ij}}$$

where $\mathbf{r}_{ij}$ is the position vector of pixel i, j in the image and $\mathbf{K}_{mk}$ is the position vector of pixel m, k in the visibility transform (on what radio astronomers call the UV plane). If Cartesian co-ordinates are used in which the pixel centre is

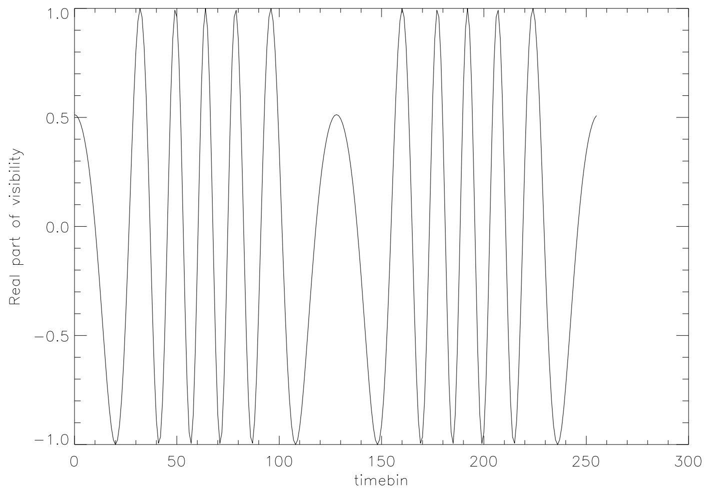

Figure 1: The visibility profile for detector 8 with 256 time bins.

at (x_i, y_j) then

$$\mathbf{K}_{mk}.\mathbf{r}_{ij} = K_k \cos \alpha_{mk} x_i + K_k \sin \alpha_{mk} y_i$$

where α_m is the roll angle for timebin m for detector k. $K_k = 2\pi/p_k$ where is the pitch of detector k. In polar co-ordinates, where the pixel centre is at (r_i, θ_i) it is given by:

$$\mathbf{K}_{mk}.\mathbf{r}_{ij} = K_k r_{ij} \cos(\alpha_{mk} - \theta_i)$$

Modulation matrices for the same detector k, but different timebins m are simply related to each other by rotation about the centre of the co-ordinate system (which is chosen to be Sun centre, though this choice is really arbitrary). For example, if $P_{ij,05}$ (timebin 0 for detector 5) is displayed as an image, then $P_{ij,m5}$ will simply be the same image rotated by $2\pi m/M_k$, where M_k is the number of timebins for detector k. In contrast with modulation patterns used elsewhere in the HESSI software, these modulation patterns are *never* adjusted to account for motion of the spin axis. Instead, the observed visibilities, constructed from the observed counts, are corrected for wandering of the spin axis. For this reason, visibilities can *literally* be added from one rotation to the next to produce a visibility time series integrated over many rotation. This is not possible with counts because they cannot be easily corrected for wandering of the spin axis.

There are good reasons to work with visibilities in polar co-ordinates. For one thing, as just discussed, visibility modulation patterns are simply related to one another by rotation. In dealing with polar visibilities as arrays on a computer, a rotation only requires shifting the elements in the array (with wrap around at the ends).

In working with visibilities it can be helpful to understand them in their continuous form. Working with a (continuously defined) map in polar co-ordinates $F(\theta, r)$ the visibilities $V(\alpha, k)$ are defined similarly to above

$$V(\alpha, k) = \int_0^{2\pi} \int_0^{\infty} P(\alpha, k; \theta, r) B(\theta, r) r dr d\theta \tag{3}$$

For the first harmonic, the modulation patterns for visibilities are given by

$$\begin{aligned} P(\alpha, k; \theta, r) &= \exp\left(-i\mathbf{k}.\mathbf{r}\right) \\ &= exp\left(-ikr\cos(\alpha - \theta)\right) \end{aligned} \tag{4}$$

The real part of P is just the standard modulation patterns (normalised to 1, with 1 subtracted). For a point source of brightness 1, $F = \delta(r - r_0)\delta(\theta - \theta_0)/(2\pi r)$, the visibilities are:

$$V(\alpha, k) = \exp\left(-ikr_0\cos(\alpha - \theta_0)\right)$$

which for given k, plotted against θ, will have the same profile as shown in Fig. 1.

It is worth noting that the modulation patterns are assumed constant in amplitude and harmonic content only within a small region of the Sun. So the artifice of using the full 360 degrees of azimuth produces accurate maps only in the neighbourhood of one azimuth. This is because the internal shadowing of the grids has a slowly-varying dependence on radius and azimuth. In practice, one has to pick grid transparency and modulation amplitude values (defined in Section 5) characteristic of one point on the Sun before making a map.

From here onwards, it should be understood that discrete modulation patterns $P_{ij,mk}$ represents a polar, visibility modulation pattern where m labels time bin, k labels the detector, i is the azimuthal pixel index, and j is the radial pixel index. In MEMVIS a polar map B_{mn} has pixels with constant size in r and θ. This means that a pixel's area increases with r, but there will be the same number of phi pixels for each radial index n. All maps will be assumed to be sun-centred.

3 Observed Counts to Visibilities

Before MEMVIS begins the iteration loop, the observed counts time series must be converted to a visibility time series with the same number of timebins. Here I will outline the process, the full details of the which can be found in Ed Schmahl's routine `hsi_calib_ev2vis.pro`. The first step is to divide the count time series into its cycles. Each cycle is fitted with a function of the form $A\cos(\omega m) + B\sin(\omega m)$, where m is the time bin index, and $\omega = 2\pi/\tau$, where τ is the length of the cycle in timebins. The A and B coefficients can then be used to construct the real and imaginary part of the visibilities for that cycle. The whole visibility time series can be constructed, one cycle at a time, in this manner. However, these visibilities are not the ones as defined above; they are spin axis centred, not sun-centred. The spin axis centred visibilities are related to the sun-centred visibilities by adding a varying phase factor in the visibility modulation pattern's exponential in equation (4). In practice therefore, because the visibilities are related linearly to P, they are easily be converted to sun-centre by multiplying P by $\exp(ia)$, where a can be obtained from the aspect solution. The visibilities then need to be shifted back or forth by a number of time bins to account for the relative orientation of each detector's grid in azimuth (i.e. the grid angle). (This is probably one source of the strange offsets in the visibility time series in MEMVIS - the visibilities should be interpolated rather than simply shifted by an integral number of time bins).

To use the visibilities in the MEM reconstruction we also need to know the variance on each visibility. To do this, we can take advantage of the fact that each visibility can be expressed as a weighted sum of counts:

$$V_{kj} = \sum_l w_{kjl} C_l$$

where the sum usually only involves a small subset of the count time-bins for detector k. The complex weights w are calculated during the visibility reconstruction process, and may be used to calculate the variance associated with the real part of visibility V_{kj} from the formula:

$$\left(\sigma_{kj}^R\right)^2 = \sum_l Re\left(w_{kjl}\right)^2 C_l$$

with the formula for the imaginary part simply involving changing Re() to Im(). Because visibilities in neighbouring time-bins depend on common count values, the noise on adjacent visibilities will be correlated. This will broaden the χ^2 distribution's peak, but does not alter the mean. The degrees of freedom of χ^2 as used with HESSI are very large, certainly no less than 64 for the coarsest detector. This means that the peak is very narrow to start with, and is only broadened by a small amount because of the correlated noise on the visibilities. Using Monte-Carlo tests designed to re-create the χ^2 distribution of the reconstructed visibilities I have confirmed this conclusion, even for low count-rates (less photons than time bins) where the use of the χ^2 becomes invalid for the counts. The reason for this is almost certainly the smoothing that is introduced in the construction of visibilities from the counts.

In addition to the variance of the visibilities arising from photon counting statistics, we must also know the systematic error arising from the count to visibility conversion. There is no easy way to do this on a theoretical basis, though methods of estimating it during the conversion process are being devised and tested in cases where the noiseless visibilities are known. Currently, apart from small offsets that are currently present (and easily corrected for), the systematic error in the visibilities is not important until the simulation parameter **sim_photon** is about 100,000, and is difficult to reliably measure because of small time-shift offset that is present in the visibilities.

4 Implementation of MEM for polar visibilities

The MEM equations in MEMVIS are derived in a similar manner as described by Sato, Kosugi and Makishima (1999). The main difference is that the visibilities are complex, and so require the maximisation is performed with respect to their real and imaginary parts separately. The function to be maximised is

$$\begin{aligned} Q &= S + \lambda_E E + \sum_{mk} \left[\lambda^R_{mk} Re(D_{mk}) + \lambda^I_{mk} Im(D_{mk})\right] \\ &- \frac{\lambda}{2} \sum_{mk} \left[\left(\frac{Re(D_{mk})}{\sigma^R_{mk}}\right)^2 + \left(\frac{Im(D_{mk})}{\sigma^I_{mk}}\right)^2 \right] \end{aligned} \tag{5}$$

where m is the time bin index and k is the detector index. The first term is the entropy $S = -f_{ij} \log f_{ij}$, which is largest when the image is smoothest. $f_{ij} = F_{ij}/\bar{F}$ are the pixel fluxes F_{ij} normalised by their average value $\bar{F}$. The second and third terms are constraints, with associated Lagrange multipliers λ^R_{mk} and λ^I_{mk}, that $E = 0$ and $D_{mk} = 0$, where:

$$D_{mk} = \sum_{ij} \bar{F} f_{ij} P_{ij,mk} - V'_{mk} + n_{mk} \tag{6}$$

$$E = \sum_{ij} f_{ij} - N_p^2 \tag{7}$$

where V'_{mk} are the observed visibilities. The E constraint enforces the normalisation of f_{ij}, so that its sum is equal to the number of pixels. The D_{mk} constraint enforces the fact that n_{mk} are the residuals from subtracting the actual visibilities from the predicted visibilities from the image. Note the constraints on the real and imaginary parts of D_{mk} have to be enforced separately, because the maximisation of Q would not be possible if it were a complex quantity. The final term is just $-\frac{\lambda}{2}\chi^2$, where λ is the smoothing parameter.

The method used by MEMVIS to maximise Q is fixed-point iteration. The iteration equations are obtained by setting partial derivatives of Q with respect to f_{ij}, n^R_{mk} and n^I_{mk} to zero, and re-arranging the expressions to give, respectively:

$$f_{ij} = e^{\lambda_E - 1} e^{\bar{F} \sum_{mk}\left[\lambda^R_{mk} Re(P_{ij,mk}) + \lambda^I_{mk} Im(P_{ij,mk})\right]} \tag{8}$$

$$\lambda^R_{mk} = \lambda \frac{Re(n_{mk})}{(\sigma^R_{mk})^2} \tag{9}$$

$$\lambda^I_{mk} = \lambda \frac{Im(n_{mk})}{(\sigma^I_{mk})^2} \tag{10}$$

where λ_E is determined by inserting the first of these equations into the constraint that $E = 0$, using (7). This means that $\lambda_E - 1$ represents the log of the normalization factor $N_p^2 / \sum f_{ij})$.

An iteration equation for $\bar{F}$ is not included in MEMVIS at present because the visibilities cannot be used to determine the total flux of the image. In contrast with standard modulation patterns (i.e. those for calculating counts from an image), visibility modulation patterns can be negative and positive This means that response to a source in visibilities can be positive or negative, whereas for standard modulation patterns it is always positive. In principle therefore, it is easy to imagine that there are distributions of source flux which "cancel" out in the visibilities. The problem is clearly illustrated for a map of constant intensity. The counts observed for such an image would be (nearly) the same in every time bin - the constant value being related to the total image brightness. The visibilities for such an image would be (nearly) zero in every time bin, containing no information about the total flux of the source. There are many potential solutions to this problem - e.g. searching for the dimmest image that matches the observations, adding a non-zero offset to the visibilities according to the average observed counts - but understanding the effect on the final solution image requires careful consideration. For the moment the total flux is estimated at the start of the algorithm and fixed to this value. The total flux is estimated as follows:

$$B_{tot} = \frac{1}{N_d} \sum_k \frac{1}{N_t} \sum_m \frac{C_{mk}}{T_{mk} * \tau_{mk}}$$

(I might be including the modulation amplitude as a factor on the denominator in the code***) where C_{mk} is the count from detector k in time bin m, T is the grid transparency, τ is the live-time, N_d is the number of detectors and N_t is the number of timebins.

The variance of B_{tot} due to photon counting statistics will be

$$B_{tot} = \frac{1}{N_d} \sum_k \frac{1}{N_t} \sum_m \frac{C_{mk}}{(T_{mk} \tau_{mk})^2}$$

However, another source of error arises because the normalisation by the grid transparency T is not quite correct. The reason can be seen by considering the basic modulation profile from a point source of flux F, located at polar co-ordinates (r, θ)

$$M(\alpha) = FT\left[1 + A\cos\left(Kr\cos(\theta - \alpha)\right)\right]$$

where A is the modulation amplitude $K = 2\pi/p$, p being the pitch of the detector. In averaging this over roll angle α, as is done in calculating B_{tot}, the cos term does not necessarily average to zero. This means that dividing by g will not give the brightness F. The average of M over α, $\bar{M}$ is periodic in Kr, having maxima/minima at $Kr = j\pi$, where j is an integer. The average also decreases with increasing Kr, so that this problem is least severe for the finest detectors, and sources close to the limb. At $Kr \sim 100$, the cos term average will be about 5% of the average of the first term. In practice, this error will be reduced because of the motion of the spin axis and if the source has extended structure (this needs to be checked in more detail in test simulations).

5 Implementation of MEMVIS equations

In polar co-ordinates, and because the visibility modulation patterns do not need to be corrected for spin axis motion, it is not necessary to calculate every element of the visibility modulation pattern matrix $P_{ij,mk}$. In fact, only $P_{ij,0k}$ (the modulation for the 0th time bin) is needed because it can be rotated (shifted in i) to make the modulation pattern for the mth timebin. If the number of time bins for detector k is set equal to the number of ϕ pixels the relation is just $P_{ij,mk} = P_{(i+m)j,0k}$. This saves a considerable amount of memory in storing the visibility modulation patterns. From here on, we will refer to the visibility modulation patterns as $P_{ij,k}$, i.e. those for detector k for the first time bin.

One iteration of MEMVIS (step 3 in the list of steps in Section 1) implements the MEMVIS equations (8-10) as follows:

1. Perform back projection to obtain an image for each detector
2. Combine these images and normalise to obtain one image
3. Combine this image with the one from the previous iteration
4. Calculate the visibilities from this image
5. Calculate χ^2

The n_{mk} values are calculated as the difference between the observed and expected visibilities (??). They are then used to calculate the new λ_{mk} values (9,10). The back projection step is the evaluation of

$$b_{ij,k} = \bar{F} \sum_m \left[\lambda^R_{mk} Re(P_{ij,mk}) + \lambda^I_{mk} Im(P_{ij,mk}) \right]$$

which appears in the second exponential in (8). $b_{ij,k}$ is back projection image for detector k. In the MEMVIS code, the back projected images from different detectors (each term in the k sum) can have different numbers of θ pixels (though they all cover the same area); the coarser detectors having the fewest (and therefore largest) pixels. This is because the number of α pixels must equal to the number of time bins used for that detector. Apart from the advantage this brings in saving memory (explained at the start of this section), it also allows a convolution to be used. In back projection, the convolutions performed are on the right hand side of the following equations:

$$\sum_m \lambda^R_{mk} P_{ij,mk} = \sum_m \lambda^R_{mk} P_{(i-m)j,0k} \qquad (11)$$

$$\sum_m \lambda^I_{mk} P_{ij,mk} = \sum_m \lambda^I_{mk} P_{(i-m)j,0k} \qquad (12)$$

These convolutions can be performed using Fast Fourier Transform methods which can have significant speed advantages. If there are N_t α pixels, then straightforward evaluation of the above equations involves N_t terms in the summation, which must be done N_t times (for each i). The calculation speed is therefore proportional to N_t^2. The Fourier convolution has a calculation speed of $N_t \log_2 N_t$, assuming N_t is a power of 2. For $N_t = 2^{10} = 1024$, the speed up is therefore of order 100. This speed up is in comparison to the matrix method that was initially used in the HESSI imaging software. The introduction of Richard Schwartz's `annsec` modulation patterns (sometime in early 2000) has significantly improved the speed of the matrix operations. The methods used to achieve this increased speed do not scale in a simple way with N_t, and can depend on the structure of the image involved. As a result the only meaningful way to compare the relative speeds is by tests on the same computer, on the same test images.

Combining the images from back projection first involves rebinning each detector's back projected image to the correct size (corresponding to the requested number of pixels, and pixel sizes). This rebinning can cause small time-shifts between the predicted and observed visibilities. Whether this is a problem or not will depend on the number of photons (more photons, and therefore lower noise may expose these time-shifts) and on the detector in question (small shifts are more important for the finer detectors which have faster modulation cycles). If the rebinning is causing problems, then the problem can be easily solved (at a cost of speed) by increasing the number of time bins specified by the keywords `time_bin_def` and/or `time_bin_min`.

Once the individual images have been rebinned and added together the final image is obtained by putting the summed image through the exponential function (the second exponential in (8)). This step has the effect of accentuating the largest pixel values. The resulting image is then normalised (multiplication of the first exponential in (8)) to give f_{ij} and then multiplied by $\bar{F}$ to give actual pixel brightnesses F_{ij}^{new} of the new image.

The next step is to combine the image F_{ij}^{old}, obtained in the previous iteration, with the current image F_{ij}^{new} to give the resultant image for the present iteration F_{ij}:

$$F_{ij} = gF_{ij}^{new} + (1-g)F_{ij}^{old}$$

where g is the gain factor, which is less than 1 and is altered throughout the iteration process (the method for altering it is taken from MEM SATO, and so may not be optimised for MEMVIS***). This combining of old and new images serves to stabilise the iterative process, by introducing a kind of momentum that prevents drastic changes in one iteration ruining the current solution.

The next step is to calculate the visibilities from the new image:

$$V_{mk} = \sum P_{ij,mk} B_{ij}$$

This step can also be speeded up by using the convolution method described for back projection above. As before, it is necessary to rebin the image so that it has the same number of α pixels as $P_{ij,mk}$, which is equal to the number of time bins for detector k.

The χ^2 for this iteration can then be computed as

$$\chi^2 = \frac{1}{N_d}\sum_k \frac{1}{N_t}\sum_m \frac{1}{2}\left[\frac{Re(V_{mk} - V'_{mk})^2}{(\sigma^R_{mk})^2} + \frac{Im(V_{mk} - V'_{mk})^2}{(\sigma^I_{mk})^2}\right]$$

where V'_{mk} are the observed visibilities (i.e. those constructed from the counts time series). If the change in χ^2 between iterations is such that $\Delta\chi^2/\chi^2 < \delta$, then the smoothness parameter is incremented by a certain amount. The size of the change depends on how much χ^2 has changed since the last increment in λ. The iteration process will stop once χ^2 is less than 1, or if the maximum number of λ iterations is reached. The most recent image is then turned into a Cartesian map and returned to the calling routine.

6 Software details - new objects

MEMVIS has been written in an object oriented way, and (by design) has the exact same input parameters as MEM SATO, though the way these parameters are used may differ in MEMVIS.

Figure ?? shows the hierarchy of objects in MEMVIS with The MEMVIS objects have exactly the same organisation as the standard objects for dealing with the standard (count-based) modulation patterns. The objects `hsi_modul_pattern` and `hsi_modul_profile` have corresponding MEMVIS counterparts `hsi_vismod_pattern` and `hsi_vismod_profile`. The `hsi_bproj` object also has an analogous MEMVIS object `hsi_vismod_bproj`. When data is requested from these objects, they will call the following routines: `hsi_memvis_makepat`, `hsi_memvis_doprofile` and `hsi_memvis_dobproj`.

All MEMVIS routines are in `ssw/hessi/idl/image/memvis`. Also in that directory are several demonstration routines: `hsi_memvis_demo_masuda`, `hsi_memvis_demo_bignsmall`, `hsi_memvis_demo_default` and `hsi_memvis_demo_pntsrc_test`. These can be run from command line using `.r` , e.g.

```
SSWIDL> .r hsi_memvis_demo_masuda
```

If they don't work let me know immediately (a.j.conway@open.ac.uk).

7 Future Work

The following items will be investigated:

1. Small offsets are present in the visibilities, these may be due to an error in shifting to correct for the relative orientation between different detectors' grids.

2. The systematic error in constructing the observed visibilities from the observed counts needs to be measured, at present it is masked by the small offsets.

3. Several types of source structure (e.g. those in the demo) should be examined to see how accurately the algorithm is dividing the photons between several sources in the same image.

4. It would be desirable to fit the total flux as is done in MEM SATO.

5. Several improvements in speed are possible, e.g. `modpadskip` as is used in the annsec modulation patterns. Also, symmetry in the real/imaginary parts and between the two halfs of the rotation are currently not exploited.

6. The algorithm seems more robust to low photons counts than I would expect. I believe this is due to the effective smoothing that takes place in constructing the visibilities from the counts. Further investigation of why this is so is needed, along with the systematic error introduced and the effect on the final image.

7. Investigate effect off changing various iteration parameters.