AT Memoranda Series
AT/10.1/032

ON THE USE OF REDUNDANT SPACINGS

Jan Noordam - 23 August 1983

The remarks of Cornwell/Ekers on the pros and cons of redundant spacings (AT/10.1/031) have made it abundantly clear that we should get together to compare our diverging views.

In the first place I would like to state that in my view SELFCAL is a method to eliminate telescope-based errors with the help of a model of the brightness distribution. After producing "perfect" data in this way, we still have to reconstruct the image by means of some deconvolution scheme, but that is a separate problem.

C/E seem to consider the SELFCAL/CLEAN combination as a single process, thereby assuming that CLEAN is the best and/or only way to generate the model. This need not be the case at all, and indeed it may be undesirable to use CLEAN for this purpose in many cases.

The inevitable compromise between map size and points per beam constitutes a straightjacket that limits the dynamic range. The very advantage of automatic operation turns into a severe disadvantage if it gets stuck on the wrong solution (convergence is no guarantee for truth!). I have recently proposed an interactive system that takes care of most of these problems and makes use of the considerable processing power of the eye-brain system. A very important aspect is that the user gets some feeling for the range of possible solutions.

The only requirement on the model is, that it forces the data to be perfect within the noise. We can either achieve this by generating a perfect model (which CLEAN unsuccessfully tries to do) or by reducing the degree of freedom in the data as much as possible, so that simple models will approximate the desired effect.

In SELFCAL (my definition), the number of independent variables and thus the degree of freedom is reduced by assuming that all errors are telescope-based. This leaves 2N independent phase and gain errors per scan (=time-slot), rather than N(N-1): One may define a "reduction-factor" here as the ratio between the number of independent errors and the number of available data. In this case this number would be (N-1)/2, reflecting the superiority of more telescopes.

Redundant spacings provide a model-independent relation between telescope-errors, reducing the number of independent errors per scan to a minimum of one gain error (absolute amplitude) and one phase error (absolute phase slope over the array). The reduction factor in this case should be defined as the number of independent errors divided by the number of different (non-redundant) spacings, which makes the number 38 for the WSRT. This should be compared to 6½ for the WSRT without redundancy, and 13 for the VLA.

Even though the reduction factor with redundant spacings is considerably higher, SELFCAL (or other means) must still be used to determine the remaining variables. Thus redundant spacings only help to make SELFCAL more reliable by reducing its freedom to converge to the wrong solution. I do not understand what C/E think they do.

In the case where the observed field is dominated by strong compact sources, the reduction factor is not of crucial importance. Much more important is the intimate link between short and long baselines: It is relatively easy to generate an accurate model for the long baselines, that only "see" the compact sources, and by emphasizing the fit between data and model there, the short baseline interferometers are automatically corrected if all telescopes participate in both kinds of interferometers, or if the link is provided by redundant spacings.

I think that this aspect is responsible for the success of SELFCAL. It also takes care of the "subtle" point raised by C/E, that an excess of sample points in one region of the uv-plane (long baselines) is used to bypass a deficiency elsewhere (short baselines). But it only helps to correct the available data, not to transfer information to the short-baseline region! And it has nothing to do with redundant spacings.

It seems reasonable to expect that a high reduction factor will becomeprogressively more important as the higher dynamic range is required, although it is difficult to quantify this. The absence (not only stability) of interferometer-based errors or "closure-errors" is vital in this case.

For extended sources without compact sources for reference, a high reduction factor will also be an advantage, but it is not very clear how it compares with more interferometers or a more thorough sampling of the short-baseline region. The power of redundant baselines however, is by no means limited to improving the reliability of SELFCAL. It opens the way to model-independent techniques like the Centroid method:

The WSRT is a one-dimensional array, regularly spaced. After redundancy-calibration and in the absence of interferometer-based errors (including noise), we have a set of near-perfect one-dimensional scans of 38 regularly spaced uv-data each. By CLEANing these scans one-dimensionally, followed by a polynomial fit, we may make an accurate estimate of the zero-spacing, thereby determining the total flux and position in the sky of the source-centroid. Since these values should be the same for each scan, this provides us with a means other than SELFCAL to determine the two missing parameters per scan. Especially in the case where model-generation becomes extremely doubtful by any means (Crab, Cas A), such a model-independent scheme is very valuable.

Possibly the most important future use of perfect scans is the detection and interpretation of variability.

Finally I should point out, that the extra (model-independent) information provided by redundant spacings could be crucial for very accurate frequency- and polarization measurements. Although no specific schemes can be indicated now, the door should not be closed to this possibility.

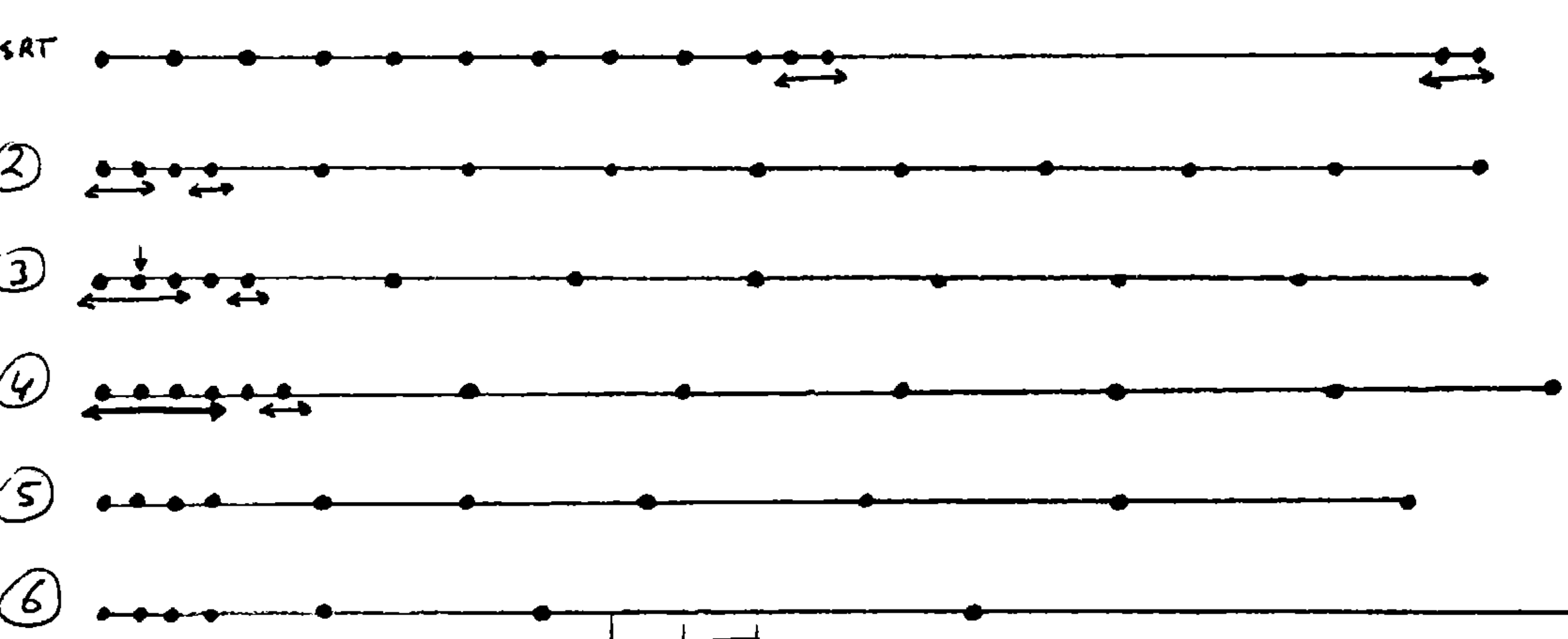
Redundant Arrays

An efficiency factor may be defined for a redundant array as the number of different (non-redundant) interferometers divided by the total number available. For an equidistant array of N telescopes, this factor is 2/N and for the WSRT in its most redundant configuration, it is 0.42.

But arrays with efficiency factors as high as 0.8 can be worked out, that still allow both a gain and a phase solution and provide regular coverage of the uv-plane. Even higher values can be obtained if one compromises for some properties of the array. It should be realised that even if only a subset of the telescopes can be linked by redundant spacings, this already constitutes a sizeable reduction factor.

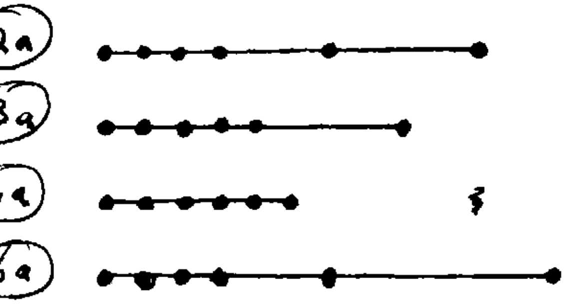
The conclusion must be, that redundant arrays are far less expensive in telescopes as is popularly believed, and that by paying this small price, an enormous increase in the potential power of the array may be bought - even for only six telescopes. The possibility of using redundant spacings should not be ruled out by selection of stations and/or design of the correlator.

Some redundant arrays



Telescopes	Interferom. (total)	efficiency	reduction	Remarks
14	91	0.42	38	big gap
13	78	0.49	38	Neither gain nor phase of shifts
12	66	0.58	38	No gain solution of shifted
12	66	0.60	40	
10	45	0.77	36	1st missing 16
8	28	0.75		Nothing

6 telescopes:



eff	red
0.67	10
0.53	8
0.33	5
0.67	10

efficacy factor = Nr of nonredundant / total ifrs

reduction factor = Nr of nonredundant ifrs

AT/10.5/001

COULD YOU PLEASE NOTE THE FILE NO. ON THIS PREVIOUS DOCUMENT. IT WAS SENT WITH THE NO. AT/10.4/001 ON IT, BUT SHOULD BE AT/10.5/001.

Software for the Australia Telescope Project

R. D. Ekers 21 March 1983

Although it has long been the practice in radio astronomy to take advantage of hardware developments in other radio astronomy laboratories and from industry this is still not generally true for software development, even though software costs may exceed computer hardware costs. Two factors which contribute to this situation are: i. Each telescope is a little different and since software always seems to be very flexible there is a tendency to make a customized product and ii. It is both possible and good fun to develop a new software system. Each new system will have some improvements and will have some new innovative features. Most good software architects will wish to construct a new system even though the development of such a system may be neither cost effective nor expedient. In the following comments I have tried to indicate what existing software would be available for the various stages of the AT project. Some of the salient features of existing systems are summarized in enclosed table.

There are three fairly distinct areas of software development needed for the AT.

1. The on-line system

This handles the interface to the array hardware, collects data and monitor information and allows control of the array. The on-line software will be quite specific to the AT. It may borrow algorithms from other systems (eg. WSRT, VLA) but much of the system will have to be designed for AT hardware and opertional requirements. This is a real-time processing enviroment and the software and hardware architecture will be different compared with the rest of the system. It may be useful to set up a separate software group for this area because the different programming style in this enviroment.

2. Calibration and mapping system

These operations are specific to a radio sysnthesis telescope and only software systems developed for synthesis telescopes could be easily used. There is no single system in use which could do all this. The VLA calibration software is unsuitable since it is written in SAIL, an ALGOL type language, which is no longer adequately supported. The WSRT calibration software is correlator based and contains WSRT specific features.

After the visibility function has been edited and calibrated the NRAO AIPS system could be used. It has a large amount of well tested software to do all the basic synthesis telescope tasks, and has the advantage of being a well debugged and an exportable system. It has good quality control, does not assume VLA specific features, and is CPU independent. AIPS currently operates on a large number of VAX system under VMS, a few UNIX systems and on two MODCOMP cpu's. Its major disadvantage is its unsuitably for use as a programming enviroment and its intensive use of two non-standard peripherals; the FPS array processor and the I^2S image display. Of these only the array processor is critical for the main sysnthesis processiong tasks. Although a non-AP version is available it is not optimized and would cause an unacceptable loss in efficiency.