PART III

PROCESSING TECHNIQUES AND DISPLAY METHODS

https://doi.org/10.1017/S0252921100074807 Published online by Cambridge University Press

DIGITAL PROCESSING METHODS FOR APERTURE SYNTHESIS OBSERVATIONS

(Invited paper)

B. G. Clark
National Radio Astronomy Observatory. Operated by Associated Universities, Inc. under contract with the National Science Foundation.

1. INTRODUCTION

The rapid growth, over the last two decades, of the amount of digital arithmetic capability available for a given cost is obvious to all, and is apparently not yet at an end. Therefore, any discussion of the best equipment and even of the best algorithms with which to attack a given problem will soon become obsolete as decreasing costs bring things previously considered inconceivable within economic feasibility. The general trend is that, as the cost of hardware decreases, the discussions of algorithms and procedures becomes simpler, as fewer approximations have to be made, and the mathematically simple correct formulae may be directly implemented.

This paper discusses two disjoint subjects. The first is simply a discussion, much limited by the state of the art of computer construction, of the least costly way of doing Fourier transforms for producing images from correlation data. The answer of course depends on the size and other particulars of the problem. The second is a very brief discussion of the relation of the map made by means of the Fourier transform to the sky brightness.

2. FAST FOURIER TRANSFORMS WITH GENERAL PURPOSE COMPUTERS

The discussion in this section is predicated on the use of the algorithm of Cooley and Tukey (1965). Occasionally, this is not the most appropriate algorithm; the discussion of this point is found in the following section.

The cost of doing a Fourier transform is comprised of two elements: arithmetic elements and memory. They are essentially non-interacting, and can be discussed separately.

In evaluating the cost of doing arithmetic operations, it is conventional to make a distinction between several classes of computers:

C. van Schooneveld (ed.), Image Formation from Coherence Functions in Astronomy, 113-120.

microprocessor systems, minicomputers, "main frames" and array processors. The distinction between the various classes is somewhat a matter of opinion; they are reviewed here. Perhaps the most useful distinction between "microprocessor systems" and "minicomputer systems" is that modern mini's have hardware floating point arithmetic units (at least optionally) and microprocessors do not. The most common (and most primitive) microprocessors, e.g., Intel 8080, Zilog Z80, National SC MP, do not even have hardware multiply.

The distinction between minicomputers and the class of device best given the computer jargon term "main frame" is even fuzzier. The traditional division is that a minicomputer is built around a 16, 18 or 24 bit word length, or is an upward compatible member of a family of 16 bit word length computers. Main frames were conceived and planned around a longer word. The distinction is not useful in comparing the arithmetic capabilities needed for the FFT; some main frames are many times slower than some minicomputers. The chief practical distinction between main frames and minicomputers is that the former are provided by the manufacturer with vastly superior software systems.

The term "array processor" has accumulated a meaning not adhering to it four or five years ago. The most successful devices now sold under this name are not, like the ones marketed half a decade ago, devices equipped to perform a few simple operations on linear arrays of numbers. They are instead general purpose minicomputers, with all accouterments not directed to fast floating point arithmetic stripped, to hold down costs, and the hardware optimized to do floating point arithmetic as fast as practicable. The manufacturers of these devices also hold down costs by providing, chiefly, a library of arithmetic routines, and only a minimum of resource management, device handler, and other expensive system software. The array processors do not currently support higher level languages (except, perhaps, as a device for calling low-level-coded subroutines), and the low-level languages require much care and sophistication to utilize the full capabilities of the device. This situation may be expected to change, and this will make a great improvement in their usability.

The times and system costs for doing Fourier transforms in systems of various classes is given in Table I. The reader is warned that this table is assembled from highly heterogenous sources, and reflects a variety of implementating, done by different people in different languages. The numbers are indicatory only, and cannot be taken seriously to a factor of 2 or so. All numbers quoted are Fortran programs, though by different authors with different details, except those for the 8080, the Cray I, and the array processors.

Memory costs are also rapidly decreasing. However, for large images it is still worthwhile to implement a hierarchy of memories. For making a large set of million point images one might have 16 K words of array processor memory (access time 300 ns, cost $20,000 per million bits), 1 million words of minicomputer storage (function:

TABLE I

TIMES FOR 1024 COMPLEX FFT

DEVICE	TYPICAL SYSTEM COST	TIME, SECONDS
Microprocessors		
Intel 8080	$ 4,000	27
Minicomputers		
Sperry-Univac 77-400 (no floating point hardware)	20,000	16
Modcomp II	50,000	1.6
DEC-PDP-11/70	100,000	0.8
Main Frame		
IBM 360/50	700,000	2.6
DEC KI/10	1,000,000	1.1
IBM 360/65	1,500,000	0.8
IBM 370/155	2,000,000	0.3
TI ASC/P2	4,000,000	0.004
Cray I	8,000,000	0.001
Array Processors		
FPS AP120B	150,000*	.005
CSPI MAP-300	150,000*	.006

*including host

transposing rows and columns, access time 1 μs, cost $12,000 per million bits), 200 million words of disk (working store, buffer to output; access time 25 ms, cost $3 per million bits) and an indefinite number of magnetic tapes (access 5 minutes, cost $0.25 per million bits).

3. FAST FOURIER TRANSFORMS WITH SPECIAL PURPOSE DIGITAL HARDWARE

Opening the door to consideration of special purpose devices introduces more options into questions of design than merely choice of the particular chip. For general purpose machines, for any really massive aperture synthesis, the only reasonable choice is to build the procedure around a Fourier transform using the Cooley-Tukey algorithm. Anything else is either grossly inefficient use of the equipment or is too difficult to organize. In special purpose equipment, efficiency is

https://doi.org/10.1017/S0252921100074807 Published online by Cambridge University Press

not particularly a consideration. It must be fast enough to do its hardest job, but it doesn't matter if for easier jobs it does them efficiently or not; being a special purpose device there is probably nothing useful it could be doing in the time saved by doing easy jobs efficiently.

For some special cases, the use of the Cooley-Tukey algorithm is not the most efficient solution either. Some corrections for instrumental problems can conveniently be done by modifying the nature of the transform relationship, which may involve a much slower algorithm than that of Cooley and Tukey, but which may still be feasible for a special device. Corrections which may be done this way include the non-coplanar baseline effect and the correction for instrumental bandwidth, both mentioned later. If one is interested in making, in real time, a map that is not too large with input data rates not too high, one can build a classical discrete Fourier transform device to do it. The expenditure of a few thousand dollars on arithmetic elements will result in a capability approaching 10^9 multiplies per second. If one is constructing an output image of, say, 10^6 points, one could support an input data rate of the order 10^3 correlation points per second. The possibilities have been explored extensively by Frater (1978). The limitation of this method for real-time use is that one is often interested in collecting data for several images at the same time. The memory for several images of this size is costly if implemented as semiconductor store, and inconvenient if implemented as a rotating magnetic memory. The approach is less attractive for non-real-time use because, once the data have been introduced into a general purpose computer system for management and recording, the question of efficiency again raises its head, and one has hopes of saving on the cost of the device by using the most efficient algorithms possible, and using the time saved on easy problems to catch up on work fallen behind during hard problems.

In cases where the machine can be built to do one job only and need not, for instance, do transforms of different lengths, Winograd's algorithm (Winograd, 1978) may be superior to that of Cooley and Tukey. It involves many fewer multiplications and a similar or smaller number of additions. It does seem to require about one additional bit of precision (Patterson and McClellan, 1978), and the structure of the algorithm does not lend itself to the beautiful symmetry of the Cooley-Tukey algorithm; therefore the wiring to accomplish the transform in hardware will be a much larger chore to design. This last consideration is the primary reason why the algorithm is not rapidly displacing the Cooley-Tukey algorithm in general purpose computers - the overhead required may eat up most or all of the time saved by the fewer multiplies, and the result is an opaque program that offers only small speed increases over the Cooley-Tukey algorithm.

Even in special purpose hardware implementations, the Cooley-Tukey algorithm is not unattractive. Using only a few chips of LSI, with parallel multipliers, it is possible to build a hardware "butterfly" (a

butterfly on a single chip is being designed by TRW, Inc.) which should result in a transform speed of about .1 μs times $\log_2 N$ if they are provided in a parallelism of N and of 1 μs if in a parallelism of $N\log_2 N$, for a transform of length N. With these extremely large data rates all the difficult problems fall in storing, managing, and assimilations of the data. Arithmetic operations have become the easier part of the job.

Parallel multipliers grow in complexity as the square of the datum length, so the Winograd algorithm increases in attractiveness for higher precision operations.

The usual rule of thumb is that one loses half a bit of precision, in the Cooley-Tukey algorithm, per butterfly stage. In floating point one should carry mantissa lengths equal to the desired output accuracy plus $\frac{1}{2}\log_2 N$, where N is the image size. For fixed point operations, the convention is to represent the input data to its inherent accuracy, and then add a bit on the left every butterfly stage and drop a bit on the right alternate butterfly stages. In most, but not quite all, images, this ends up with many empty bits on the left, but simplifies the management greatly.

4. SOME THEORETICAL PROBLEMS IN FOURIER TRANSFORM ANALYSIS

Although the most exciting areas for theoretical research lie with the non-FFT algorithms of image formation, which are the topic of most of the discussion at this symposium, there are questions in the simple case of Fourier transforming which are not without interest.

Perhaps the simplest of these is the correction for the non-coplanar baseline effect. For earth rotation synthesis with non-east-west baselines the loci of the antennas as the earth rotates do not describe a plane in space, but occupy a solid. The relation between the sky brightness and the measured correlation functions involves all three direction cosines of the brightness element, and therefore the image is not recoverable by a two-dimensional transform. For an interferometer element located at (u,v,w) relative to a reference element (u and v conventionally lie in a plane perpendicular to the "phase tracking center"), and a brightness element at (s_x, s_y, s_z) the phase is $us_x + vs_y + (s_z - 1)w$. The -1 in the last term arises because the phase tracking center $\{s = (0, 0, 1)\}$ has been removed from the calculation. Since s is a unit vector, $s_z - 1 \approx -\frac{1}{2}(s_x^2 + s_y^2)$. It is easy to verify that this is an important effect for many practical cases. Its relative importance is measured (roughly) by the product of the size of the synthesized image in radians and the size of the synthesized image in beamwidths. If this product approaches one the effect must be corrected.

There are three practicable cures for this problem. One may deal only with small areas at a time, so that the eventual synthesized image

https://doi.org/10.1017/S0252921100074807 Published online by Cambridge University Press

is generated from a mosaic of images with differing phase tracking centers. One may make maps at frequent intervals, short enough that the rotation of the earth does not destroy the two-dimensional nature of the array. Or one may make a three-dimensional transform of the correlation data, and sample it along the spherical surface

$$s_x^2 + s_y^2 + s_z^2 = 1$$

which automatically aligns the phases. This last solution appears to be the most attractive computationally. It does raise the interesting question of how many samples are needed in the third dimension. The answer is roughly the number of samples needed in the other dimensions times the image size in radians. However, the effects of sampling are well understood only for a uniform random distribution of data within the sampling intervals, but in this case there are usually sufficiently few points called for that the distribution of points within the cells is far from uniform, and will vary greatly from place to place. A good calculation of this effect for practical cases apparently has not been done.

A second interesting problem is the breakdown of the quasi-monochromatic assumption. The usual effect of different frequencies is simply a scale change of the image with frequency. The data at various frequencies within the band of reception are, however, added together before the image is produced, and the net effect is a radial smearing that becomes more serious at the edges of the image. This smearing effect may be reduced to a simple convolution by merely reinterpolating the image onto polar coordinates logrithmic in the radius. The full power of conventional convolution theory is then available for correcting for the effect.

Perhaps the most interesting problem in conventional image formation is the question of weighting to compensate for incompleteness. To see how the problem arises, suppose we have a randomly located collection of samples of the correlation function plane. If we take a very coarse grid, and average the points within each cell (or convolve in some other fashion and resample at the cell center), we will have a nearly complete sampling of the plane, and the synthesized beam (after a suitable taper is applied) is well behaved, has low side lobes, and causes a minimum amount of trouble. If we try to do the same thing on a very fine grid, however, we shall find that we have only one sample within each grid cell, and our synthesized beam will have side lobes at the mercy of accidental or systematic fluctuations in point density about the plane. In some sense this is fundamental; a side lobe of the synthesized beam, taken in this manner, reflects a real difficulty in distinguishing between radiation coming from two locations separated by the separation of beam and side lobe. On the other hand, the difficulty must be to some extent apparent, since these side lobes due to uneven sampling vanish when one interpolates to a coarse grid. The difference lies in the assumption, implicit in resampling on a coarse grid, that the radiation comes from a restricted area of the sky. For many purposes,

one desires to have a large image produced under the assumption that the radiation comes from one or a few relatively isolated regions. This is achieved, in a very rough approximation, by making the fine grid map with the weighting of each sample set to that appropriate for the coarse grid map. The implication is that, for each emitting region, a tight cluster of points in the correlation function plane carry much the same information, and therefore are not weighted as heavily as isolated points. A more elegant approach would be to convolve the unit weights of the points with a suitable smooth function, and to weight the individual points with the reciprocal of the convolved weighting function at their location. This has the effect of suppressing the side lobes of the synthesized beam within an area whose size is the reciprocal of that of the convolving function.

A yet more elegant procedure is to choose weights which cause the synthesized beam to approach most closely a desired beam (say a Gaussian) within a specified area, or approach most closely with errors weighted by a grading function of distance from synthesized beam center. One can't carry this approach too far; for instance it is easy to show that if one asks to have the least squares fit of the synthesized beam to a Gaussian over the whole image, the answer is to simply Gaussian weight the sample points.

5. CONCLUSION

The computation of a Fourier inversion of the correlation data is nearly always the first step in any formal image formation procedure, and is part of an iterative step in many of them. For this reason, and because the transform map is an easily produced case of something that is at least consistent with the data, the Fourier transform technique will never be without interest. The facilities for producing transforms have dramatically dropped in cost, so that the limitations of the devices are more strongly met in providing or organizing storage for the output than in arithmetic elements. With the possibility of building, at attainable costs and complexities, devices which can produce a Fourier transformed output approaching 10^9 numbers per second, we may say that the arithmetic of the Fourier transform is not a conceptual difficulty for image formation.

REFERENCES

Cooley, J.W. and Tukey, J.W.: 1965, "Math. of Computation" 19, 297.

Frater, R.: 1978, "Astron. and Astrophys.", in press.

Patterson, R.W. and McClellan, J.H.: 1978, "IEEE Trans. on Acoustics, Speech and Signal Processing", in press.

Winograd, S.: 1978, "Math. of Computation" 32, 175.

DISCUSSION

Comment A.T. MOFFET

The problem of bandwidth smearing may be a red herring. In cases where $\Delta\nu/\nu$ is appreciable it is usually necessary to split the data into a number of narrow-band channels. Otherwise the "delay beamwidth" would be smaller than the primary beamwidth; in many cases this multi-channel processing is also necessary in order to selectively reject interference. Westerbork and the VLA are both implementing this scheme, I believe.

Reply B.G. CLARK

Even if you split the band into pieces, this correction can be important within each piece.

Comment R.H. HARTEN

The bandwidth problem is noticeable even in the present Westerbork data. Comparisons of source fluxes at several frequencies require accurate fluxes and source sizes. The large number of sources per field and the large number of fields makes detailed corrections prohibitive. Thus some care must be taken to restrict the bandwidth or the field of view, to minimize the source smearing.

Comment P. DEWDNEY

In relation to the previous remarks I can cite an extreme example of a large field aperture synthesis telescope. I have a 22 MHz instrument which covers a field 40 degrees wide with 15 arc minute beam in which these radial bandwidth distortions are apparent if not corrected for. However, if fan beam analysis is used, it is possible to produce a circular beam on all positions of the map.

Comment T.W. COLE

The suggested procedure of converting three dimensional data back into two by Fourier transforming the sky brightness, sampled along the sky sphere, back to the U,V plane and resampling it at the original U,V points, to obtain "corrected" data, needs classification. One is likely to run into trouble if one tries to use these data for other algorithms; for clean, for instance, since the true beam shape is still varying with position in the field.

Reply B.G. CLARK

Yes, the effects occur, and the data differ from that that would have been obtained had the geometry been such as to allow the direct sampling of the points in question, but it is not clear that these effects will be a practical limit to a reconstruction process.