NASA CR- 166793

(NASA-CR-166793) MULTIFREQUENCY APERTURE-SYNTHESIZING MICROWAVE RADIOMETER SYSTEM (MFASMR). VOLUME 2: APPENDIX (Hughes Aircraft Co.) 168 p HC A08/MF A01 CSCL 14B G3/35

N82-25486

Unclas 21919

HUGHES

HUGHES AIRCRAFT COMPANY

1 JULY 1981

MULTIFREQUENCY APERTURE-SYNTHESIZING MICROWAVE RADIOMETER SYSTEM (MFASMR)

APPENDIX

HUGHES PROPRIETARY CONTRACT NAS 5-25755

HUGHES
HUGHES AIRCRAFT COMPANY
SPACE AND COMMUNICATIONS GROUP

VOLUME II

SCG 810345

CONTENTS

APPENDIX A

FELLGETT'S ADVANTAGE (MULTIPLEX ADVANTAGE)

APPENDIX A

FELLGETT'S ADVANTAGE (MULTIPLEX ADVANTAGE)

A.1 THEORY

Fellgett's Advantage is the name of a factor which shows the improvement obtained when the spectrum of thermal radiation is measured by a Fourier transform spectrometer, rather than by a scanning filer spectrometer.

This difference in performance only occurs when there is no amplifier ahead of the spectrometer. Then Fellgett's Advantage is a measure of spectrometer performance when all of the input power is applied to a single detector in the autocorrelator compared to the performance obtained when a narrow, bandpass filter scans the spectrum. Such a filter wastes the spectrum power outside the filter passband.

Fellgett's Advantage is found as follows. Let

B = total system bandwidth

T_{SYS} = system noise temperature (A-1)

$N = k T_{SYS} B$ = noise power (A-2)

T = total observation time

M = number of frequency bins resolved

Then in the case of the scanning filter

b = B/M = radiometer bandwidth using a scanning filter

τ = T/M = time available to measure temperature of each frequency bin

So

$$\Delta T_{FILTER} = \frac{K\,T_{SYS}}{\sqrt{\left(\frac{B}{M}\right)\left(\frac{T}{M}\right)}} = \frac{MKT_{SYS}}{\sqrt{B\tau}} = \text{temperature resolution} \qquad (A\text{-}3)$$

In the correlator case

$b = B$

$\tau = \frac{T}{M}$ = time available for measurement of each correlation sample. Since $\phi(\tau)$ is complex, a real and an imaginary sample are taken. The spectrum generated by Fourier transformation has the same number of samples. However, the samples are real. Therefore, the variance of spectrum bin is the sum of the variances of the real and imaginary samples.

$$\Delta T_{CORR} = \frac{\sqrt{2}\,K\,T_{SYS}}{\sqrt{B\left(\frac{T}{M}\right)}} = \frac{\sqrt{2M}\,KT_{SYS}}{\sqrt{BT}} \qquad (A\text{-}4)$$

Fellgett's Advantage, F, relates T_{FILTER} and T_{COR} as follows

$$\Delta T_{CORR} = \sqrt{\frac{2}{M}}\,T_{FILTER} \qquad (A\text{-}5)$$

Equation (A-5) states that if we cannot increase the power at our disposal by amplification, the noise level of a scanning filter spectrometer is greater than that of a Fourier spectrometer.

However, if amplification is available, power is available to feed a channelizing filter bank or a delay line with a set of taps. These devices will have a temperature resolution which does not depend on M.

With no restriction on the number of taps, we can measure $\phi(\tau)$ for both positive and negative values of τ. Now $\phi(\tau)_{REAL}$ is symmetric and $\phi(\tau)_{imaginary}$ is antisymmetric. So the corresponding positive and negative real correlation samples, $\phi(\pm\tau_1)$ can be added together. The imaginary samples paired with the real samples can be added after reversal of the sign of one sample. In both

cases the noise adds noncoherently and the signal coherently. This reduces the temperature resolution by $2^{-1/2}$. This compensates for the 2 increase in the Fourier spectrometer case caused by the fact that $\phi(\tau)$ is complex, but the power-density spectrum is real.

The $\tau = 0$ tap is a special case. The $\phi(0)_{\text{imaginary}}$ sample must be zero. So it is not used in computing the transform. This produces a two-to-one noise reduction consistent with the noise reduction obtained with the pairs of taps where the delays are $\pm\tau_1$ ($\tau \neq 0$).

A.2 PERFORMANCE CALCULATIONS

As examples of the relative performance of these devices let us consider three correlator/spectrometers operating between the water vapor line and the O_2 line complex, say 23-53 GHz.

One device will be a crystal video system using a Hughes Electron Dynamics GaAs diode, a second system uses an evacuated barretter, a third system uses a TRG mixer and a Narda GaAs FET IF amplifier. The amplifier has an 8-18 GHz passband.

This amplifier has a noise figure of about 9 db according to the NARDA ads.(1) The TRG mixer appears to have a single sideband noise figure of about 6 db.(2) This superhet must operate in a single sideband mode. The double sideband mode would put a 16 GHz hole in the middle of the reception band. The total noise figure is then about 15 db.

A TSS of 6×10^{-11} watts will be used for the evacuated barretter.(3)

Figure A-1 shows that a TSS of -55 dbm or 3.16×10^{-9} watts is a good number for the diode performance.

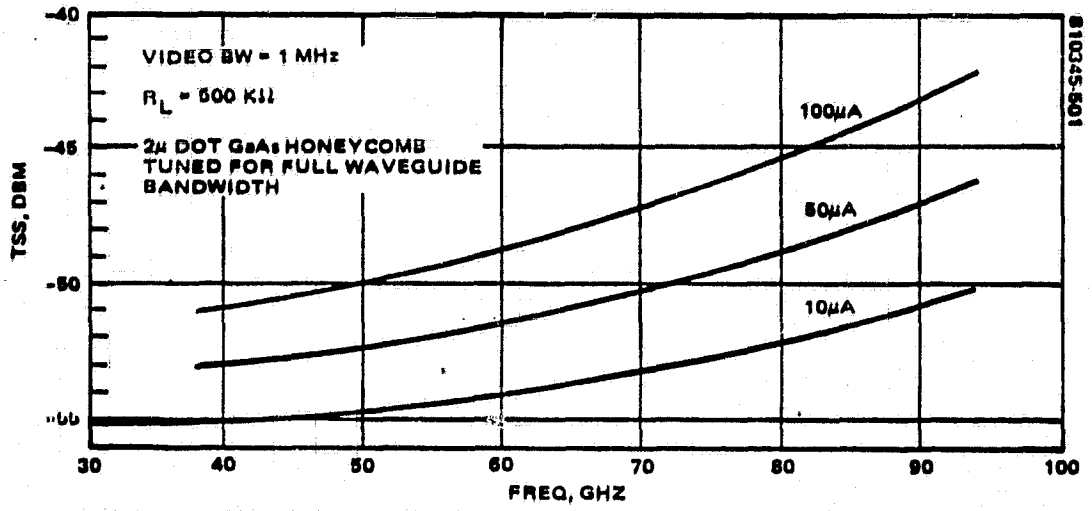
Performance calculations for the three cases follow. A radiometric performance parameter of K = 1, and an integration time of one second will be used in all three cases. The result is a representative temperature resolution which is useful for comparison purposes. It also will be assumed that 1024 frequency bins are desired. It then takes 17.07 minutes to measure the spectrum.

---

(1) Microwaves, p. 7, April 1980.

(2) A. Cardiasmenos, "Planar Devices Make Production Practical," Microwave Systems News, April 1979, pp. 15-21.

(3) Long and Butterworth, "New Technique for Microwave Radiometry," IEEE TRANS MTT, Sept 1963, pp. 389-396.



NOTE: BIAS CURRENT IS A PARAMETER. TSS IS 4 dB TO 6 dB BETTER WHEN TUNED FOR NARROW BAND RESPONSE (< 10 GHz).

FIGURE A-1. TSS VS FREQUENCY

SUPERHET

$\overline{BW}$ = 16 GHz

NF = 15 db

$T_{SYS} = 10^{1.5} \times 290°K = 9170.61°K$

FILTER BANK OR TAPPED DELAY LINE $\Delta T' = \frac{T_{SYS}}{\sqrt{B\tau}} = 0.070°K$

SCANNING DELAY $\Delta T = \sqrt{2048} \times \Delta T' = 3.28°K$

SCANNING FILTER $\Delta T = 1024 \times \Delta T' = 71.68°K$

BARRETTER

$T_{SYS} = \frac{6 \times 10^{-11} \text{ W}}{1.37 \times 10^{-23} \times 2 \times 10^{10}} = 218.98°K$, $\Delta T' = 0.00173°K$

FILTER BANK OR TAPPED LINE NOT POSSIBLE

SCANNING DELAY $\Delta T = \sqrt{2048} \times 1.73 \times 10^{-3} = 0.08°K$

SCANNING FILTER $\Delta T = 1024 \times 1.73 \times 10^{-3} = 1.77°K$

CRYSTAL VIDEO

$$T_{SYS} = \frac{3.16 \times 10^{-9} \text{ W}}{1.37 \times 10^{-23} \times 2 \times 10^{10}} = 11{,}533°K, \Delta T' = 0.0815°K$$

FILTER BANK OR TAPPED DELAY LINE NOT POSSIBLE

SCANNING DELAY $\Delta T = \sqrt{2048} \times 8.15 \times 10^{-2} = 3.69°K$

SCANNING FILTER $\Delta T = 1024 \times 8.15 \times 10^{-2} = 83.46°K$

It may be seen that the scanning filter mechanization is a very poor performer. However, the Fourier transform system with a scanning delay is a reasonable system mechanization, if simplicity and cost are important. In cases where the ultimate in performance is not needed, it is a viable candidate.

APPENDIX B

GENERALIZED INTERFEROMETER MAPPING BEHAVIOR

APPENDIX B

GENERALIZED INTERFEROMETER MAPPING BEHAVIOR

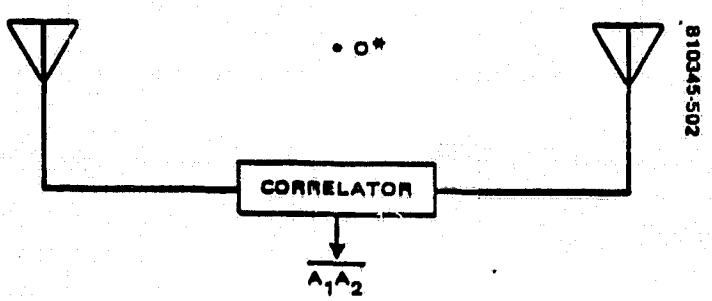
The basic properties of correlating interferometers as mappers should be addressed. The fact that the system is linear in its response to power or intensity will be used in several key derivations. To show under what conditions such linear response is obtained, a calculation of the mapping response of a correlation radiometer in the general case follows.

The interferometer system is indicated below.

Let F(x, t) be the electric field distribution across the source. Assume it to be a stationary random process. Truncate this process to form F(T, x, t) so that a function with a Fourier transform is produced. This field is complex and analytic. The second property arises because the real and imaginary components of F are assumed to be Hilbert transforms. This guarantees that F is a causal signal.

The Fourier transform of V(x, t) is

$$\hat{F}(T_1, X, \nu) = \int_{-\infty}^{\infty} F(T, X, t)\, e^{2\pi i \nu t} \qquad \text{(B-1)}$$



*O = INTERFEROMETER PATTERN ORIGIN

FIGURE B-1. CORRELATING INTERFEROMETER

where ν = frequency, t = time, X = source coordinate and hat symbol indicates transform function.

In order to characterize the antennas, let $G_1(x)$ and $G_2(x)$ be the amplitude diffraction patterns of A_1 and A_2. Both patterns are in a polar coordinate system with origin at O in Figure B-1. Then the output of the first antenna is

$$\hat{V}_1(T, x, \nu) = G_1(x, \nu) * \hat{V}(T, x, t)$$

Similarly

$$\hat{V}_2^*(T, x, \nu) = G_2^*(x, \nu) * \hat{V}^*(T, x, t) \qquad \text{(B-2)}$$

The product of these two signal produces the video signal output with the form

$$\hat{V}_1(T, x, \nu)\, \hat{V}_2^*(T, x, \nu) = \left[G_1(x_1, \nu)\, G_2^*(x_2, \nu)\right] ** \left[\hat{V}(T, x_1, \nu)\, \hat{V}^*(T, x_2, \nu)\right] \qquad \text{(B-3)}$$

where ** indicates a two-dimensional convolution (* = one-dimensional convolution).

Divide both sides of Equation (B-3) by 2T. Since G_1 and G_2 do not depend on T, the statistics are stationary. In the limit

$$\hat{P}(x, \nu) = \lim_{T \to \infty} \left[G_1(x_1, \nu)\, G_2(x_2, \nu)\right] ** \hat{\Gamma}(x_1, x_2, \nu) \qquad \text{(B-4)}$$

where $\hat{P}(x, \nu)$ is the complex spectral density of the output and $\hat{\Gamma}(x_1, x_2, \nu)$ is the mutual spectral density of the radiation in the plane of the antennas. (The Van Cittert-Zernike theorem describes how these coherence functions propagate from the object scene to the antenna plane.) By definition

$$\hat{\Gamma}(x_1, x_2, \nu) = \lim_{T \to \infty} \frac{\hat{V}(x_1, \nu)\, \hat{V}(x_2, \nu)}{2T} \qquad \text{(B-5)}$$

Equation (B-4) is the general solution for the interferometer response to an extended, partially coherent, polychromatic source.

For the quasi-monochromatic case, which holds for most practical systems, there are two conditions. These are

$$\Delta\nu \ll \bar{\nu}$$

$$|\tau| \ll \frac{1}{\Delta\nu} \quad \text{and} \quad \ell \ll \frac{c}{\Delta\nu} \tag{B-6}$$

That is, the spectral width, $\Delta\nu$, must be small compared to the mean frequency, $\bar{\nu}$; also all system path differences must be small compared to $c/\Delta\nu$.

For this case the mutual coherence function becomes

$$\Gamma(x_1, x_2, \tau) = \Gamma(x_1, x_2, 0)\, e^{-2\pi i \bar{\nu}\tau} \tag{B-7}$$

and the mutual power cross-spectral density is

$$\hat{\Gamma}(x_1, x_2, \tau) = \hat{\Gamma}(x_1, x_2, 0)\, \delta(\nu - \bar{\nu}) \tag{B-8}$$

Substituting (B-8) into (B-4) and evaluating at $\tau = 0$

$$P(x) = [G_1(x)\, G_2(x)] ** \Gamma(x_1, x_2, 0) \tag{B-9}$$

$G_1(x)$ and $G_2(x)$ have become antenna patterns of A_1 and A_2 at frequency $\bar{\nu}$. This is the solution for the output response to a partially coherent, quasi-monochromatic, extended source distribution.

Let us look at the limiting cases of coherent and noncoherent extended sources. For a coherent source

$$\Gamma(x_1, x_2, 0) = V(x_1)\, V^*(x_2)^{(1)} \tag{B-10}$$

---

(1) M. Beran and G. Parrent, Theory of Partial Coherence, Wiley & Sons, 1964, Sec. 4.3 and 4.4, pp. 54 and 57.

Then (B-9) becomes

$$P(x) = [G_1(x) * V(x_1)]\ [G_2(x_2) * V^*(x_2)] \qquad \text{(B-11)}$$

This equation shows that for mapping extended objects which are coherently illuminated (or radiate coherently across their surface) the system is essentially nonlinear. Unscrambling such a nonlinear signal to form an image appears to be a hopeless undertaking.

In the limit, where $V(x) = \delta(x - x_1)$ for a point source (B-11) reduces to

$$P(x) = e^2\ G_1(x)\ G_2(x) \qquad \text{(B-12)}$$

Thus, for point sources, the system sees a power diffraction pattern which is the product of the patterns of the two antennas. This appears intuitively to be correct.

For self-luminous sources which are essentially noncoherent

$$P_0(x_1, x_2, 0) = \bar{\beta}\ P_0(x)\ \delta(x_1 - x_2)^{(1)} \qquad \text{(B-13)}$$

The subscript "0" denotes that this is the distribution at the object source.

Substituting (B-13) in (B-9) and integrating over the source, we get

$$P(x) = \int_{\text{SOURCE}} \bar{\beta}\ P_0(x')\ [G_1(x - x')\ G_2^*(x - x')]\ dx' \qquad \text{(B-14)}$$

Where $P_0(x')$ is the distribution of power over the surface of the source. Equation (B-14) describes a system which is linear in power and which has a complex impulse response $K(x)$ given by

$$K(x) = G_1(x)\ G_2^*(x_2) \qquad \text{(B-15)}$$

Then (B-14) can be written

$$P(x) = \overline{\beta}\, P_0(x) * K(x) \tag{B-16}$$

Thus the system under consideration, although nonlinear for mapping extended coherent sources, is certainly linear for mapping extended incoherent sources such as the radio-sky or radiometric-earth.

Since G_1 and G_2 are defined relative to the origin 0, we can set the antenna phase centers at that point. Then

$$\begin{cases} G_1(x) = G_1'(x)\, e^{-i\xi x} \\ G_2(x) = G_2'(x)\, e^{+i\xi x} \end{cases} \tag{B-17}$$

where $G_1'(x)$ and $G_2'(x)$ are the radiation patterns of A_1 and A_2 relative to their own phase centers and ξ is the separation between these phase centers. Therefore, the impulse response can be written in the form

$$K = G_1'(x)\, e^{-i\xi x} + G_2'(x)\, e^{+i\xi x} \tag{B-18}$$

Let A_1 and A_2 have real, symmetric weighting functions, $A_1(\eta)$ and $A_2(\eta)$, where η is the antenna aperture coordinate. Then G_1 and G_2 will also be real and symmetric. For this special case,

$$K_{\text{symmetric}}(x) = 2G_1(x)\, G_2(x) \cos \xi x \tag{B-19}$$

In this special case the interferometer impulse response is simply the product of the amplitude patterns of the two antennas and the interference factor resulting from the phase center separation.

By the convolution theorem the Fourier transform of (B-16) is

$$\begin{array}{ccccc} P(x) & = & P_0(x) & * & K(x) \\ \Downarrow & & \Downarrow & & \Downarrow \\ S(F) & = & O(F) & \cdot & \overline{MTF}(F) \end{array} \tag{B-20}$$

where $O(F)$ and $S(F)$ are the complex spatial frequency spectrums of image and object. $\overline{MTF}(F)$ is the modulation transfer function of the interferometer.

Taking the Fourier transform of (B-15)

$$\begin{array}{ccccc} K(x) & = & G_1(x) & \cdot & G_2^*(x) \\ \Updownarrow & & \Updownarrow & & \Updownarrow \\ \overline{MTF}(F) & = & A_1(\eta) & * & A_2(\eta) \end{array} \qquad \text{(B-21)}$$

The second two transforms are the Fraunhoffer diffraction relation between aperture excitation and the resulting radiated field.

Bracewell gets rid of the characteristic folding inversion due to the convolution by using the mirror image of $K(x)$. Call it $\tilde{K}(x)$. Then the bottom relation in (B-21) becomes an autocorrelation. Thus

$$\overline{MTF}(F) = A_1(\eta) * A_2(\eta) \qquad \text{(B-22)}$$

where

$$\tilde{K}(x) = \tilde{G}_1(x) \cdot \tilde{G}_2(x) \qquad \text{(B-23)}$$

It should be obvious that $K(x)$ is the antenna power pattern of the interferometer.

APPENDIX C

MICROWAVE RADIOMETER IMAGE STATISTICS

APPENDIX C

MICROWAVE RADIOMETER IMAGE STATISTICS

C.1 SPECTRUM SELECTION USING RESULTS DUE TO MANDELBROT

Mandelbrot's computer generated Brownian landscapes on pp. 210-215 of his book show how changing and image parameter, H, changes the appearance of the terrain images. They show how the correct value of H is needed in order to produce the subjective impression that one is looking at an "image" and not noise. Unfortunately, these figures will not xerox. However, the computer generated Brownian trails shown below could be reproduced. These trails show the change in long range coherence due to changing the parameter, H.

Mandelbrot concludes that in order for his Brownian landscapes to have the properties of real landscapes the redundancy parameter, H, should have a value of 3/4. This corresponds to a spectrum of the form,

$$G(f) = sf^{-2H-1} = Sf^{-5/2}.$$

The transform-mate autocorrelation expression is given in Figure C-2 along with a plot of this function.

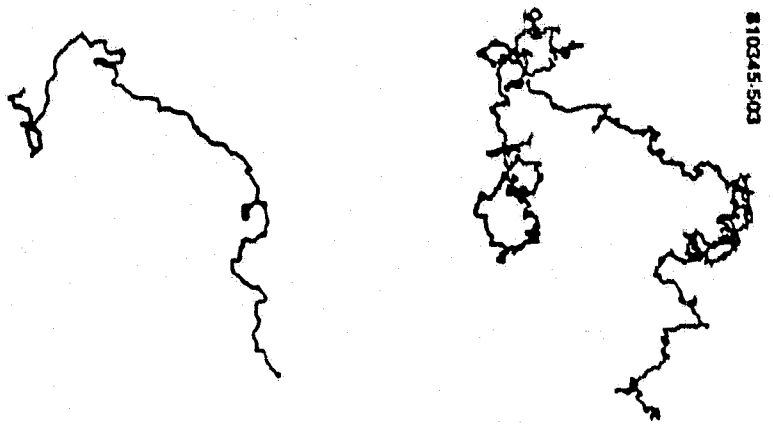
C.2 CALCULATION OF REDUNDANCY PARAMETER FROM IMAGE COMPRESSION DATA

In order to calculate the properties of the various imaging systems, we need to use a power spectrum for the radiometer images which has the right redundancy.

Assume that all images (visual, IR, or microwave) have about the same statistical properties, because they originate from the same object scene. Then we can estimate H from knowledge of properties of images in compressed format. Such compression is carried out in order to send images with reduced

---

(1) B. Mandelbrot, Fractals — Form, Chance and Dimension, W. Freeman & Co., 1977.



The Figure to the left constitutes an example of a statistically self similar fractal curve with the dimension D = 1/0.9000 = 1.1111. Its coordinate functions are independent fractional Brownian functions of exponent H = 0.9000, which implies that self intersections, though not strictly prohibited, are greatly discouraged by forcing the curve to be very persistent. Thinking of complicated curves as the superimpositions of large, medium, and small convolutions, it may be said that in the case of high persistence and dimension close to 1, small convolutions are barely visible.

The Figure to the right uses the same generating program and the same pseudo random seed but a dimension increased to D = 1/0.7000 = 1.4285. Without having to change the shape of the convolutions which add up to form this trail we have increased the relative importance of the small ones, and to a lesser extent, of the medium ones. Previously invisible details become very apparent. The general underlying shape, however, remains easily recognizable.

FIGURE C-1. FRACTIONAL BROWNIAN TRAILS
(DIMENSIONS D = 1.1111, D = 1.4285)

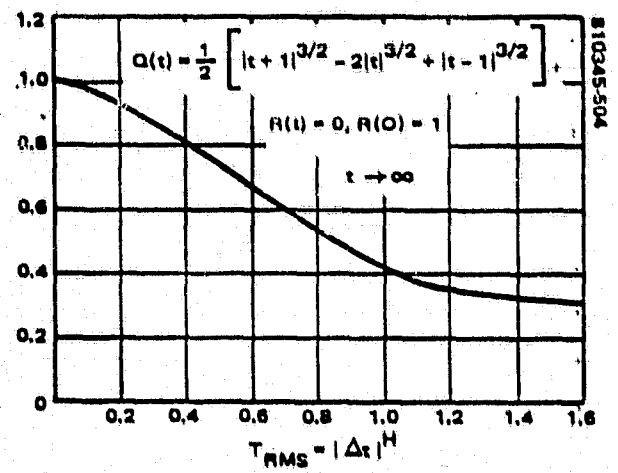


FIGURE C-2. AUTO CORRELATION FUNCTION OF GENERALIZED BROWNIAN MOTION FOR H = 3/4

bandwidth. Shannon's source coding theorem states that if the information density of the code is greater than the information density of the source, the source message can be sent without error. This is an existence theorem. In practice, all codes exhibit distortion which rises to high levels as the code density approaches the source density. This distortion level can be computed from rate distortion theory.

Now images can set sent with tolerable distortion at a code density corresponding to an information density of one bit per pixel.[2] Therefore, we will assume this image information density in calculating our estimate of H.

Let the signal spectrum be of the form postulated by Mandelbrot. Let the noise be white. Then the signal and noise power spectrums are

$$\begin{cases} G(f) = Sf^{-2H-1} \\ N(f) = N \end{cases} \qquad \text{(C-1)}$$

Information rate as a function of the signal and noise spectrums is given by the relation[3]

$$r = \epsilon \int_{f_1}^{f_2} \ln\left[1 + \frac{G(f)}{N(f)}\right] df \quad \text{binits/second}, \quad \log_2 e = 1.443 \qquad \text{(C-2)}$$

Integrate over G(f) and N(f) to obtain total powers.

$$\begin{cases} P_S = \int_{f_1}^{f_2} G(f)\, df = S\int_{f_1}^{f_2} f^{-2H-1}\, df = -\frac{S}{2H} f^{-2H}\Big|_{f_1}^{f_2} = \frac{S}{2H}\left(f_1^{-2H} - f_2^{-2H}\right) \\ P_N = \int_{f_1}^{f_2} N(f)\, df = \int_{f_1}^{f_2} N\, df = N(f_2 - f_1) \end{cases} \qquad \text{(C-3)}$$

---

(2) W. Pratt, et. al., "Slant Transform Image Coding," pp. 191-209, Fig. 12, Digital Image Processing (H. Andrews, Ed.), IEEE Press, 1978.

(3) S. Goldman, Information Theory, Sec. 5.2, Eq. (43), Prentice-Hall, 1953.

Let M be the ratio of highest to lowest image harmonic (number of pixels per scan line). That is $M = f_2/f_1$, $f_2 \gg f_1$ and $M \gg 1$.

SOLVING FOR S AND N

$$\begin{cases} S = \dfrac{2HP_S}{f_1^{-2H} - f_2^{-2H}} = \dfrac{2HP_S}{\left[\left(\dfrac{f_2}{f_1}\right)^{2H} - 1\right] f_2^{-2H}} = \dfrac{2HP_S}{M^{2H} f_2^{-2H}} \\ N = \dfrac{P_N}{f_2 - f_1} = \dfrac{P_N}{\left(1 - \dfrac{f_1}{f_2}\right) f_2} = \dfrac{P_N}{(1 - M^{-1}) f_2} = \dfrac{P_N}{f_2} \end{cases} \tag{C-4}$$

Therefore,

$$\frac{G(f)}{N(f)} = \frac{\dfrac{2HP_S\, f^{-2H-1}}{M^{2H} f_2^{-2H}}}{P_N/f_2} = \left(\frac{2\,\overline{SNR}\,H}{M^{2H}}\right) \frac{f^{-2H-1}}{f_2^{-2H-1}} = ax^{-2H-1} \tag{C-5}$$

where x is a normalized frequence and $P_s/P_n = \overline{SNR}$.

Now for the cases we are interested in, G(f)/N(f) is a very large number compared to one, except at the band edges near f_2 and f_1. Therefore (C-2) can be approximated by the relation

$$r = \varepsilon \int_{f_1}^{f_2} \ln\left[\frac{G(f)}{N(f)}\right] df = \varepsilon f_2 \int_{x_1}^{x_2} \ln(ax^{-2H-1})\, dx$$

$$= \left[\ln a(x_2 - x_1) - (2H + 1) \int_{x_1}^{x_2} \ln x\, dx\right] \varepsilon f_2 \tag{C-6}$$

The f_2 factor appears because $df = f_2\,(df/f_2) = f_2\,dx$. Also when $f = f_2$, $x = 1$ and when $f = f_1$, $x = (f_1/f_2) = M^{-1}$. Carrying out the integration in (C-6).

$$r = \varepsilon f_2 \left[(1 - M^{-1}) \ln a - (2H + 1)\,(x \ln x - x) \Big|_{M^{-1}}^{1} \right] \qquad \text{(C-7)}$$

Note that $M^{-1} \ll 1$. Also

$$x \ln x \underset{x \to 0}{=} \frac{1}{y} \ln \left(\frac{1}{y}\right)_{y \to \infty} = -\frac{\ln y}{y} = \frac{\frac{d \ln y}{dy}}{\frac{dy}{dy}} = \frac{\left(\frac{1}{y}\right)}{1} \underset{y \to \infty}{=} 0$$

$$r = \varepsilon f_2 (\ln a + 2H + 1) = 2\varepsilon f_p (\ln a + 2H + 1) \qquad \text{(C-8)}$$

where $f_2 = 2f_p$.

Now f_2 is in cycles/sec (HZ). In order to meet the Nyquist sampling criterion it takes two pixels (a line pair) to define f_2. That is $f_2 = 2f_p$, where f_p is pixels scanned per second. This observation justifies the second right-hand expression in (C-8).

Next we must convert information rate to information density, R. The relation is

$$\frac{r(\text{bnits/sec})}{f_p(\text{pixels/sec})} = R\left(\frac{\text{bnits}}{\text{pixel}}\right)$$

So dividing both sides of (C-8) by f_p and using the value of a from (C-5)

$$R = 2\varepsilon \left[\ln \left(\frac{2\,\overline{SNR}\,H}{M^{2H}} \right) + 2H + 1 \right] = 2\varepsilon \left[\ln(2\overline{SNR}) + \ln H - 2H \ln M + 2H + 1 \right] \qquad \text{(C-9)}$$

Neglecting the lnH term, which is small, and solving for H we get

$$H = \frac{1 + \ell n(2\,\overline{SNR}) - R/2\varepsilon}{2(\ell nM - 1)} \qquad \text{(C-10)}$$

In the case of a typical visual image a representative set of parameters could be

$\overline{SNR}_{MAX}$ = 1000 (density range of 3)

M = 1000 lines (pixels)

R = 1 bit/pixel

$\left(\frac{R}{2\varepsilon}\right)$ = 0.347

Then

$$H_{IMAGE} = \frac{1 - 0.347 + \ell n(2000)}{2[\ell n(1000) - 1]} = 0.70 \qquad \text{(C-11)}$$

The value of H for the image will be greater than for the object scene because the imaging system decreases the high frequency spatial frequency content. This increases redundancy.

Compensation of H for the imager modulation transfer function (MTF) will be discussed in the next section where H is calculated from measured data.

C.3 DETERMINATION OF H FROM MEASURED DATA

The only spectrum available were several measured by Charles Hawthorne of China Lake. These spectrums were measured near Famoso in the Central Valley of California. The terrain type is agricultural. Four fixed beams produced resolvable elements which were moved over the terrain by the forward motion of the plane. The radiometer signals were digitized and the power spectrums were computed. Figures C-3 and C-4 show results for two separate runs. These spectrums were fitted to a power law curves using an HP-97 program. The calculated confidence level was 0.7. This is not too good a number. However the dispersion in the data is great. In addition there is evidence of saturation of the spectrometer channels at low frequencies.

At any rate, Table C-1 below gives the results of the curve fitting calculations as well as the average parameters. The power law curve is of the form $y = ax^b$.

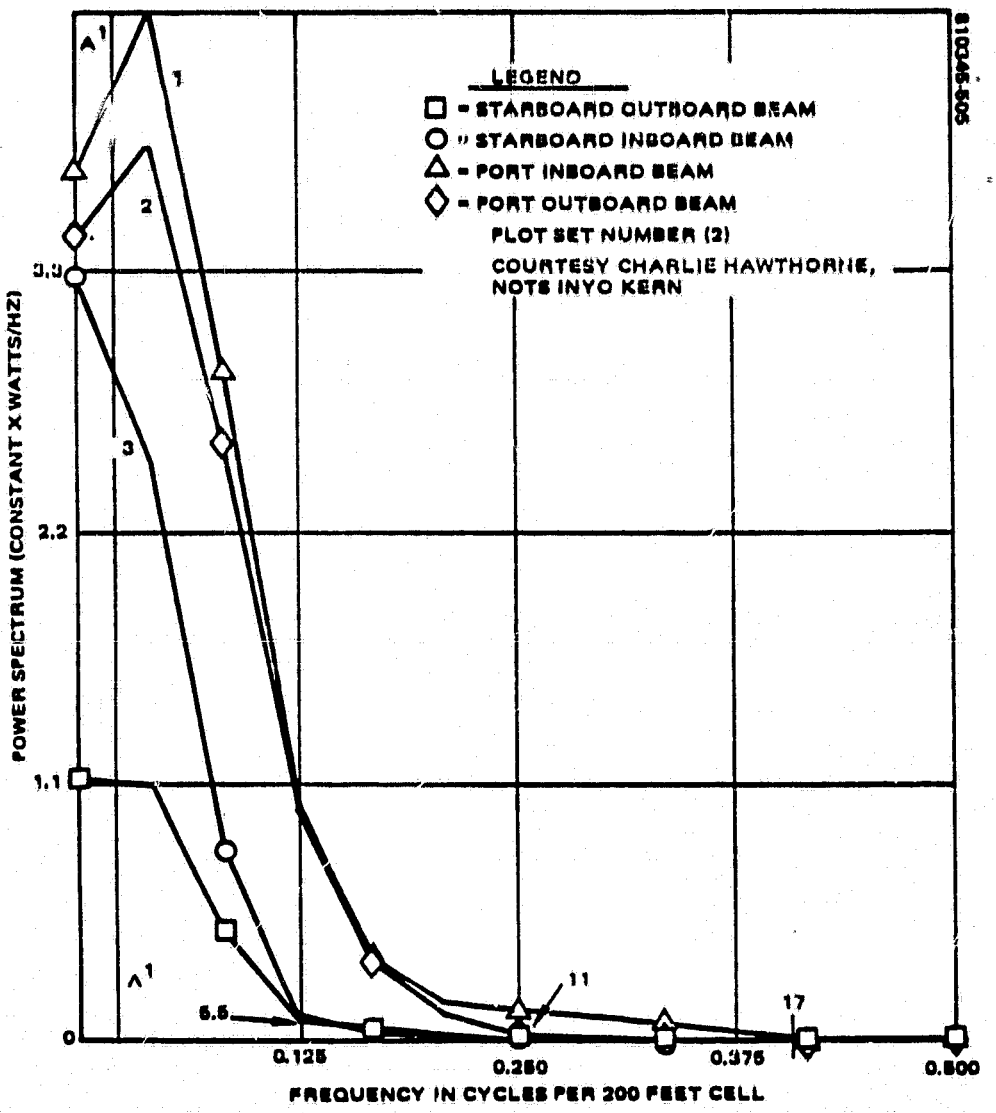


FIGURE C-3. POWER SPECTRUM VS FREQUENCY

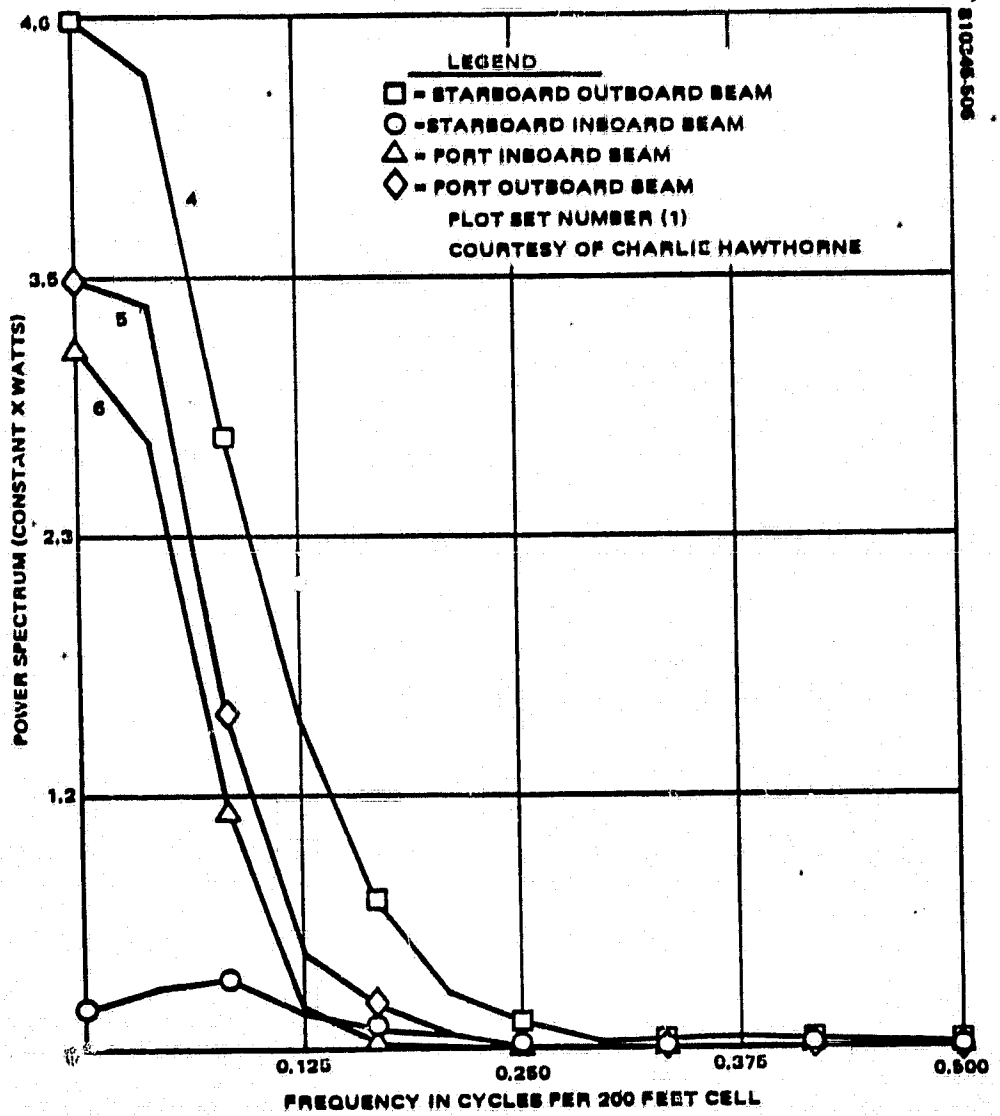


FIGURE C-4. POWER SPECTRUM VS FREQUENCY

TABLE C-1. POWER DENSITY SPECTRUM FIT TO POWER LAW CURVE

Run	1	2	3	4	5	6	Average
a	24.96	22.2	23.62	19.41	22.20	18.06	21.7
b	-1.86	-1.86	-3.05	-1.42	-2.18	-2.79	2.2
Coefficient of Determination	0.53	0.51	0.64	0.81	0.82	0.79	0.7

The H-value for the image is then

$$H_{IMAGE} = \frac{b - 1}{2} = \frac{2.2 - 1}{2} = 0.60 \qquad (C\text{-}12)$$

This is even lower than the value of H computed from the image information rate, namely 0.70.

Due to the sparseness of the data it is felt that it is reasonable to continue to use Mandelbrot value of H, namely 3/4. For the same of compactness, we will set $H + 1 = h$. Then $G(F) = F^{-h}$.

C.4 SPECTRUM NORMALIZATION

In order to use G(F) to represent the scene imaged by the radiometer, it must be converted to normalized form. In Appendix B it is shown that the system responds linearly to radiation intensity. Therefore, the RF power-density spectrum becomes a video voltage spectrum proportional to the RF spectrum. The scene spectrum we have been analyzing is given in terms of this video power spectrum. Let $\theta\rho^{-5/2}$ be this normalized video power spectrum. That is

$$G(F)G^*(F) \propto V_s(F)\, V^*(F) = S(F) = \theta\rho^{-5/2} \qquad (C\text{-}13)$$

The spectrum is two dimensional. The symbol ρ was chosen to emphasize that (C-13) gives the radial component of the spectrum.

The integral over this spectrum must be the mean square variance of scene radiant temperature, ΔT^2. The rms fluctuations ΔT vary from 20°K for rural areas to 40°K for urban areas. Land-water and ice-water scenes exhibit high contrast with a possible rms values as high as 50°K.

The constant θ is found by integrating the video power spectrum, setting the result equal to ΔT^2 and then solving for θ.

$$\Delta T^2 = \theta \int_0^{2\pi} \int_{\Delta F}^{\infty} \rho^{-5/2} (\rho \, d\rho \, d\psi) = 4\pi\theta \, \Delta F^{-1/2} \qquad \text{(C-14)}$$

Since $\Delta F^{-1} = \overline{FOV}$

$$\Delta T = 4\pi\theta \, \overline{FOV}^{1/2} \qquad \text{(C-15)}$$

Solving for θ

$$\theta = \frac{\Delta T^2}{4\pi \, \overline{FOV}^{1/2}}$$

and

$$S(F) = \left(\frac{\Delta T^2}{4\pi \sqrt{\overline{FOV}}} \right) \rho^{-5/2} = \theta \rho^{-5/2} \qquad \text{(C-16)}$$

APPENDIX D

OPTIMIZING ANTENNA ILLUMINATION AND BANDWIDTH FOR MAXIMUM RADIOMETRIC INFORMATION TRANSFER

By Jack Gustincic (Aug 1966)

INTRODUCTION

The purpose of this technical report is to provide antenna design criteria for an airborne radiometric mapping system. The analysis employed in determining optimum antenna illumination and bandwidth is considered somewhat novel in that communication theory concepts have been used for establishing an objective mathematical evaluation of system performance. A model of a radiometric mapping system is formulated and the amount of information (in the communications sense) available at the system output is calculated as a function of various system parameters. The information content is then maximized to predict an optimum antenna configuration for a receiver having a given temperature resolving ability.

Using the information theory approach, it is demonstrated that a definite microwave bandwidth and illumination taper must be chosen to provide maximum information at the system output. It is shown that conventional design approaches which include severe tapering of the antenna illumination for sidelobe reduction and large front end bandwidths for increased temperature resolution do not lead to maximum information content. The effects of taper and bandwidth on the spatial frequencies being scanned by the system are described in this analysis and for a typical system, calculations indicate that optimum information is achieved with illuminations near uniform (maximum monochromatic gain) and bandwidths around 4% - 7%, depending on the resolving capabilities of the receiver and the variance of the temperature profile being mapped.

The use of information content as a measure of system performance has several definite advantages. The system output, for example, can be processed in many ways to yield a number of representations of the profile being mapped. A good information content at the output insures that most processing or coding schemes will have the most amount of input data possible and that little data has been destroyed inadvertently by the receiving system before processing. Also, since information is an objective quantity, it can be maximized without regard to subjective quandaries such as how much of the main beam is actually usefull, etc.

Clearly, it is impossible to analyse every radiometric mapping system. The particular antenna and receiving system discussed in the following was chosen because it was thought to be physically realizable. In any event the following analysis will serve as an example of how many similar systems can be treated to provide maximum information transfer.

Section I begins with a basic description of how microwave noise from the earth is received by the mapping antenna. The concepts of emissivity and temperature are introduced and an expression for power received at the antenna is given. In Section II, this formula is applied to a particular mapping system having an antenna with a separable aperture distribution. The antenna pattern is then written as the product of patterns longitudinal and transverse to the direction of flight of the mapping vehicle.

It is then shown in Section III that the two dimensional mapping problem reduces to a series of one dimensional problems involving the transverse or side-looking properties of the antenna alone. In keeping with the state of the art, the transverse antenna configuration is assumed to be an array, electronically scanned with ferrite phase shifters. Such an antenna choice allows good pattern control and a fast data gathering capability. Also, in this section, the concept of spatial frequencies is introduced. The temperature profile is expanded in terms of its spatial frequency components and the spatial frequency transfer characteristics of the antenna are discussed for the monochromatic case. Finally, the effect of microwave bandwidth is calculated and it is demonstrated that increased bandwidth degrades the antenna by tapering the spatial frequency spectrum and introducing unwanted noise.

In Section IV the effects of the receiver are discussed. A particular receiver employing straight through detection of the microwave signal is selected for the sake of argument. The noise at the output due to inadequate temperature resolution is calculated. The mean square signal to noise ratio at the nth spatial frequency

$$\left(\frac{S}{N}\right)^2_n$$

is then calculated in terms of both antenna and receiver effects. The total information at the receiver output is then written as the sum of the information contained in each spatial frequency

$$I = \sum_{n=1}^{N} \ln \left[1 + \left(\frac{S}{N}\right)^2_n\right]$$

and this quantity is used as an objective measure of system performance.

Information is plotted as a function of bandwidth for various illumination tapers assuming a cosine-squared-on-a-pedestal family of illumination functions. Bandwidth and tapers which provide maximum information are presented for a 51 element array.

In the last section, the loss of information due to antenna tapering is discussed. It is shown that unwarranted antenna taper equivalently increases the receiver noise temperature or increases the microwave bandwidth required to produce a given amount of information at the output. A quantity called "resolution efficiency" is presented as a function of taper. This factor gives a total system efficiency and is analogous to aperture efficiency in the case of a monochromatic antenna.

I. BASIC CONSIDERATIONS

The fundamental equations describing the reception of ground mapping radiometric information will now be developed. For purposes of analysis, it will be assumed the earth is flat and positioned in the x-y plane as shown in Figure 1. A receiving aperture, S_a, is located a height, h, above the earth as shown. A spherical coordinate system described by the angles θ and ϕ is also indicated.

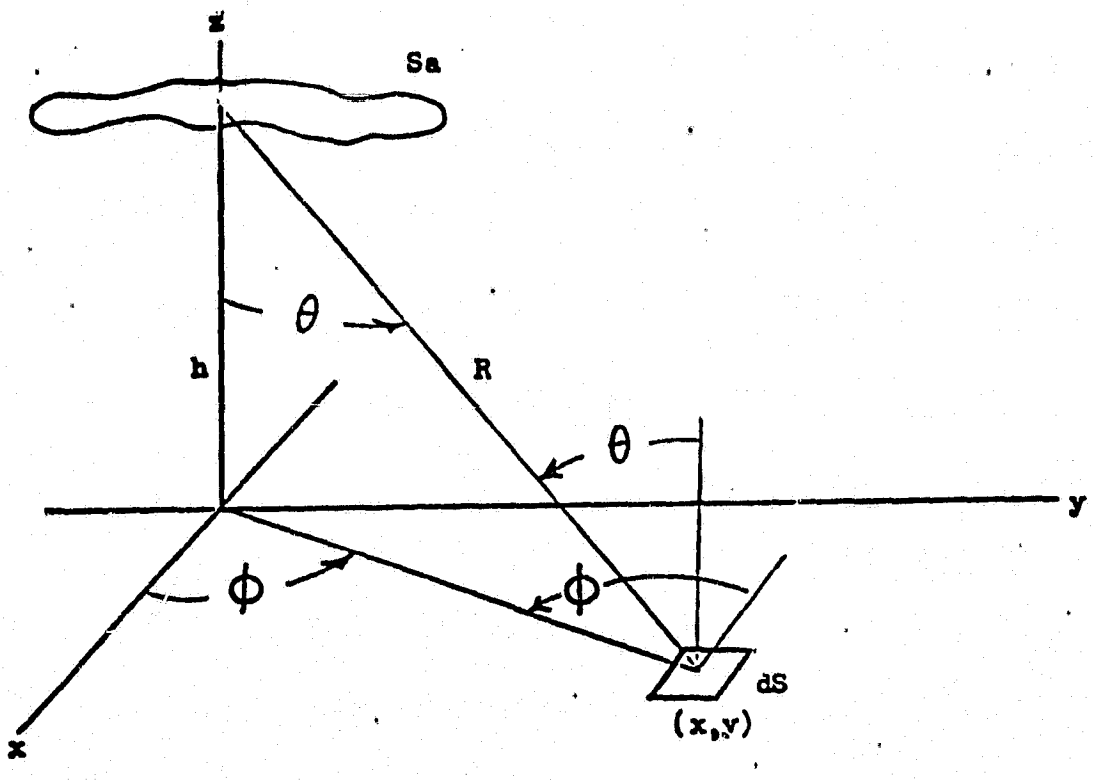


Figure 1.

Consider an element of the earth, dS, located at the point (x,y). This element radiates microwave noise upward into the half space $z > 0$. The noise is produced by thermal effects within the earth and by reflection of noise signals from other sources. It is assumed that dS is taken large enough so that the noise emitted from it is statistically independent of the noise emitted by any other similar element in the x-y plane. The element of dS is completely described by two additional characteristics; emissivity and temperature.

I. BASIC CONSIDERATIONS (Cont'd)

The temperature of the point (x,y) can be defined in terms of the density of power flowing up and out of dS. In particular, if dS radiates a total average power dP, then the temperature of (x,y) is defined as:

$$T(x,y) \equiv \frac{\lambda_o^2}{2kB} \frac{dP}{dS} \quad {}^\circ K \tag{1}$$

where

k = Boltzmann's constant
B = bandwidth under consideration, Hz
λ_o = center frequency wavelength

Thus, temperature defined in this manner is a measure of the time average z component of the Poynting vector averaged over the area dS.

The emissivity of the element describes any directional radiation characteristics it might have. The emissivity of dS can be defined as simply the antenna gain function of the aperture which dS represents. In general the emissivity function will be different for the different frequencies contained in the spectrum of noise emitted by dS. In particular the following definition for the emissivity of the point (x,y) can be made:

$$e(\theta, \phi, x, y, \omega) = \frac{\text{power radiated in the direction } \theta, \phi \text{ at frequencies between } \omega \text{ and } \omega + d\omega}{\text{power radiated in the direction } \theta, \phi \text{ at frequencies between } \omega \text{ and } \omega + d\omega \text{ if dS radiated isotropically into half space } z > 0.} \tag{2}$$

Recognizing that Equation (2) is simply the gain of the aperture dS, the noise power received by the antenna aperture can easily be written. Each spectral component of noise which leaves dS may be treated as a monochromatic signal transmitted with a gain given by Equation (2). The power, dP_ω, carried away from dS by spectral componencts of the noise having frequencies between ω and $\omega + d\omega$ can be calculated assuming the noise follows the Rayleigh—Jeans law:

$$dP_\omega = C\omega^2 \, d\omega \tag{3}$$

where C is a constant. The constant C can be determined by requiring that the total power contained in all frequencies be dP, the total time average power emitted by dS. Thus,

$$dP = \int_0^\infty dP_\omega = C \int_{\omega_o - \pi B}^{\omega_o + \pi B} \omega^2 \, d\omega \approx \omega_o^2 \, 2\pi BC \tag{4}$$

Solving Equation (4) for the constant C the spectral power leaving dS can be written in terms of the temperature as given by Equation (1).

$$dP_\omega = \frac{dP}{2\pi B}\,\frac{\omega^2}{\omega_o^2}\,d\omega = \frac{k}{\pi\lambda^2}\,T(x,y)\,d\omega dS \qquad (5)$$

Now if the power given by Equation (5) leaves dS with a gain given by Equation (2), the power received by the antenna, S_a, in the frequency range ω, $\omega + d\omega$ is simply:

$$\frac{dP_\omega}{8\pi^2 R^2}\,\lambda^2\,\epsilon(\theta,\phi,x,y,\omega)\,G(\theta,\phi,\omega) \qquad (6)$$

where $G(\theta,\phi,\omega)$ is the power gain of the antenna. Substituting for dP its value given by Equation (5) and integrating over all frequencies, the total time average power received by the antenna from the element dS is then:

$$\frac{k\,T(x,y)}{(2\pi)^3 R^2}\int_o^\infty \epsilon(\theta,\phi,x,y,\omega)\,G(\theta,\phi,\omega)\,d\omega\,dS \qquad (7)$$

Finally, since dS is statistically independent of position, the time average power received by the antenna from all possible elements of earth is simply the integral of Equation (7) over dS and is given by:

$$P_{rec} = \frac{k}{(2\pi)^3}\int_{-\infty}^{+\infty}\int\frac{T(x,y)}{R^2}\int_o^\infty \epsilon(\theta,\phi,x,y,\omega)\,G(\theta,\phi,\omega)\cdot d\omega\,dx\,dy \qquad (8)$$

in which $R^2 = h^2 + x^2 + y^2$ and $\cos\theta = h/R$, $\cos\phi = x/R\sin\theta$. Equation (8) describes the manner in which the antenna receives noise power from the earth and is fundamental to the analysis of a ground mapping radiometer.

II. AN AIRBORNE MAPPING SYSTEM

Since the width of any airborne antenna is limited by the width of the underside of the aircraft, it seems reasonable to design a radiometric mapping system with the idea of enhancing the aperture limited side-looking capability. Thus, in order to provide pattern control and rapid scan ability, it will be assumed that the antenna is chosen to be an electronically scanned array in the plane transverse to the direction of the mapping vehicle. Furthermore,

II. AN AIRBORNE MAPPING SYSTEM (Cont'd)

In order to facilitate the initial formulation of the problem, it will be assumed that the emissivity of the ground is independent of look-angle and position.

For the situation under consideration, scanning effects are best described in terms of the coordinate system shown in Figure 2. A general point (x,y) on the earth is located in terms of the direction cosine, u, of the distance vector and the longitudinal distance, y, from the airplane. The coordinates (u,y) are related to the coordinates (x,y) as follows:

$$y = y$$
$$u = x / \sqrt{h^2 + x^2 + y^2} \qquad (9)$$

The Jacobian of the transformation (Equation (9)) can be calculated directly as:

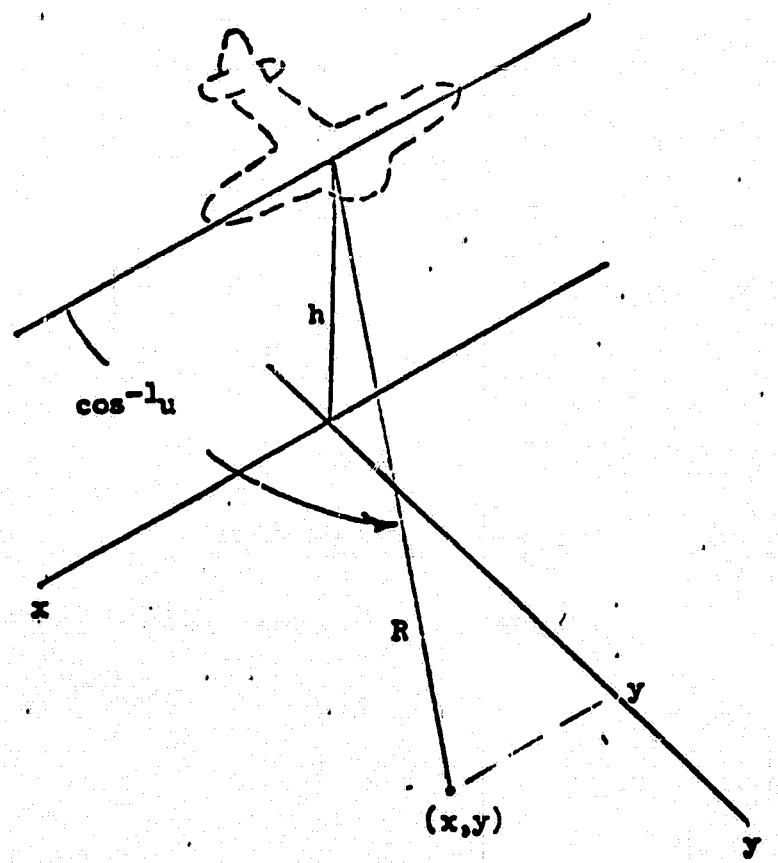


Figure 2.

$$J\left(\frac{u,v}{x,y}\right) = \frac{\partial u}{\partial x} = \frac{h^2 + y^2}{R^3}$$

so that the element of area becomes:

$$dx\,dy \longrightarrow \frac{R^3}{h^2 + y^2}\,du\,dy \tag{10}$$

At any single frequency, it is assumed that antenna pattern is separable and can be written as:

$$G(\omega, \theta, \phi) = G_t\,(u - u_s)\,G_L\,(u,y)\,\frac{h}{R} \tag{11}$$

The function G_t is the transverse gain function of the antenna where u_s is the scan produced by an electronic phase shift. G_L is the longitudinal gain of the antenna and is in general a function of both u and y. The factor h/R represents the element factor of the array as it is the cosine of the angle the distance vector makes with the antenna normal.

Substituting Equation (10) and (11) into (8) and integrating over frequency gives the following expression for the power received by the antenna:

$$P_{rec}(u_s) = \frac{k}{(2\pi)^3}\int_0^{\infty}\int_{-1}^{+1} G_t(u - u_s)\int_{-\infty}^{+\infty} T(u,y)\,G_L(u,y)\cdot \frac{h}{h^2 + y^2}\,dy\,du\,d\omega \tag{12}$$

The emissivity has been taken as unity and T(u,y) is the temperature at the point (x,y) located by the coordinates (u,y), i.e., T(u,y) is the temperature at the point:

$$\left(x = \frac{u^2}{1 - u^2}\,[h^2 + y^2]\,,\ y\right) \tag{13}$$

Finally, as the mapping vehicle moves along, the temperature profile appears to translate below it in the y direction. Thus, when the airplane moves a distance y_s from the origin, the received power becomes:

II. AN AIRBORNE MAPPING SYSTEM (Cont'd)

$$P_{rec}(u_s,y_s) = \frac{k}{(2\pi)^3}\int_0^{\infty}\int_{-1}^{+1} G_t(u-u_s)\int_{-\infty}^{+\infty} T(u,y-y_s)\cdot G_L(u,y)\,\frac{h\,dy}{h^2+y^2}\,du\,d\omega \qquad (14)$$

The form of Equation (14) indicates that the received power as a function of u_s,y_s is related to the earth temperature by a linear transformation. Information about the temperature can be obtained from the received power by investigating the manner in which the spacial frequencies of the temperature are affected by convolution with the antenna pattern. This will be done in the following section.

III. RECEPTION OF SPATIAL FREQUENCIES

The form of Equation (14) indicates that at any given position of the aircraft, y_s, the transverse gain function scans an effective temperature

$$\bar{T}(u) = \frac{1}{2\pi^2} \int_{-\infty}^{+\infty} T\,(u,\, y - y_s)\, G_L\,(u,\, y)\, \frac{h}{h^2 + y^2}\, dy \qquad (15)$$

which is the actual temperature integrated longitudinally with the longitudinal gain of the antenna. The objective now will be to design a system which reproduces $\bar{T}(u)$ as accurately as possible so that Equation (15) can be inverted to obtain the best possible representation of the actual temperature. In this way the total two dimensional problem involving u and y first requires the solution to a one dimensional problem involving u alone.

As a first step it will be assumed that $\bar{T}(u)$ is independent of frequency. The consequences of this assumption will be discussed later. Now at any given y_s the received power is given by Equations (14) and (15) as

$$P_{rec}\,(u_s) = \frac{k}{4\pi} \int_{0}^{\infty} \int_{-1}^{+1} G_t\,(u - u_s)\, \bar{T}(u)\, du\, d\omega \qquad (16)$$

Over the range $-1 \leq u \leq 1$ the integrated temperature can be expressed in terms of a Fourier series of spatial frequencies

$$\bar{T}(u) = \sum_{n=-\infty}^{+\infty} T_n\, e^{j n\pi u} \qquad (17)$$

where

$$T_n = \tfrac{1}{2} \int_{-1}^{+1} \bar{T}(u)\, e^{j n\pi u}\, du \qquad (18)$$

If the transverse array is made up of N + 1 half-wave spaced* elements with frequency independent phase shifters, the gain function is given by

---

*It can be shown that at any particular frequency a half-wave spaced array receives all the resolution information which is practicably obtainable. Closer or farther spacing produces no additional resolving capability.

III. RECEPTION OF SPATIAL FREQUENCIES (Continued)

$$G_t(u - u_s) = \sum_{n=-N}^{+N} C_n e^{-j n \pi \left| \frac{\omega}{\omega_0} u - u_s \right|} \qquad (19)$$

where ω_0 is the center frequency of the microwave band. Actually, the gain function is only a function $u - u_s$ at the center frequency. The fact that the spatial frequencies of G_t depend on ω / ω_0 will be shown to give rise to output degradation. The amplitudes, C_n, are determined by the excitation of the array. For an array with progressive excitation, a_K, $K = 1, 2, \ldots N + 1$, the spatial frequency power gain amplitudes are given by

$$C_n = \frac{\sum_{K=1}^{N+1-n} a_K a_{K+n}}{\sum_{K=1}^{N+1} a_K^2} \qquad n = 0, 1, \ldots N \qquad (20)$$

As will be seen, the C_n will have the effect of weighting the spatial frequencies of the temperature and will play an important role in determining the total system output. The C_n then represents the spatial frequency transfer characteristics of the antenna at the center frequency of the microwave band.

Figure 3 shows how antenna illumination effects the C_n for a 51 element array. Since C_0 is the average value of the power pattern

$$C_0 = \int_{-1}^{+1} G_t(u)\, du = 1 \qquad (21)$$

its value is unity regardless of the illumination. For a uniform array, $a_K = 1$, the amplitudes decay linearly with increasing n,

$$C_n = 1 - \frac{n}{N+1} \qquad n = 0, 1, \ldots N \qquad (22)$$

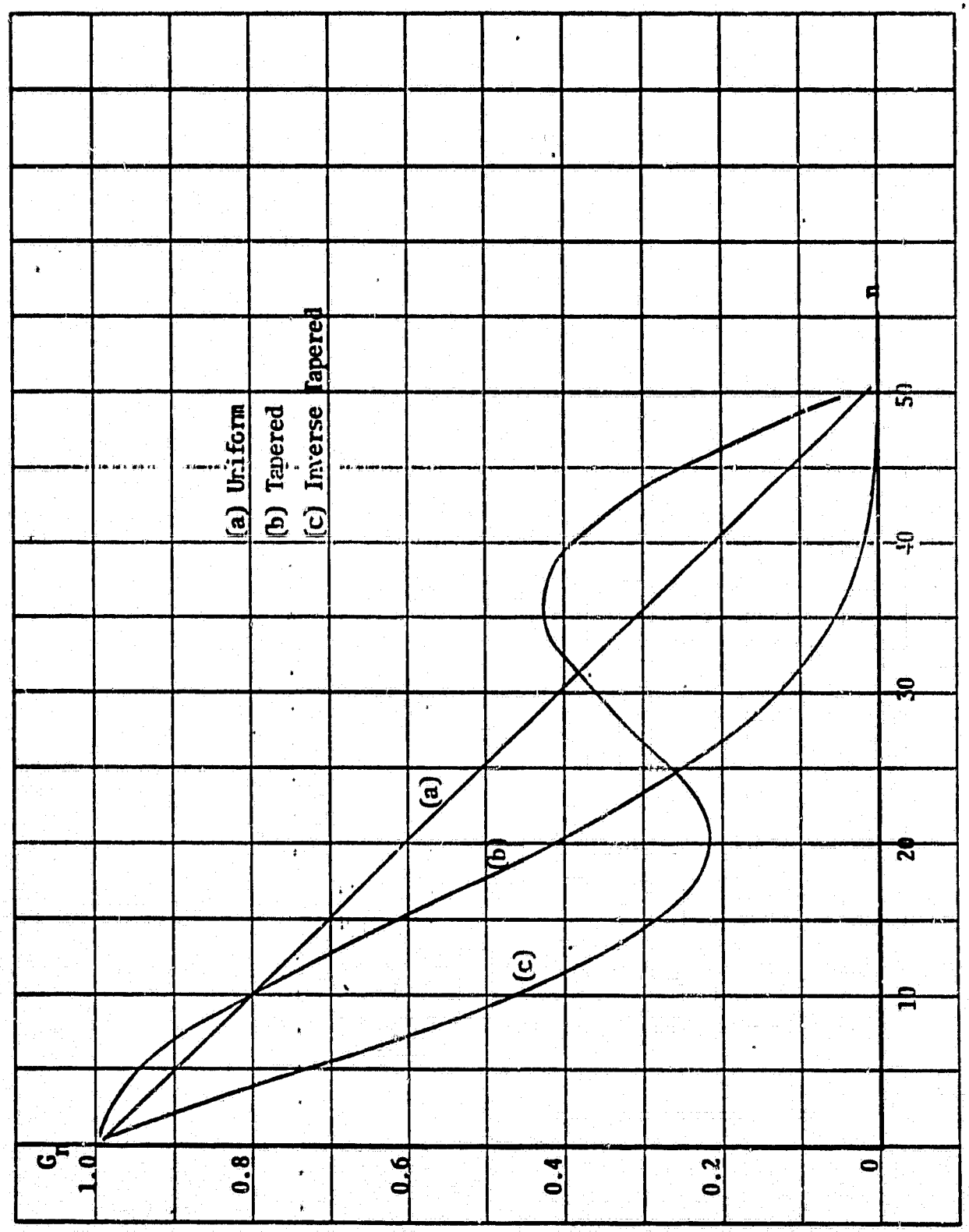


Figure 3. Spatial Frequency Transfer Characteristics for Various Illuminations

III. RECEPTION OF SPATIAL FREQUENCIES (Continued)

as shown by curve (a). Since all the a_K are assumed to have the same phase, the area under the G_n curve is simply the maximum gain of the far field pattern

$$\sum_{n=-N}^{+N} G_n = G_t(0) = G_{t\,max} \tag{23}$$

Since a uniform illumination produces maximum gain, a tapering of the illumination will give a curve having less area than (a). Curve (b) illustrates the effect of tapering. An illumination of the form

$$a_K = .1 + \sin^2\left[\frac{K-1}{50}\right]\pi$$

was chosen for this case. This excitation produces a far field pattern having a -41 db side lobe level. As indicated, the tapering tends to increase the low spatial frequency response at the expense of the high frequencies.

Inverse tapering has the opposite effect as shown by curve (c). In this case the illumination was taken to be the inversely tapered function

$$a_K = -1.143 + \sin^2\left[\frac{K-1}{50}\right]\pi$$

which produces a 5 db side lobe level in the far field. For the inversely tapered illumination the high spatial frequency response is increased at the expense of the lows. It is apparent that by selecting the array illumination properly, any part of the spatial frequency spectrum of $G_t(u)$ can be emphasized but only at the cost of lowering other parts of the spectrum.

Substituting Equation (19) into (16) the expression for the received power becomes

$$P_{rec.}(u_s) = \frac{k}{4\pi} \sum_{n=-N}^{N} G_n \int_{-1}^{+1} \overline{T}(u) \left[\int_{\omega_0 - \pi B}^{\omega_0 + \pi B} e^{-j n\pi \frac{\omega}{\omega_0} u}\, d\omega \right] du$$

$$e^{j n\pi u_s} \tag{24}$$

III. RECEPTION OF SPATIAL FREQUENCIES (Continued)

If the bracketed exponential in Equation (24) did not have to be integrated over the band then Equation (24) would become

$$P_{rec}(u_s) = \frac{k}{2\pi} \sum_{n=-N}^{N} G_n T_n e^{j n\pi u_s}$$

and the received power would simply contain the spatial frequencies of the temperature weighted by the amplitudes G_n. However, this is not the case. Integrating the exponential gives

$$\int_{\omega_0 - \pi B}^{\omega_0 + \pi B} e^{-j n\pi \frac{\omega}{\omega_0} u} d\omega = 2\pi B e^{-j n\pi u} \frac{\sin(n\pi\Delta/2)u}{(n\pi\Delta/2)u} \qquad (25)$$

where Δ is the percent bandwidth

$$\Delta = 2\pi B/\omega_0 \qquad (26)$$

The sin X/X factor in Equation (25) represents an error caused by the fact that the element spacing is not exactly a half wavelength over the entire band. This error can be described mathematically by expanding the sin X/X term in a Fourier series as follows

$$\frac{\sin(n\pi\Delta/2)u}{(n\pi\Delta/2)u} = \sum_{K=-\infty}^{+\infty} C_K\left(\frac{n\pi\Delta}{2}\right) e^{j K\pi u} \qquad (27)$$

where

$$C_K\left(\frac{n\pi\Delta}{2}\right) = \frac{1}{2}\int_{-1}^{+1} \frac{\sin(n\pi\Delta/2)u}{(n\pi\Delta/2)u} e^{-j K\pi u} du$$

$$= \frac{1}{n\pi\Delta}\left[\mathrm{Si}\left(K\pi + \frac{n\pi\Delta}{2}\right) - \mathrm{Si}\left(K\pi - \frac{n\pi\Delta}{2}\right)\right] \qquad (28)$$

with

$$Si\ (x) = \int_0^x \frac{\sin t}{t}\ dt$$

The first three coefficients C_0, C_1 and C_2 are shown in Figure 4 for values of bandwidth

$$\Delta < 16/N\pi$$

For a 50 element array this would correspond to bandwidths up to about 10 percent. Over this range, the remaining coefficients C_3, C_4, . . . are negligably small. Using Equations (24) and (25) in (24) the received power becomes

$$P_{rec}\ (u_s) = kB \sum_{n=-N}^{+N} G_n\ C_0\ \left(n \frac{\Delta\pi}{2}\right) T_n e^{j\ n\pi u_s}$$

$$+ kB \sum_{n=-N}^{N} G_n \left[C_1\ \left(n \frac{\Delta\pi}{2}\right) (T_{n+1} + T_{n-1}) \right.$$

$$\left. + C_2\ \left(n \frac{\Delta\pi}{2}\right) (T_{n+2} + T_{n-2}) \right] e^{j\ n\pi u_s} \qquad (29)$$

where all the C_K for K23 have been neglected.

The effects of system bandwidth are exhibited in two ways in Equation (29). First of all, the spatial frequency components in the first sum and weighted by the factor $C_0\ (n\Delta\pi/2)$. It is seen from Figure 4 that this weighting tends to reduce the amplitude of the higher spatial frequencies. The larger the bandwidth the more pronounced the tapering of the spectrum of the T_n. Secondly the second sum in Equation (29) represents an addition of noise. The nth frequency component of this sum is related to T_{n+1}, T_{n-1}, T_{n+2} and T_{n-2}. Since these amplitudes are completely independent of T_n they may be considered as noise. Figure 4 indicates that the weighting factors C_1, C_2 and thus the noise, increase with increasing bandwidth. Antenna bandwidth then corrupts the system output in two ways: tapering down the spatial frequencies of the temperature profile and introducing unwanted noise in the output. Further degredation of the received signal by the receiving system will now be discussed.

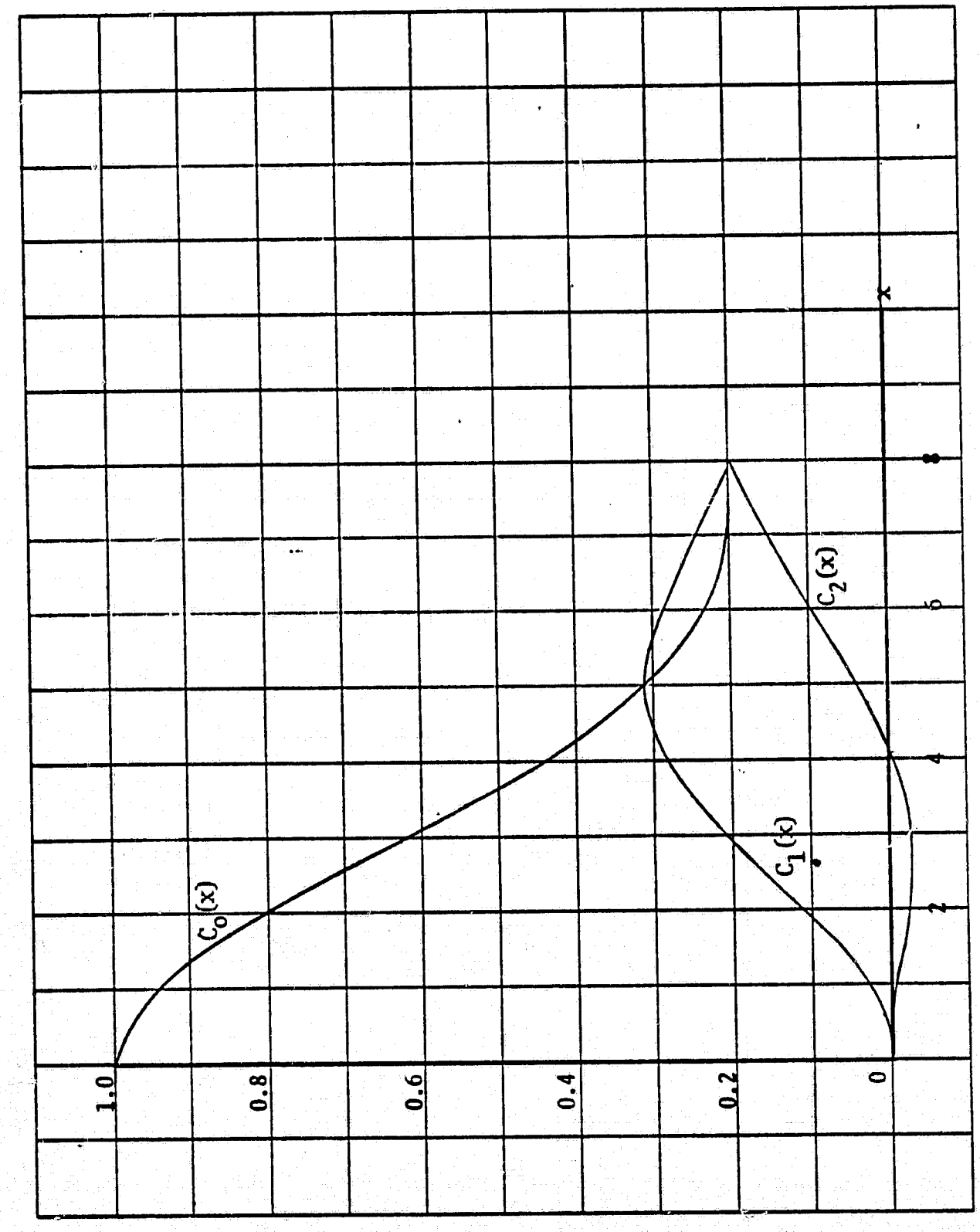


Figure 4. The First Three Coefficients in the Expansion of Sin X/X

IV. RECEPTION OF THE MICROWAVE SIGNAL

A true measurement of average power requires an indefinitely long period of time over which instantaneous power is averaged. Since practical measuring times are limited, radiometric receiving systems all have one factor in common. The receiver output varies with time producing an indication of received microwave power which fluctuates about the true average level. In most cases, when gain instability is neglected, the variance of the received power, ΔP_{rec}^2, and the corresponding variance in measured temperature, ΔT^2, can be estimated by the formula

$$(\Delta P_{rec} / kB)^2 = \Delta T^2 = (K F T_o)^2 \frac{b}{B} \qquad (30)$$

in which

K = constant depending on receiver type
($K \approx 1.8$ for direct detection of microwave signal)

T_o = ambient temperature

F = receiver noise figure

b = post detection bandwidth

B = predetection bandwidth

Receiver gain fluctuations also contribute to a variance in the output but these effects can be minimized independently by utilizing high scan or switching rates. On the other hand, the fluctuations described by (30) are inversely proportional to the microwave bandwidth and thus represent an important consideration in front end system design. The exact manner in which the receiver output variations effect the radiometric information under investigation is dependent on the system configuration. A typical system involving direct detection of the microwave noise will now be discussed.

The system to be considered is illustrated in Figure 5. The antenna is scanned in syncronism with a storage or display unit at a rate of f_s scans per second. For the purposes of this analysis it will be assumed that the entire 180 degree range, $-1 < u_s < 1$, is scanned although this restriction is not necessary.

IV. RECEPTION OF THE MICROWAVE SIGNAL (Continued)

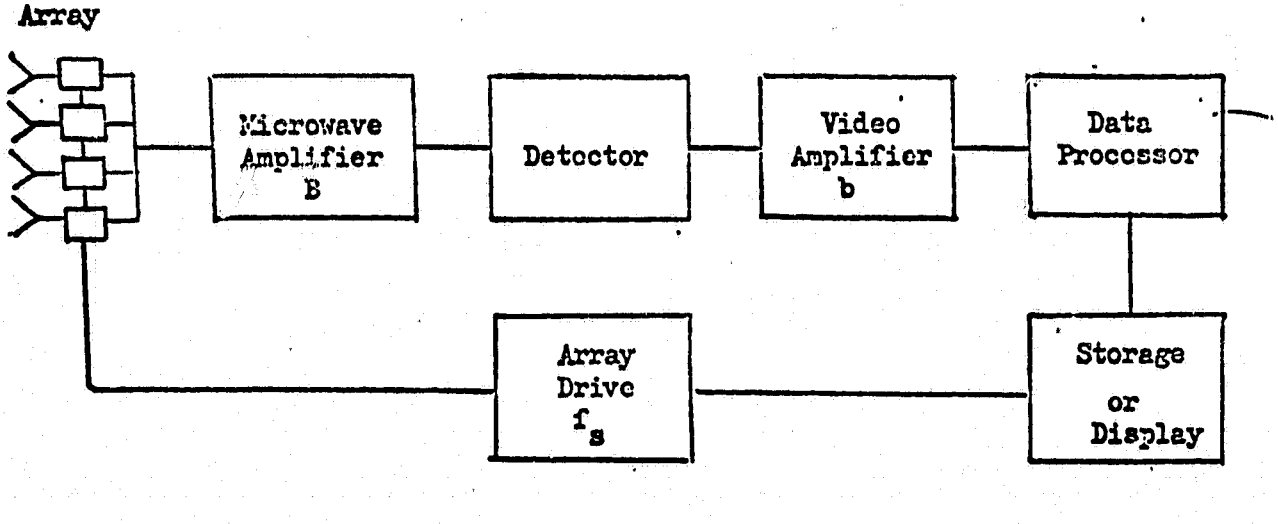


Figure 5

It is assumed that f_s is high enough so that receiver gain fluctuations only effect the d-c value of the output. As shown, the microwave noise from the antenna is amplified, detected, amplified again and then passed through a processor before storage or display. The data processor has been included since it provides an immediate means of real time processing at the scan rate f_s. For example, this stage could simply be a filter which would variously weight the spatial frequencies of the input after the scan process converts them to time frequencies.

When the antenna is scanned, the average power indicated at the output of the video detector begins to vary with time. In particular, the "signal" part of (29) produces a video output temperature

IV. RECEPTION OF THE MICROWAVE SIGNAL (Continued)

$$T(t) = \sum_{-N}^{+N} G_n \, C_o \left(\frac{n\pi\Delta}{2} \right) T_n \, e^{j n 2\pi f_s t} \qquad (31)$$

In this way the spatial frequencies of the input are converted to time frequencies which are multiples of the scan frequency f_s. Since the video amplifier must be capable of passing all these frequencies, the bandwidth of the amplifier must be chosen as

$$b = N f_s \qquad (32)$$

Now if a total time τ is available for transverse scanning there will be τf_s total scans. If in addition, the receiver output is averaged over these scans, the variance in the output is given by (30) reduced by the factor τf_s. Making use of (26) and (32) gives

$$\Delta T_{av}^2 = \frac{(K F T_o)^2}{\tau f_s} \frac{b}{B} = \frac{(K F T_o)^2}{\tau f_o} \frac{N}{\Delta} \qquad (33)$$

Thus the output fluctuation becomes independent of the scan rate, only determined by the total number of microwave cycles sampled; $\tau f_o \Delta$.

We are now in a position to examine the information passing through the system. If the n th frequency component of the "signal" given by (31) is sampled it will be found to have a variance

$$\sigma_s^2(n) = G_n^2 \, C_o \left(\frac{n\pi\Delta}{2} \right)^2 \sigma_{T_n}^2 \qquad (34)$$

where $\sigma_{T_n}^2$ is the variance of the magnitude of the n th spatial frequency of the temperature. If the n th frequency component of the noise part of (29) is sampled the variance of the noise due to antenna bandwidth effects will be

$$\sigma_{NA}^2(n) = G_n^2 \, C_1^2 \left(\frac{n\Delta\pi}{2} \right) \left| \sigma_{T_{n-1}}^2 + \sigma_{T_{n-1}}^2 \right| + C_2^2 \left(\frac{n\Delta\pi}{2} \right) \left| \sigma_{T_{n+2}}^2 + \sigma_{T_{n-2}}^2 \right| \qquad (35)$$

$$\approx 2 G_n^2 \, \sigma_{T_n}^2 \, C_1^2 \left[\left(\frac{n\Delta\pi}{2} \right) + C_2^2 \left(\frac{n\Delta\pi}{2} \right) \right]$$

where it has been assumed that the variance of the temperature components does not vary radically with spatial frequency. This variance can be estimated by assuming it is the same for all spatial frequencies. Then, if all the spatial frequencies which could be passed by the antenna produced a temperature profile which deviated δT degrees about the average, the variance in the n th spatial frequency would be

$$\sigma_{T_n}^2 = \frac{\delta T^2}{N} \tag{36}$$

Finally, assuming the receiver noise is distributed uniformly about the spatial frequency spectrum, when sampled at the n th frequency, the noise will have a variance

$$\sigma_{NR}^2 = \Delta T_{av}^2 / N = \frac{(K F T_o)}{\tau_{f_o}} \frac{1}{\Delta} \tag{37}$$

Now, if, for the sake of argument, the values of the magnitudes of each of the spatial frequency components are distributed normally among all possible temperature profiles to be investigated and if the receiver noise is assumed purely random then the variances given by (34) (35) and (36) may be superimposed to predict the variance of the receiver output at a given spatial frequency. Figure 6 illustrates the expected value of the magnitude of the frequency spectrum at the output of the video amplifier. The signal portion of the output spectrum is made up of the original spatial frequencies weighted by the gain and frequency effects of the antenna. The noise portion of the spectrum includes the effects of antenna bandwidth and receiver temperature resolution.

IV. RECEPTION OF THE MICROWAVE SIGNAL (Continued)

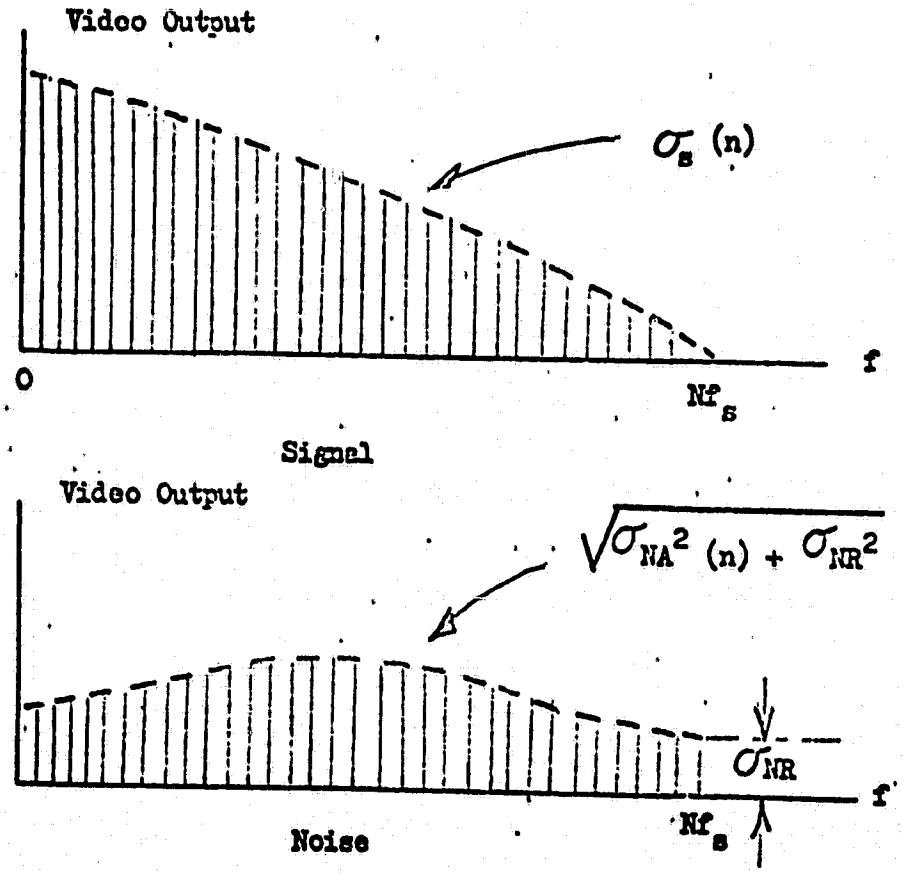


Figure 6

The square signal to noise ratio at the n th spatial frequency is given by

$$\left(\frac{S^2}{N^2}\right)_n = \frac{\sigma_s^2 (n)}{\sigma_{NA}^2 (n) + \sigma_{NR}^2} \qquad (38)$$

IV. RECEPTION OF THE MICROWAVE SIGNAL (Continued)

In order to obtain an objective figure of merit for the system, the classical ideas of information theory will be invoked and the total information present at the output of the video amplifier will be defined as

$$I = \sum_{n=1}^{N} \ell n \left[1 + \left(\frac{S^2}{N^2} \right)_n \right] \tag{39}$$

The information contained in the n = o, d - c, component of the output has been neglected because of receiver instability considerations. For a given receiver and antenna illumination the information is a function of percentage microwave bandwidth, Δ. Substituting into (39), the expression for information as a function of bandwidth becomes

$$I(\Delta) = \sum_{n=1}^{N} \ln \left[1 + \frac{A_n(\Delta)}{B_n(\Delta) + \frac{R}{\Delta G_n}} \right] \tag{40}$$

where

$$A_n(\Delta) = \left[\mathrm{Si}\left(\frac{n\pi\Delta}{2}\right) / \left(\frac{n\pi\Delta}{2}\right) \right]^2$$

$$B_n(\Delta) = \frac{2}{(n\pi\Delta)^2} \left\{ \left[\mathrm{Si}\left(\pi + \frac{n\pi\Delta}{2}\right) - \mathrm{Si}\left(\pi - \frac{n\pi\Delta}{2}\right) \right]^2 + \left[\mathrm{Si}\left(2\pi + \frac{n\pi\Delta}{2}\right) - \mathrm{Si}\left(2\pi - \frac{n\pi\Delta}{2}\right) \right]^2 \right.$$

$$R = \left(\frac{K F T_o}{\delta T} \right)^2 \frac{N}{\tau f_o} \tag{41}$$

IV. RECEPTION OF THE MICROWAVE SIGNAL (Continued)

which is valid for bandwidths up to about 10 percent. The parameter R describes the temperature resolving ability of the system relative to the temperature profile being investigated. It increases as the square of the receiver noise temperature and inversely as the variance of the temperature profile being scanned and integration time. For a profile that might have a deviation of $\delta T = 60$ degrees and radiometer figures which might be typically

$$K = 1.8$$

$$F = 5 \text{ db}$$

$$\tau = 1 \text{ ms}$$

$$f_o = 10 \text{ gH}$$

$$N = 50$$

one finds from (40) that

$$R \cong .004$$

Such a system would be capable of about a 2 degree temperature resolution with a bandwidth of about 10 percent.

Figure 7 shows information as given by (40) plotted for various values of the resolution factor R, in the case of a uniformly illuminated array of 51 elements. The curves show definite maxima where the compromise between conflicting antenna and receiver bandwidth effects occur.

In order to investigate the effects of tapering the illumination of the array, a "cosine-squared-on-a-pedestal" distribution function of the form

$$a_n = H + \sin^2 \frac{(n-1)\pi}{50} \qquad (42)$$

was used to determine the spatial frequency power gain amplitudes, G_n, in (40). Curves of information as a function of bandwidth and resolution were obtained similar to those of Figure 6 but for various values of the pedestal height H. The values of pedestal height and bandwidth which maximized the information were obtained from these curves and these results are shown in Figures 8 and 9.

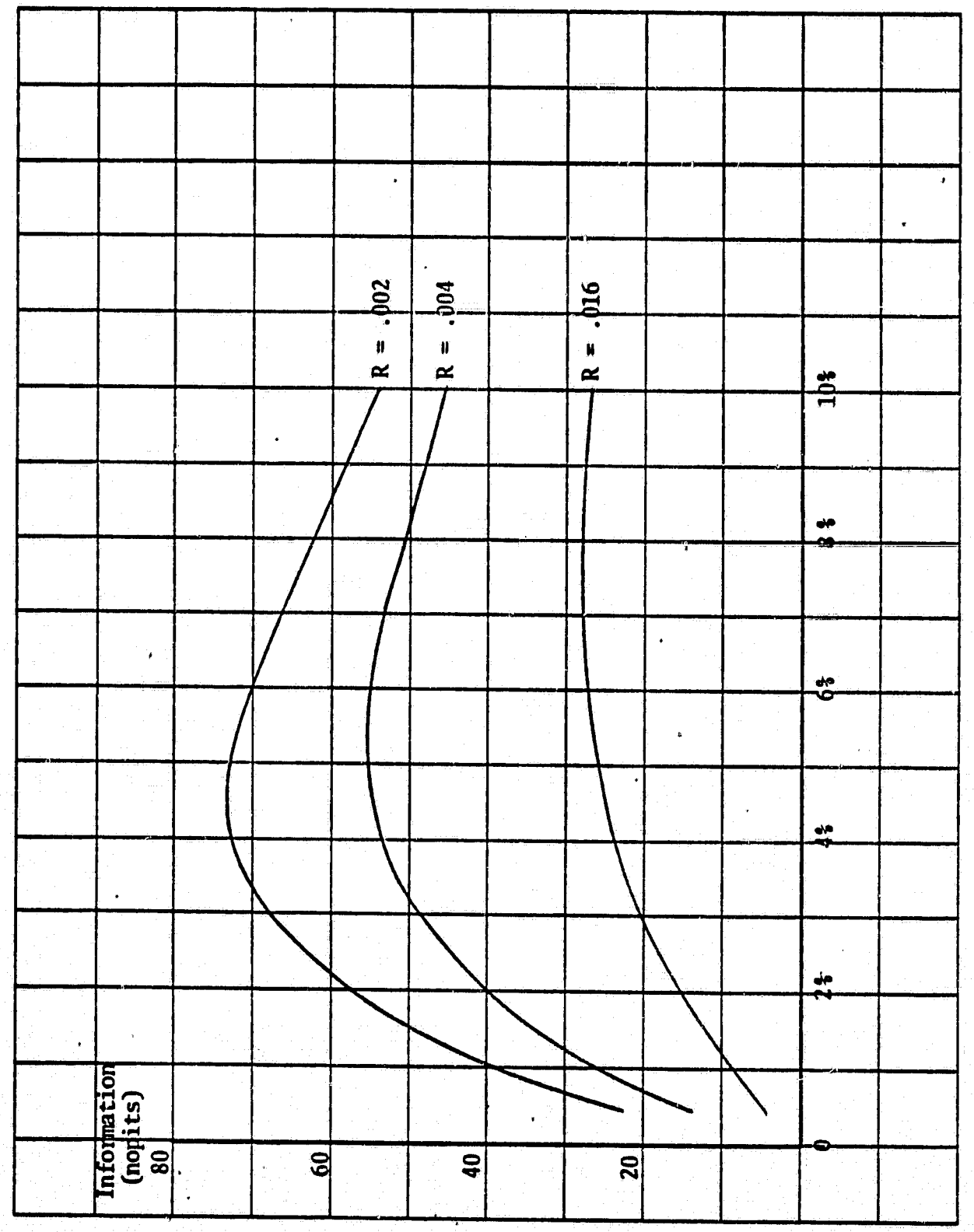


Figure 7. Information Vs. Bandwidth for a 51 Element Uniform Array

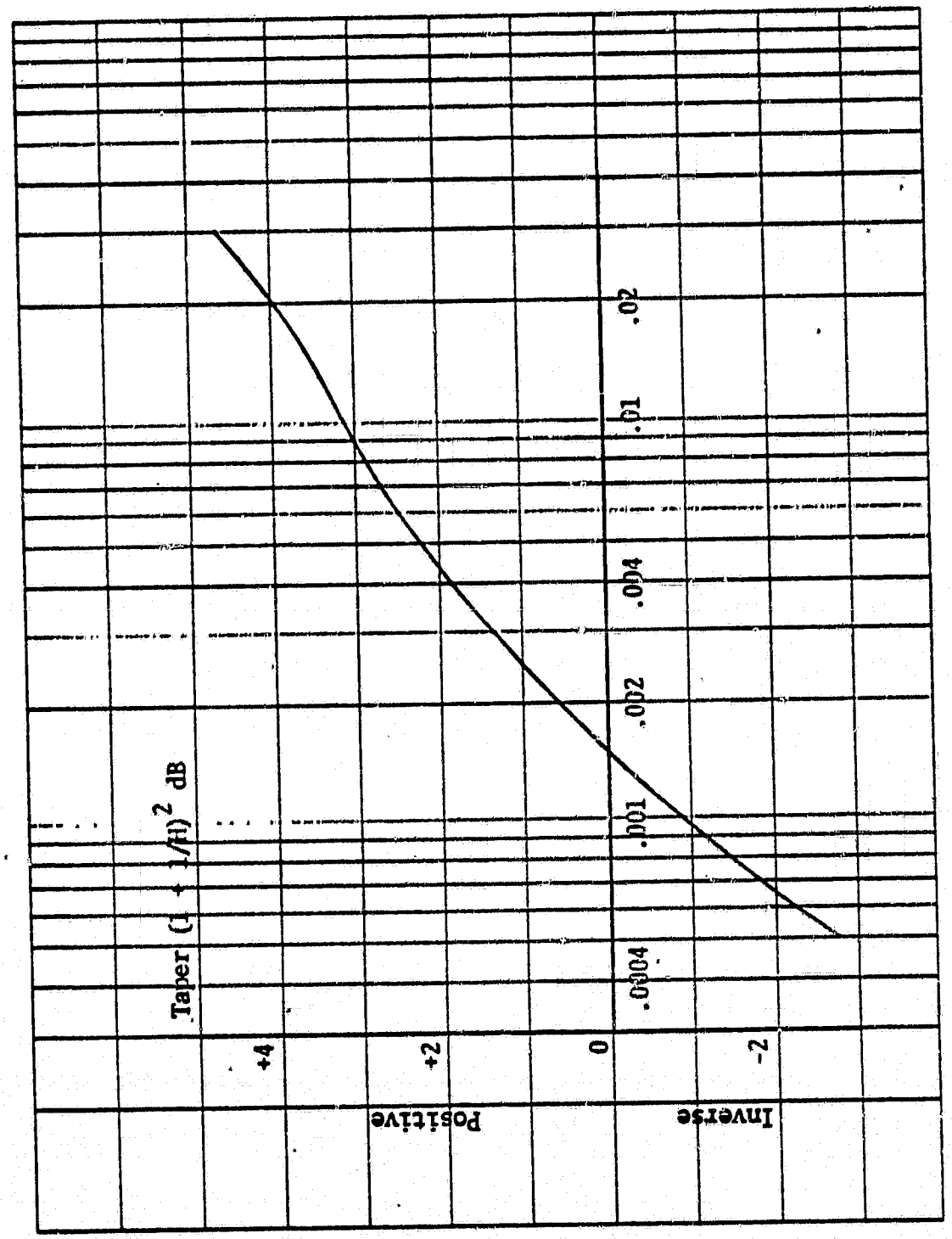


Figure 8. Illumination Taper as a Function of Resolution Factor for Maximum Information Transfer

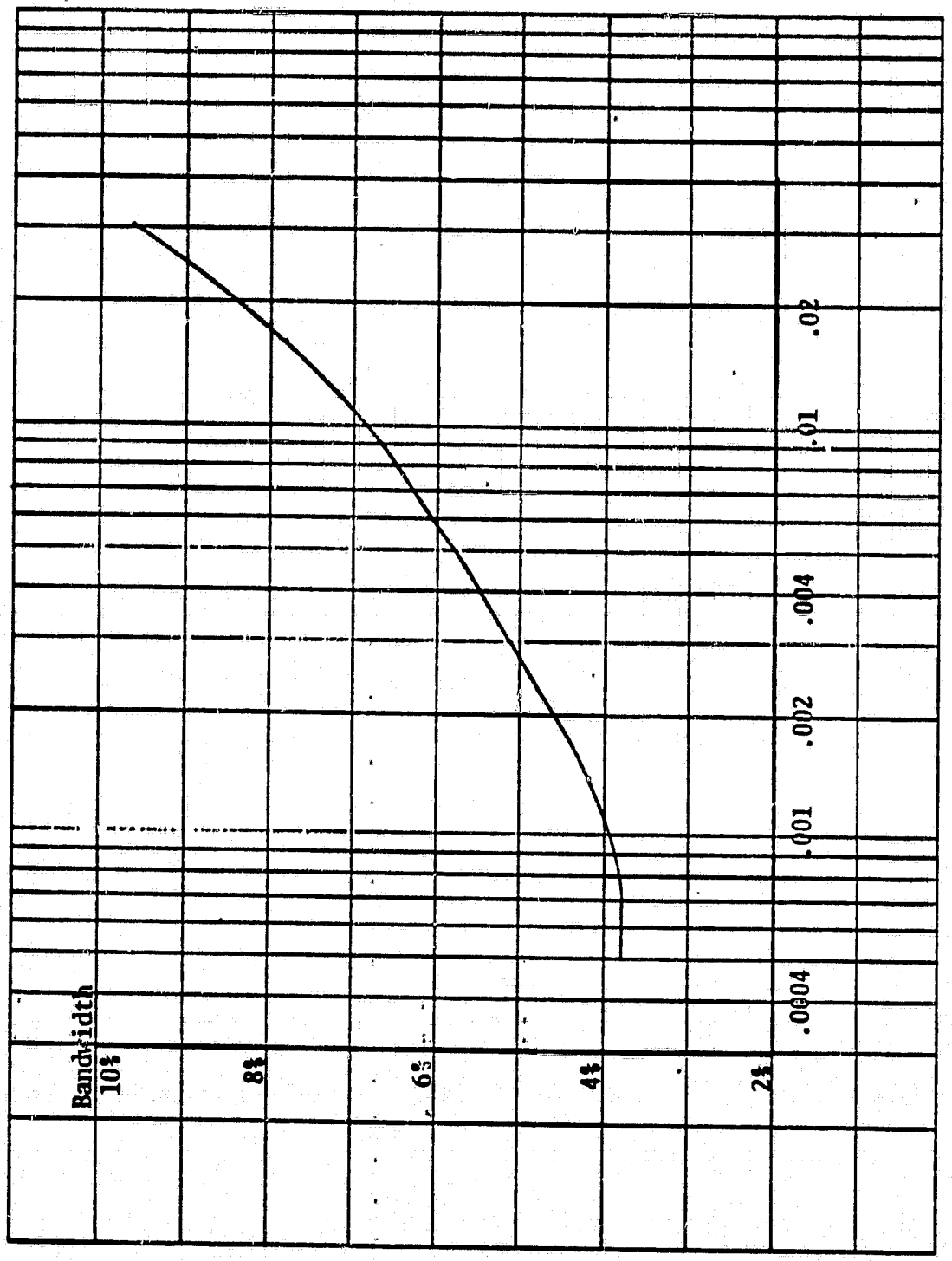


Figure 9. Microwave Bandwidth Vs. Resolution Factor for Maximum Information Transfer

IV. RECEPTION OF THE MICROWAVE SIGNAL (Continued)

The pedestal height for maximum information is indicated in Figure 8 in terms of the db taper from the center to the ends of the array. When R is small and the system is capable of good temperature resolution (low receiver noise temperature, long integration time, etc.) the optimum bandwidth is narrow (Figure 9) and the optimum illumination is slightly inversely tapered. In this case the high spatial frequencies passed by the antenna are sufficiently above the receiver noise so that they may be emphasized at the expense of the low frequencies to produce a significant increase in information. The emphasis is achieved by the inverse antenna taper (high frequency weighting) and narrow bandwidth (high frequency antenna noise reduction). For large values of R, antenna effects are swamped out by poor noise integration and information is maximized by large bandwidth and positive taper (low spatial frequency emphasis).

It is interesting to note that over a wide range of the resolution parameter, R, the illumination for optimum information transfer does not differ from uniform by more than a few db. Thus, despite sidelobe considerations, maximum information transfer roughly corresponds to maximum gain. (A discussion of this result will follow in the next section.) In any case, once the variance in the temperature profile to be investigated has been estimated, the curves of Figure 8 and 9 can be used to select the optimum illumination taper and bandwidth for the array.

V. COMPARISON WITH CONTEMPORARY DESIGN

The results of the previous analysis have indicated that maximum information transfer, in the classical sense, requires a particular choice of microwave bandwidth and antenna taper, dependent upon the receiver being used and the temperature profile under investigation. These design criteria are to be contrasted with design concepts based on goals which are more subjective than information transfer. For example, it seems intuitively plausible that a reduction in sidelobe level automatically betters the information gathering capabilities of the antenna as it reduces the amount of spurious power received by the system. Such a statement, however, tacitly assumes that only energy received by the main beam is of any value in diagnosing the target profile. Actually, all the energy at the output of the antenna could conceivably be used to describe the scanned profile. In target recognition, for example, the entire spatial frequency spectrum at the system output could be cross-correlated against a synthetic target spectrum using all the available information to deduce the presents of the target, not merely that information coming from the main beam.

When only the main beam at the antenna pattern is considered valuable, the natural tendency is to choose the front end bandwidth as large as possible to provide maximum temperature resolution in the receiver. Large bandwidths, however, tend to fill in the sidelobe variations in the pattern and destroy some of the information which the antenna could obtain with a narrow bandwidth.

The actual loss in information due to tapering is shown in Figure 10 where the illumination given by (42) was used in (40) to calculate maximum information for various pedestal heights. As previously mentioned, maximum information occurs near a uniform illumination where $1/H = 0$. As shown, tapering in either direction which both increases and decreases the sidelobe level, causes a reduction in the maximum information which can be passed by the system.

Although the numerical value of the information lost may be open to some subjective interpretation, the objective statement of the information at the system output can be used to compare one system against another to predict levels of performance. This idea is illustrated in Figure 11. Where systems transferring the same amount of information but with different illumination tapers are compared. This figure dramatically demonstrates the cost of sidelobe reduction by illumination tapering. A resolution factor of $R_{max} = .004$ was chosen which requires a pedestal height of $H = 4$, ($1/H = .25$, -14 db sidelobe level) to produce a maximum information of 55.7 nepits at a bandwidth of 5.5%. Now as H is increased and the illumination becomes more tapered, the

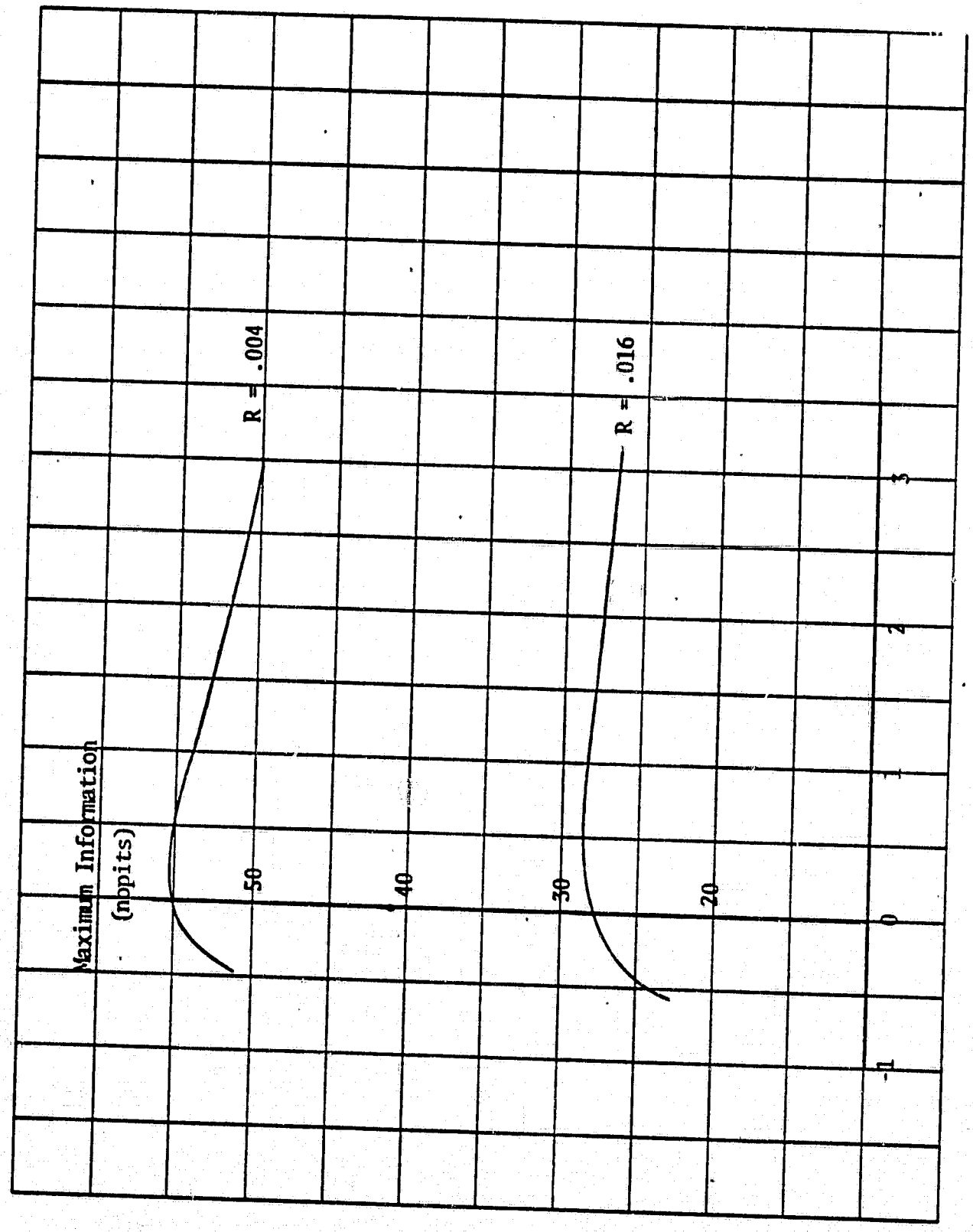


Figure 10. Maximum Information Vs. Inverse Pedestal Height for a 51 Element Array

V. COMPARISON WITH CONTEMPORARY DESIGN (Continued)

resolution faction of the system must be decreased to keep the maximum information at 55.7 nepits. The ratio

$$R/R_{max}$$

is shown on curve (b) as a function of inverse pedestal height and is a measure of the resolution efficiency of the system. Thus, at a pedestal height of H = .1, (1/H = 10, -41 db sidelobe level) the resolution efficiency is .65 which means the integration time must be increased by a factor of 1/.65 = 1.54 or the receiver noise figure decreased by .94 db to maintain an output of 55.7 nepits. For the sake of comparison, curve (a) gives the monochromatic aperture efficiency and serves to illustrate the reduction in monochromatic gain due to taper.

Finally, curve (c) illustrates the reduction in bandwidth which can be made if the optimum taper is used and the output kept at 55.7 nepits. As indicated, the system with the pedestal height of .1 requires about 3 times the bandwidth as the system with a pedestal height of 4 to produce the same information at the system output. The curves of Figure 11 serve to show that severe tapering of the array equivalently produces a significant increase in receiver noise figure or amount of microwave bandwidth needed to produce a given amount of output information.

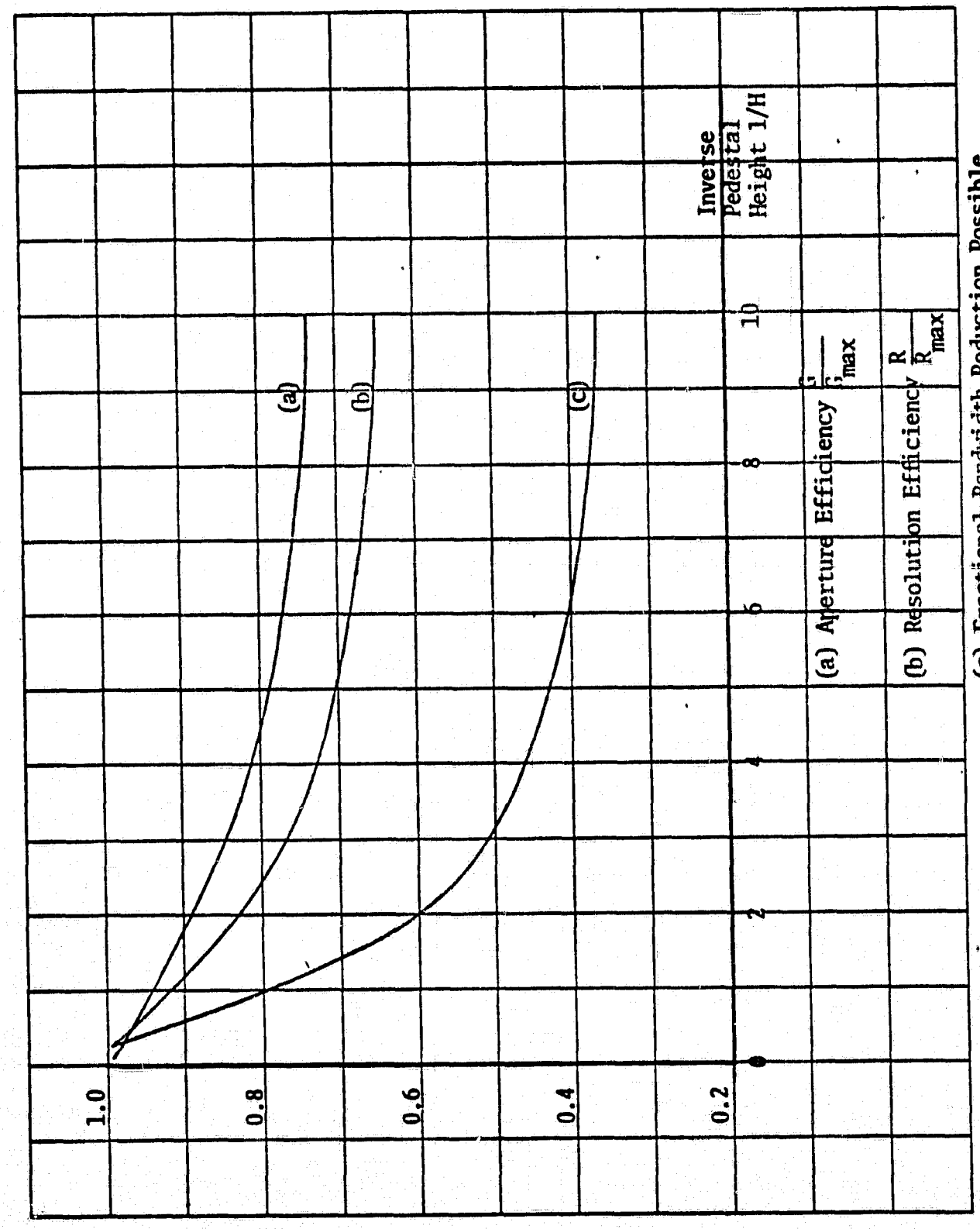


Figure 11. Variation of Antenna and System Performance with Antenna Taper

APPENDIX E

THE VAN CITTERT-ZERNIKE THEOREM

From PRINCIPLES OF OPTICS, Born and Wolf
PERGAMON PRESS (1964)

and P_2 in the wave-field is characterised by J_{12} rather than by $\Gamma_{12}(\tau)$, i.e. by a quantity which depends on the positions of the two points, but not on the time-difference τ. It follows from (10a) that, within the accuracy of this elementary theory,

$$|\gamma_{12}(\tau)| = |\mu_{12}| \qquad (13)$$

so that $|\mu_{12}|$ $(0 \leqslant |\mu_{12}| \leqslant 1)$ represents the degree of coherence of the vibrations at P_1 and P_2; and we see from (11) that the phase β_{12} of μ_{12} represents their effective phase difference. μ_{12}, just like $\gamma_{12}(\tau)$ of which it is a special case, is usually called *the complex degree of coherence* (sometimes *the complex coherence factor*), and J_{12} is called *the mutual intensity*.

10.4.2 Calculation of mutual intensity and degree of coherence for light from an extended incoherent quasi-monochromatic source

(a) *The Van Cittert–Zernike theorem*

We shall now determine the mutual intensity J_{12} and the complex degree of coherence μ_{12} for points P_1 and P_2 on a screen $\mathscr{A}$ illuminated by an extended quasi-monochromatic primary source σ. For simplicity σ will be taken to be a portion of a plane parallel to $\mathscr{A}$, and we will assume that the medium between the source and the screen is homogeneous. We also assume that the linear dimensions of σ are small

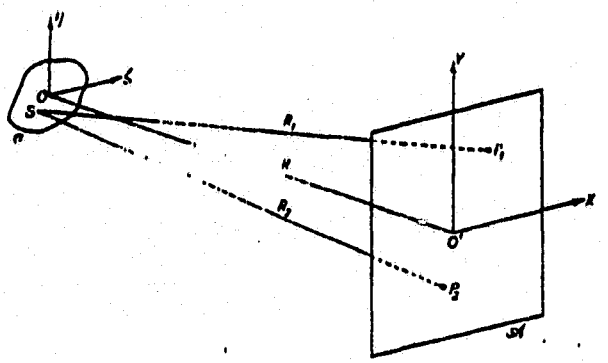


Fig. 10.3. Illustrating the VAN CITTERT–ZERNIKE theorem.

compared to the distance OO' between the source and the screen (Fig. 10.3), and that the angles between OO' and the line joining a typical source point S to P_1 and P_2 are small.

Imagine the source to be divided into elements $d\sigma_1$, $d\sigma_2$, . . . centred on points S_1, S_2, . . ., of linear dimensions small compared to the mean wavelength $\bar{\lambda}$. If $V_{m1}(t)$ and $V_{m2}(t)$ are the complex disturbances at P_1 and P_2 due to the element $d\sigma_m$, the total disturbances at these points are

$$V_1(t) = \sum_m V_{m1}(t), \qquad V_2(t) = \sum_m V_{m2}(t). \qquad (14)$$

Hence

$$J(P_1,P_2) = \langle V_1(t)V_2^*(t)\rangle = \sum_m \langle V_{m1}(t)V_{m2}^*(t)\rangle + \sum_m \sum_{n \neq m} \langle V_{m1}(t)V_{n2}^*(t)\rangle. \qquad (15)$$

Now the light vibrations arising from different elements of the source may be assumed to be statistically independent (mutually incoherent), and of zero mean value, so that*

$$\langle V_{m1}(t)V^*_{n2}(t)\rangle = \langle V_{m1}(t)\rangle\langle V^*_{n2}(t)\rangle = 0 \quad \text{when} \quad m \neq n. \tag{16}$$

If R_{m1} and R_{m2} are the distances of P_1 and P_2 from the source element $d\sigma_m$, then

$$V_{m1}(t) = A_m\left(t - \frac{R_{m1}}{v}\right)\frac{e^{-2\pi i\nu(t-R_{m1}/v)}}{R_{m1}}, \qquad V_{m2}(t) = A_m\left(t - \frac{R_{m2}}{v}\right)\frac{e^{-2\pi i\nu(t-R_{m2}/v)}}{R_{m2}}, \tag{17}$$

where $|A_m|$ charactorizes the strength and arg A_m the phase of the radiation from the mth element,† and v is the velocity of light in the medium between the source and the screen. Hence

$$\begin{aligned}\langle V_{m1}(t)V^*_{m2}(t)\rangle &= \left\langle A_m\left(t - \frac{R_{m1}}{v}\right)A^*_m\left(t - \frac{R_{m2}}{v}\right)\right\rangle \frac{e^{2\pi i\nu(R_{m1}-R_{m2})/v}}{R_{m1}R_{m2}} \\ &= \left\langle A_m(t)A^*_m\left(t - \frac{R_{m2}-R_{m1}}{v}\right)\right\rangle \frac{e^{2\pi i\nu(R_{m1}-R_{m2})/v}}{R_{m1}R_{m2}}.\end{aligned} \tag{18}$$

If the path difference $R_{m2} - R_{m1}$ is small compared to the coherence length of the light we may neglect the retardation $(R_{m2} - R_{m1})/v$ in the argument of A^*_m, and we obtain from (15), (16), and (18)

$$J(P_1,P_2) = \sum_m \langle A_m(t)A^*_m(t)\rangle \frac{e^{2\pi i\nu(R_{m1}-R_{m2})/v}}{R_{m1}R_{m2}}. \tag{19}$$

The quantity $\langle A_m(t)A^*_m(t)\rangle$ charactorizes the intensity of the radiation from the source element $d\sigma_m$. In any practical case the total number of the source elements may be assumed to be so large, that we may regard the source to be effectively continuous. Denoting by $I(S)$ the intensity per unit area of the source, i.e. $I(S_m)d\sigma_m = \langle A_m(t)A^*_m(t)\rangle$, (19) becomes‡

$$J(P_1,P_2) = \int_\sigma I(S)\frac{e^{ik(R_1-R_2)}}{R_1R_2}\,dS, \tag{20}$$

where R_1 and R_2 denote the distances between a typical source point S and the points P_1 and P_2, and $k = 2\pi\nu/v = 2\pi/\lambda$ is the wave number in the medium. The complex degree of coherence $\mu(P_1,P_2)$ is, according to (20) and (9b), given by

$$\mu(P_1,P_2) = \frac{1}{\sqrt{I(P_1)}\sqrt{I(P_2)}}\int_\sigma I(S)\frac{e^{ik(R_1-R_2)}}{R_1R_2}\,dS, \tag{21}$$

where

$$I(P_1) = J(P_1,P_1) = \int_\sigma \frac{I(S)}{R_1^2}\,dS, \qquad I(P_2) = J(P_2,P_2) = \int_\sigma \frac{I(S)}{R_2^2}\,dS, \tag{21a}$$

are the intensities at P_1 and P_2.

* Incoherence always implies a finite (though not necessarily wide) spectral range, and equation (16) is, in fact, not valid for the idealized case of strictly monochromatic light. For monochromatic light one has $V_{m1}(t) = U_{m1}e^{-2\pi i\nu t}$, $V_{n2}(t) = U_{n2}e^{-2\pi i\nu t}$, where U_{m1} and U_{n2} are independent of time, so that $\langle V_{m1}(t)V^*_{n2}(t)\rangle = U_{m1}U^*_{n2}$ and this quantity is in general different from zero.

† In general A_m also depends on direction, but for simplicity we neglect this dependence.

‡ From now on we shall frequently use the notation $dS, dP_1, \ldots$ for surface elements centred on the points $S, P_1, \ldots$.

We note that the integral (21) is the same as that which occurs in quite a different connection; namely in the calculation, on the basis of the HUYGENS–FRESNEL principle, of the complex disturbance in the diffraction pattern arising from diffraction of a spherical wave on an aperture in an opaque screen. More precisely, (21) implies that *the complex degree of coherence, which describes the correlation of vibrations at a fixed point P_2 and a variable point P_1 in a plane illuminated by an extended quasi-monochromatic primary source, is equal to the normalized complex amplitude at the corresponding point P_1 in a certain diffraction pattern, centred on P_2. This pattern would be obtained on replacing the source by a diffracting aperture of the same size and shape as the source, and on filling it with a spherical wave converging to P_2, the amplitude distribution over the wave-front in the aperture being proportional to the intensity distribution across the source.* This result was first established by VAN CITTERT* and later in a simpler way by ZERNIKE.† We shall refer to it as the *van Cittert–Zernike theorem*.

In most applications the intensity $I(S)$ may be assumed to be independent of the position of S on the surface (uniform intensity). The corresponding diffraction problem is then that of diffraction of a spherical wave of uniform amplitude by an aperture of the same size and shape as the source.

Let (ξ,η) be the coordinates of a typical source point S, referred to axes at O, and let (X_1,Y_1) and (X_2,Y_2) be the coordinates of P_1 and P_2 referred to parallel axes at O' (Fig. 10.3). Then, if R denotes the distance OO',

$$R_1^2 = (X_1-\xi)^2 + (Y_1-\eta)^2 + R^2,$$

so that

$$R_1 \simeq R + \frac{(X_1-\xi)^2+(Y_1-\eta)^2}{2R}. \tag{22}$$

Here only the leading terms in X_1/R, Y_1/R, ξ/R, and η/R have been retained. A strictly similar expression is obtained for R_2, so that

$$R_1 - R_2 \simeq \frac{(X_1^2+Y_1^2)-(X_2^2+Y_2^2)}{2R} - \frac{(X_1-X_2)\xi+(Y_1-Y_2)\eta}{R}. \tag{23}$$

In the denominator of the integrands in (20) and (21), R_1 and R_2 may to a good approximation be replaced by R. We also set

$$\frac{(X_1-X_2)}{R} = p, \qquad \frac{(Y_1-Y_2)}{R} = q, \tag{24}$$

$$\psi = \frac{k[(X_1^2+Y_1^2)-(X_2^2+Y_2^2)]}{2R}, \tag{25}$$

Then (21) reduces to

$$\mu_{12} = \frac{e^{i\psi}\iint_{\sigma} I(\xi,\eta)e^{-ik(p\xi+q\eta)}\,d\xi d\eta}{\iint_{\sigma} I(\xi,\eta)\,d\xi d\eta}. \tag{26}$$

Hence *if the linear dimensions of the source and the distance between P_1 and P_2 are small compared to the distance of these points from the source, the degree of coherence $|\mu_{12}|$ is equal to the absolute value of the normalized Fourier transform of the intensity function of the source.*

* P. H. VAN CITTERT, *Physica*, **1** (1934), 201.
† F. ZERNIKE, *Physica*, **5** (1938), 785.

The quantity ψ defined by (25) has a simple interpretation. According to (23) it represents the phase difference $2\pi(OP_1 - OP_2)/\bar{\lambda}$, and may evidently be neglected when

$$OP_1 - OP_2 \ll \bar{\lambda}. \tag{27}$$

For a uniform circular source of radius ρ with its centre at O, (26) gives on integration (cf. § 8.5.2)

$$\mu_{12} = \left(\frac{2J_1(v)}{v}\right)e^{i\psi}, \tag{28}$$

where

$$\left.\begin{aligned} v &= k\rho\sqrt{p^2+q^2} = \frac{2\pi}{\bar{\lambda}}\frac{\rho}{R}\sqrt{(X_1-X_2)^2+(Y_1-Y_2)^2}, \\ \psi &= \frac{2\pi}{\bar{\lambda}}\left[\frac{(X_1^2+Y_1^2)-(X_2^2+Y_2^2)}{2R}\right] \end{aligned}\right\} \tag{29}$$

J_1 being the BESSEL function of the first kind and first order.* According to § 8.5.2, $|2J_1(v)/v|$ decreases steadily from the value unity when $v = 0$ to the value zero when $v = 3{\cdot}83$; thus as the points P_1 and P_2 are separated more and more, the degree of coherence steadily decreases and there is complete incoherence when P_1 and P_2 are separated by the distance

$$\overline{P_1P_2} = \sqrt{(X_1-X_2)^2+(Y_1-Y_2)^2} = \frac{0{\cdot}61R\bar{\lambda}}{\rho}. \tag{30}$$

A further increase in v re-introduces a small amount of coherence, but the degree of coherence remains smaller than 0·14, and there is further complete incoherence for $v = 7{\cdot}02$. Since $J_1(v)$ changes sign as v passes through each zero of $J_1(v)$, the phase $\beta_{12} = \arg\mu_{12}$ changes there by π; in consequence the position of the bright and dark fringes are interchanged after each disappearance of the fringes.

The function $|2J_1(v)/v|$ decreases steadily from the value 1 for $v = 0$ to 0·88 when $v = 1$, i.e. when

$$P_1P_2 = \frac{0{\cdot}16R\bar{\lambda}}{\rho}. \tag{31}$$

Regarding a departure of 12 per cent from the ideal value unity as the maximum permissible departure, it follows that *the diameter of the circular area that is illuminated almost coherently by a quasi-monochromatic, uniform source of angular radius* $\alpha = \rho/R$ is† $0{\cdot}16\bar{\lambda}/\alpha$. This result is useful in estimating the size of a source needed in experiments on interference and diffraction.

As an example consider the size of the "area of coherence" around an arbitrary point on a screen illuminated directly by the sun. The angular diameter 2α which the sun's disc subtends on the surface of the earth is about $0° 32' \sim 0{\cdot}018$ radian. Hence, if the variation of brightness across the sun's disc is neglected, the diameter d of the area of coherence is approximately $0{\cdot}16\bar{\lambda}/0{\cdot}009 \sim 18\bar{\lambda}$. Taking the mean wavelength $\bar{\lambda}$ as $5{\cdot}5 \times 10^{-5}$ cm this gives $d \sim 0{\cdot}01$ mm.

---

* No confusion should arise from the fact that the symbol J is also used for the mutual intensity, as the latter always appears with two suffixes or with several arguments.

† As early as 1865, E. VERDET estimated that the diameter of the "circle of coherence" is somewhat smaller than $0{\cdot}5R\bar{\lambda}/\rho$. (*Ann. Scientif. de l'École Normale Supérieure*, **2** (1865), 291; also his *Leçons d'Optique Physique* (Paris, L'Imprimerie Impériale), **1** (1869), 106.)

APPENDIX F

MAPPING GEOMETRY

APPENDIX F

MAPPING GEOMETRY

System output parameters are defined in Table F-1, "Mapping Geometry Formulas." Inputs parameters and other symbols are defined in the symbol list below.

SYMBOL LIST

- O = Earth center
- P = Radiometer location
- QQ' = Location of pixel arc or edges of imager FOV
- G = Intersection of lines OP and QQ
- ρ = Orbital path length traversed per image buildup
- ϕ = Nadir angle of beam or of LOS to edge of imager FOV
- α = Maximum nadir angle
- B_T = Earth-center angle subtended by tile width
- H = Radiometer altitude above flat earth or orbital alt
- r_e = Earth radius = 6,370 km
- V_e = Orbital velocity at earth surface = 7.9 km/sec
- A = Area of a map tile
- ξ = Angle between tile diagonals and velocity vector
- N = Ratio of tile width and maximum possible width
- i = Tile ordinal number
- t_o = Time constant of phenomenon observed by radiometer
- V = Radiometry velocity (input only for flat-earth cases)
- θ = Azimuth angle traversed by scanning beam
- $\Delta\Omega$ = $\Delta\theta\Delta\phi$ = Solid angle subtended by pixel at radiometer
- D = Scanner antenna width (input only for scanners
- D_{EL} = Scanner antenna height (input only for scanners)
- k = Beam width parameter
- K = Radiometer performance parameter

INT(X) = First integer less than X

TABLE F-1. MAPPING GEOMETRY FORMULAS

Case Symbol	A Flat-Earth Scanner	B Orbital Scanner	C Flat-Earth Imager	D Orbital Imager (LEO)	E GEO Imager	Symbol Definitions
FOV	$\theta \times \phi$ Given	$\theta \times \phi$ Given	2ϕ Given	$2\tan^{-1}\left(\frac{\sin\eta}{\gamma - \cos\eta}\right)$	2η Given	Scan range and nadir angle of scanners or field-of-view of imagers
η	—	$\sin^{-1}(\gamma \sin\phi) - \phi$	—	$\frac{2\pi}{S[\mathrm{INT}(\pi/\phi_{max}) + 1]}$ (1)	Same as B	Earth-center subtending pixel arc radius or image disk radius
R	$H \sec\phi$	$r_e(\gamma - \cos\eta)\sec\phi$	$H \sec\phi$	$r_e(\gamma - \cos\eta)\sec\phi$	Same as B	Slant range to pixel arc or slant range to edge of FOV
r	$H \tan\phi$	$r_e\eta$	$H \tan\phi$	$r_e\eta$	$r_e\eta$	Radius of pixel arc or spherical cap within imager FOV
D	Given	Given	$\left(\frac{f_1}{6f_2}\right)^{2/7}$	$\left(\frac{f_1}{6f_2}\right)^{2/7}$ (2)	$\left(\frac{f_1}{6f_2}\right)^{2/7}$	Antenna width or Fourier disk diameter (aperture STH or series-fed array)
$\Delta\theta$	$k\left(\frac{\lambda}{D}\right)$	$k\left(\frac{\lambda}{D}\right)$	$k\left(\frac{\lambda}{2D}\right)$	$k\left(\frac{\lambda}{2D}\right)$	$k\left(\frac{\lambda}{2D}\right)$	Scanner azimuth beam width or imager resolution $\frac{1}{2}$ to Fourier disk
D_{EL}	Given	Given	$D\cos\phi$	$D\cos\phi$	$D\cos\phi$	Antenna height or Fourier disk diameter projected on R
$\Delta\phi$	$k\left(\frac{\lambda}{D_{EL}}\right)$	$k\left(\frac{\lambda}{D_{EL}}\right)$	$\Delta\theta\sec\phi$	$\Delta\theta\sec\phi$	$\Delta\theta\sec\phi$	Scanner elevation beam width or imager resolution at edge of FOV
AL	$H\,\Delta\theta\sec\phi$	$r_e(\gamma - \cos\eta)\,\Delta\theta\sec\phi$	$H\Delta\theta\sec\phi$	$r_e(\gamma - \cos\eta)\,\Delta\theta\sec\phi$	Same as D	Azimuth resolution at pixel arc or at edge of the FOV
Δr	$H\,\Delta\phi\sec^2\phi$	$R\,\Delta\phi\sec(\phi + \eta)$	$H\Delta\theta\sec^3\phi$	$R\,\Delta\theta\sec\phi\sec(\phi + \eta)$	Same as D	Radial resolution at pixel arc or at edge of the FOV
Δa	$H\,\Delta\Omega\sec^3\phi$	$R^2\,\Delta\Omega\sec(\phi + \eta)$	$H^2\Delta\theta^2\sec^4\phi$	$R^2\Delta\theta^2\sec\phi\sec(\phi + \eta)$	Same as D	Pixel area
M	$\frac{\theta}{\Delta\theta}$	$\frac{\theta}{\Delta\theta}$	$\frac{\pi D^2\,\overline{FOV}^2}{4}$	$\frac{\pi D^2\,\overline{FOV}^2}{4}$	$\frac{\pi D^2\,\overline{FOV}^2}{4}$	Number of pixels in arc or number of samples in Fourier disk
t^c	$\frac{H\Delta\phi\sec^2\phi}{V}$	$\frac{R\Delta\phi\sec(\phi + \eta)}{V}$	$\frac{2H\tan\phi}{V}$	$\frac{2\pi\gamma^{3/2}}{S[\mathrm{INT}(\pi/\phi\ \mathrm{MAX}) + 1]}\frac{r_e}{V_e}$ (1)	t_o	Time to scan a pixel arc or time to sample Fourier disk
τ	$\frac{H\Delta\Omega\sec^2\phi}{V\theta}$	$\frac{R\Delta\Omega\sec(\phi + \eta)}{V\theta}$	$\frac{\pi H\tan\phi}{\pi D^2\,\overline{FOV}^2 V}$	$\frac{4t_o}{\pi D^2\,\overline{FOV}^2}$	$\frac{4t_o}{\pi D^2\,\overline{FOV}^2}$	Radiometer integration time per pixel or per spectrum sample
ϕ	ϕ	$\phi + \eta$	ϕ	$\phi + \eta$	$\phi + \eta$	Angle-of-incidence at pixel arc or at edge of image disk
V	Given	$V_e/\sqrt{\gamma}$	Given	$V_e/\sqrt{\gamma}$	Zero	Radiometer velocity (orbital velocity for cases B and D)
D			$(f_1/4f_2)^{2/5}$	$(f_1/4f_2')^{2/5}$ (3)	$(f_1/4f_2)^{2/5}$	Fourier disk diameter (SPIRAD or true-time-delay antenna)

$\gamma = 1 + \frac{r_e}{H}$. (1) $\phi_{max} = \sin^{-1}\left(\frac{1}{\gamma}\right)$. (2) and (3) $f_1 = \Delta T^2/\sqrt{FOV}$. (2) $f_2 = \frac{\pi T_{SYS}^2\,\overline{FOV}^3}{2ft}$. (3) $f_2' = \frac{\pi T_{SYS}^2\,\overline{FOV}^2}{4P\,f_{EFF}}$. P = Σ bandwidth

F.1 FLAT-EARTH, PENCIL-BEAM SCANNER

Figure F-1 shows the mapping geometry in plan and elevation for this case (Case A).

From Figure F-1 we see that

$$\text{FOV} = \phi \text{ BY } \theta \tag{F-1}$$

$$R = H \text{ SEC } \phi \tag{F-2}$$

$$r = H \text{ TAN } \phi \tag{F-3}$$

$$\psi = \phi \tag{F-4}$$

$$\Delta\Omega = R\Delta\phi \text{ SEC}\phi = H\Delta\phi \text{ SEC}^2\phi \tag{F-5}$$

$$\Delta L = R\Delta\phi = H \text{ SEC}\phi \, \Delta\phi \tag{F-6}$$

$$\Delta a = \Delta r \, \Delta L = H^2\Delta\theta \, \Delta\phi \qquad \text{SEC}^3\phi = H^2 \, \Delta\Omega \, \text{SEC}^3\phi \tag{F-7}$$

$$t = \Delta r/V = H\Delta\phi \, \text{SEC}^2\phi \, V^{-1} \tag{F-8}$$

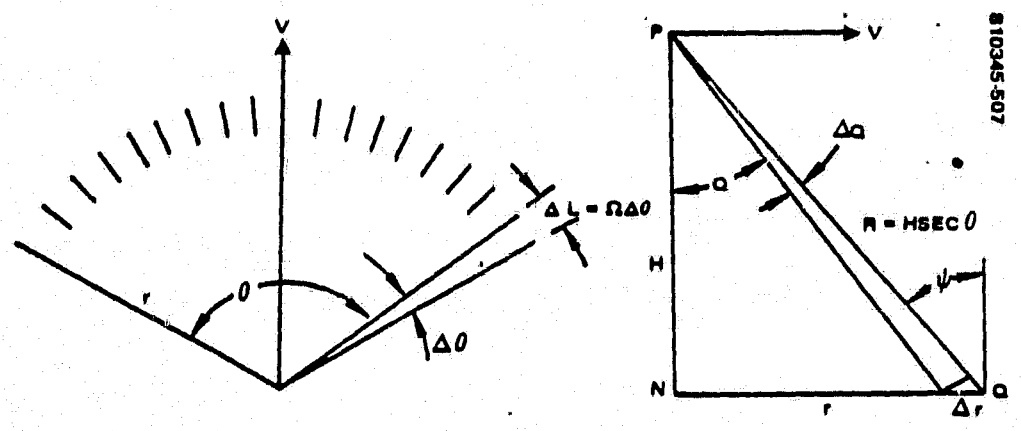


FIGURE F-1. FLAT-EARTH SCANNER MAPPING GEOMETRY

$$M = \theta/\Delta\theta \tag{F-9}$$

$$\tau = t/M = H\Delta\phi\ \Delta\theta\ SEC^2\phi\ \theta^{-1}\ V^{-1} = \frac{H\Delta\Omega\ SEC^2\phi}{V} \tag{F-10}$$

F.2 PENCIL-BEAM ORBITAL SCANNER

The geometry for this case (Case B) is presented in Figure F-2.

Using triangle OPQ and the law of sines

$$\psi = \sin^{-1}\left(\frac{r_e + H}{r_e}\sin\phi\right) = \sin^{-1}(y \sin\phi) \tag{F-11}$$

where ϕ has an assigned value and $y = 1 + H/r_e$.

From triangle PQT

$$\psi + (90° + \eta) + (90° - \psi) = 180° \tag{F-12}$$

Solving for η

$$\eta = \psi - \phi = \sin^{-1}(y \sin\phi) - \phi \tag{F-13}$$

From the right triangle GPQ

$$R = [H + (1 - \cos\eta)\ r_e]\ SEC\phi = r_e(y - \cos\eta)\ SEC\phi \tag{F-14}$$

The ground range to the pixel arc is the small circle NQ. Its length is

$$r = \eta r_e \tag{F-15}$$

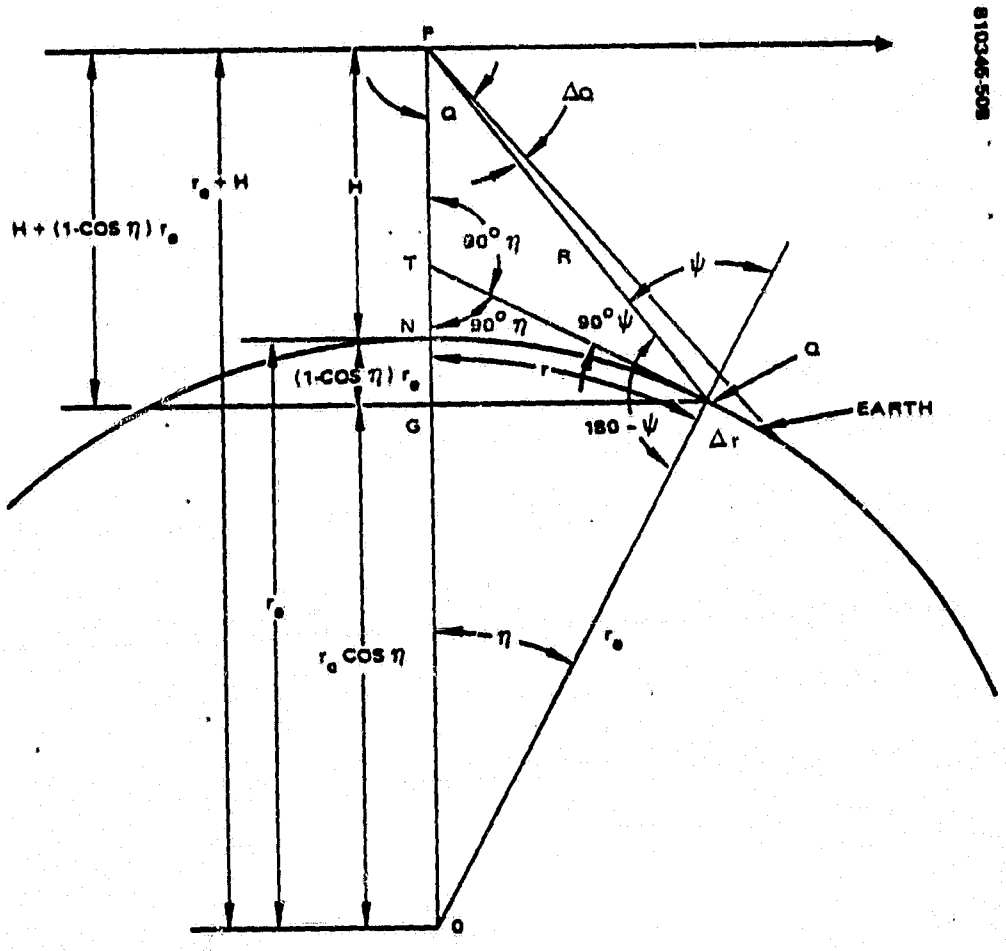


FIGURE F-2. PENCIL-BEAM ORBITAL SCANNER GEOMETRY

Then

$$\Delta r = R\Delta\phi \, \mathrm{SEC}\psi = r_e(1 - \cos\eta) \, \mathrm{SEC}\phi \, \mathrm{SEC}\psi \qquad \text{(F-16)}$$

$$\Delta L = R\Delta\theta = r_e(1 - \cos\eta) \, \Delta\theta \, \mathrm{SEC}\phi \qquad \text{(F-17)}$$

$$\Delta a = \Delta L \, \Delta r = r_e^2(1 - \cos\eta)^2 \, \Delta\Omega \, \mathrm{SEC}^2\phi \, \mathrm{SEC}\psi \qquad \text{(F-18)}$$

$$t = \Delta r/V = \frac{(R\Delta\phi \, \mathrm{SEC}\psi)}{V} = \frac{r_e}{V}(1 - \cos\eta) \, \Delta\phi \, \mathrm{SEC}\phi \, \mathrm{SEC}^2\psi \qquad \text{(F-19)}$$

$$M = \theta/\Delta\theta \qquad \text{(F-20)}$$

$$\tau = t/M = \frac{r_e}{V\theta}(1 - \cos\eta) \, \Delta\Omega \, \mathrm{SEC}\phi \, \mathrm{SEC}^2\psi \qquad \text{(F-21)}$$

F.3 FREQUENCY-PLANE, FLAT-EARTH IMAGER

Frequency plane imagers (Case C) measure the two-dimensional spatial frequency spectrum of the scene. The image is the two-dimensional Fourier transform of this spectrum. The "snap shot" version measures the spectrum of a FOV, forms the image and then repeats. The selected FOV must continue to be observed as the interferometer moves forward one image width. This implies the interferometer antennas must track one $\overline{\text{FOV}}$ during this movement and then snap to the next $\overline{\text{FOV}}$. The circular images must overlap to product a swath of continuous width x as shown in Figure F-3.

We wish to produce as great a mapped area per image buildup as possible. This mapped area is given by

$$A = (2r \sin\xi)(2r \cos\xi) = 4r^2 \sin\xi \cos\xi = 2r^2 \sin(2\xi) \qquad \text{(F-22)}$$

We wish to maximize A, so

$$\frac{dA}{d\xi} = 2r^2 \frac{d \sin(2\xi)}{d\xi} = 4r^2 \cos(2\xi) = 0 \qquad \text{(F-23)}$$

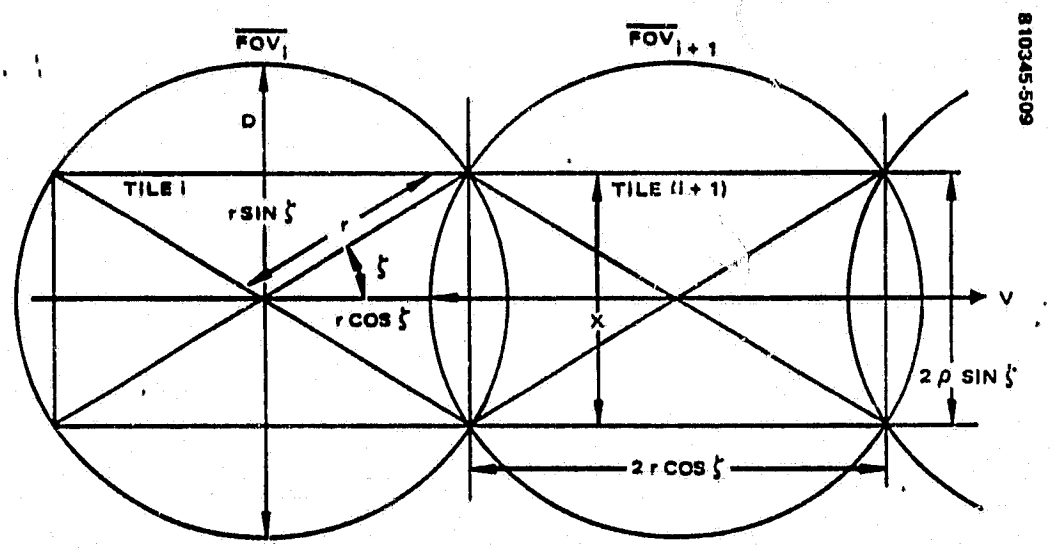


FIGURE F-3. OVERLAP GEOMETRY

In order for the derivative to be zero, ξ must be 45°. Then sin ξ times cos ξ is one half. So the mapped tile is a square. Its area is

$$A = 2r^2 \tag{F-24}$$

Parameters given for Case A, the flat-earth scanner, in (F-1) to (F-8) apply to Case C as well. However, when these equations are used to describe a Fourier imager $\Delta\theta$ and $\Delta\phi$ are no longer beamwidths. They are orthogonal angular resolutions for the final image generated by the Fourier transform operation.

Equations (F-9) and (F-10) are replaced by expressions for the number of spectrum samples which must be taken over the Fourier disk in order to prevent aliasing. From Appendix E, Equation (E-26), this number is

$$M = \frac{\pi}{4} D^2 \overline{FOV}^2 \tag{F-25}$$

where D is the diameter of the Fourier disk in wavelengths.

The time per image buildup is

$$t = \frac{2H \ TAN\phi}{V} \qquad \text{(F-26)}$$

So

$$\tau = \frac{t}{M} = \frac{8H \ TAN\phi}{\pi D^2 \ \overline{FOV}^2 \ V} \qquad \text{(F-27)}$$

also (F-24) becomes

$$A = 2H^2 \ TAN^2\phi \qquad \text{(F-28)}$$

F.4 FOURIER-TRANSFORM ORBITAL IMAGER (SNAP-SHOT MODE)

From right triangle OPT in Figure F-4

$$\phi_{MAX} = Sin^{-1}\left(\frac{r_e}{r_e + H}\right) = sin^{-1}\left(\frac{1}{y}\right) \qquad \text{(F-29)}$$

Let us find the largest angle at earth center which is less than β_{MAX} and which divides 360° integrally. Half this angle is

$$\phi_{EVEN} = \frac{\pi}{INT\left(\frac{\pi}{\phi_{MAX}}\right) + 1} \qquad \text{(F-30)}$$

Then the angle at 0 subtended by a tile-square width is

$$\beta T = \frac{2\phi_{EVEN}}{N} \qquad \text{(F-31)}$$

Also, in order to find the field-of-view we need to know η, half the angle at earth-center subtended by the circular image.

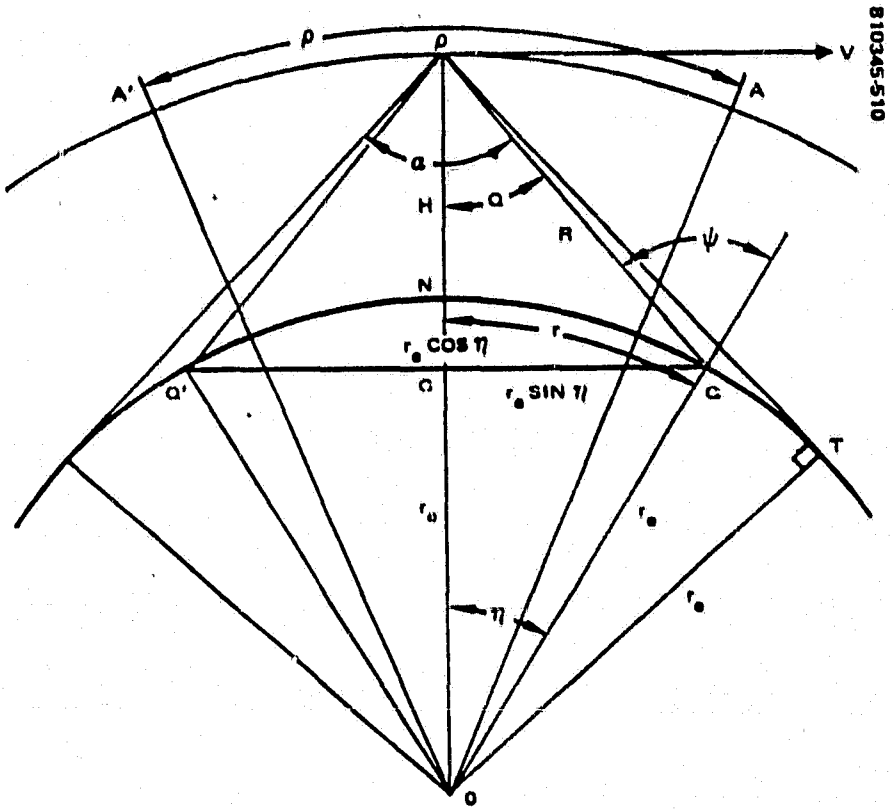


FIGURE F-4. FOURIER-TRANSFORM ORBITAL IMAGER

It is

$$\eta = \sqrt{2}\,\phi_T = \frac{\sqrt{2}\,\phi_{EVEN}}{N} \qquad \text{(F-32)}$$

The radius of the imaged cap shown in cross section between Q and Q' is

$$r = \eta r_e \qquad \text{(F-33)}$$

The orbital distance made good during image buildup, is

$$\rho = 2(r_e + H)\,\phi_T = r_e \frac{\pi\, y}{N\left[\mathrm{INT}\left(\frac{\pi}{\alpha_{MAX}}\right) + 1\right]} \qquad \text{(F-34)}$$

The time available to make an image is then

$$t = \frac{\rho}{V} = \frac{2\pi}{N\left[\text{INT}\left(\frac{\pi}{\phi_{MAX}}\right) + 1\right]}\left(\frac{r_e + H}{V_e}\right)^{3/2} = \frac{2\pi y^{3/2}}{N\left[\text{INT}\left(\frac{\pi}{\phi_{MAX}}\right) + 1\right]}\frac{r_e}{V_e} \quad \text{(F-35)}$$

where the value of V comes from Section F.5.

Since $\overline{OG} = r_e \cos\eta$ and $\overline{GQ} = r_e \sin\eta$ triangle GPQ gives

$$\phi = \text{TAN}^{-1}\left[\frac{r_e \sin\eta}{H + r_e(1 - \cos\eta)}\right] = \text{TAN}^{-1}\left(\frac{\sin\eta}{y - \cos\eta}\right) \quad \text{(F-36)}$$

and the slant range is

$$R = [H + (1 - \cos\eta)\, r_e]\ \text{SEC}\phi = r_e(y - \cos\eta) \quad \text{(F-37)}$$

The remainder of the output parameters are easily derived. They are given in Table F-1.

F.5 ORBITAL VELOCITY

For circular orbits, the orbital velocity ratio of two orbits is inversely proportional to the square of the earth radii ratio. Thus:

$$\frac{V_1}{V_2} = \sqrt{\frac{r_2}{r_1}} \quad \text{(F-38)}$$

As a special case let one orbit be at the earth surface. Then

$$V = V_e\sqrt{\frac{r_e}{r_e + H}} = V_e/\sqrt{y} \quad \text{(F-39)}$$

where $V_e = 7.9$ km/sec and $r_e = 6{,}370$ km.

F.6 SPINRAD ORBITAL GEOMETRY

We see that $\Delta\eta$ is given by $(\omega_o - \omega_e)t$, where ω_o is the satellite orbital rate and ω_e is the earth angular rate, so

$$\Delta\eta = 2\pi \frac{Vt}{CIR} - \omega_e t = 2\pi \frac{V_e t}{\sqrt{y}} \qquad \text{(F-40)}$$

now

$$CIR = \frac{r_e + H}{r_e} C_e = y\, C_e = \text{orbit circumference} \qquad \text{(F-41)}$$

$$T_e = \frac{C_e}{V_e} = 2\pi\, r_e = \text{earth-second pendulum period} \qquad \text{(F-42)}$$

then,

$$\Delta\eta = 2\pi \frac{V_e t}{y\, C_e t^{1/2}} - \omega_e t = \left(\frac{2\pi}{\tau_e\, y^{3/2}} - \omega_e\right) t \qquad \text{(F-43)}$$

Let

$$\eta' = \eta + \frac{\Delta\eta}{2},$$

So from (F-36)

$$\phi' = \phi + \Delta\phi = \frac{\tan^{-1} \eta'}{y - \cos \eta'} \qquad \text{(F-44)}$$

F.7 HP-97 MAPPING GEOMETRY PROGRAM

Table F-2, which follows, computes the pertinent geometrical parameters for orbital mapping using aperture synthesis or SPINRAD. The effective integration time, t_{eff}, for the SPINRAD case is also found. The first program, RTN/R/S, uses the orbital altitude H to find orbital velocity and period. It also finds the nadir angle of the LOS tangent to the earth's surface at the horizon, ϕ_{MAX}, and the corresponding zenith angle of the radius from earth center to the LOS tangent point, η_{MAX}.

The next two programs, LBLA and LBLB, are a pair. Given the nadir angle of the edge of the FOV, ϕ, they find the earth-center zenith angle or vice versa. They also find the angle of incidence of the line-of-sight at the edge of the FOV, ψ. If ϕ is known, LBLA enters it into the calculator. Then when LBLB is pressed, η and ψ are found and printed out. Conversely, if η is known, LBLB enters η after it is keyed in and "B" is pressed. Then pressing LBLA causes ϕ and ψ to be calculated and printed.

LBLC explores the limits of integration time in the SPINRAD case. A sequence of increasing integration times are set up and a set of performance parameters are printed for each value of t. First ϕ_{MAX} is printed, the ϕ' is outputted. This is the nadir angle of original FOV edge after satellite motion for timt t. If $\phi' > \phi_{MAX}$, a portion of the original terrain disk, whose spatial frequency spectrum is being measured, has gone below the horizon.

The program assumes that the FOV of the linear array normal to its length is increased to ensure that radiation from the total terrain disk continues to be received after time t. This decreases sensor efficiency, since the wider beam receives less power from the disk and replaces it with power from the surrounding area added to accommodate disk motion. This loss is expressed as a decrease in integration time. This decreased time, t_{eff}, is computed.

The satellite motion causes new terrain to replace old terrain at the edges of the original disk. These edge effects destroy a certain fraction of the image, $\Delta\eta/\eta$. This is printed out just before the effective integration time, t_{eff}.

Finally, the incidence angle of the LOS to the back and front edge of the disk will change during measurement. The rearward incidence angle will decrease and the forward angle increase. These angle changes, $\Delta\psi^+$ and $\Delta\psi^-$, are given. This permits the limitation of t, if these angle changes become excessive.

LBLD is concerned with the "snapshot" mode assumed for pure aperture synthesis. The number of square map sectors per orbit is entered. The program then gives the earth-center angle subtended by a map sector (tile). Next it gives the time available to measure the frequency spectrum of the disk within which the square is inscribed. Finally it gives the fields-of-view of the interferometer antennas.

TABLE F-2. APERTURE SYNTHESIS GEOMETRY PROGRAM

REGISTER	A	B	C	D	E					
VALUE	η	ϕ	r_u 6370	V_o 7.9	ω_o 7.27×10^{-5}					
REGISTER	0	1	2	3	4	5	6	7	8	9
VALUE	T_o 5066.32	y	a	ϕ_{MAX}	ψ	t	η'	ϕ'	$\lvert\Delta\eta\rvert$	ϕ_{EVEN}

H RTN/R/S		ϕ OR PRESS	η OR PRESS	L		N	η^{11}
f RAD	f $\sin^{-1}$	LBL A	LBL B	LBL C	RCL 6	LBL D	LBL E
PRTX H	PRTX η_{MAX}	STO H	STO A	STO 5	RCL 7	PRTX	↑
RCL C	E	P37	P37	PRTX $\pm$	+	X $\rightleftarrows$ Y	SIN
:	STO 3	GTO 0	GTO 0	RCL 3	RCL 4	:	X $\rightleftarrows$ Y
1	PRTX ϕ_{MAX}	RCL A	RCL B	PRTX ϕ_{MAX}	−	STO 6	COS
↑	1/x	E	SIN	RCL 5	PRTX $\Delta\psi$	2	RCL 1
STO 1 y	X $\rightleftarrows$ Y	STO B	RCL 1	RCL 2	RCL A	X	X $\rightleftarrows$ Y
1.5	:	RCL A	X	X	RCL H	PRTX β_y	−
y^x	I INT	+	$\sin^{-1}$	2	−	1.5	↑
RCL 0	1	STO 4	STO 4	↑	STO 6	RCL 1	f $\tan^{-2}$
X	+	RCL B	RCL B	f ABR	E	y^x	RTN
1/x	1/x	PRTX ϕ	−	STO H	RCL 6	X	ϕ^{11}
fπ	X $\rightleftarrows$ Y	RCL A	STO A	RCLA	+	RCL C	ϕ^{11} =
X	↑	PRTX η	RCL B	+	RCL 4	X	$\tan^{-1}\left(\frac{\sin\eta^{11}}{y-\cos\eta^{11}}\right)$
2	STO 9	RCL 4	PRTX ϕ	STO 6	−	RCL D	
X	PRTX ϕ_{EV}	PRTX ψ	RCL A	E	PRTX $\Delta\psi$	↑	
RCLE	PRT SPC	PRT SPC	PRTX η	STO 7	PRT SPC	PRTX t	
	PRT SPC	PRT SPC	RCL 4	PRTX ϕ'	RTN	RCL 6	
STO 2 a	RTN	LBL 0	PRTX ψ	RCL 8	OUTPUT	2	
RCL D	OUTPUT	RTN	PRT SPC	RCL A	t	$\sqrt{x}$	
RCL 1	Π_u		PRT SPC	↑	ϕ_{MAX}	X	
/x	V_o		LBL 0	PRTX $\Delta\eta/\eta$	ϕ'	E	
:	T_o		RTN	RCL B	$\Delta\eta/\eta$	PRTX ϕ	
PRTX V	η_{MAX}		OUTPUT	RCL 7	t_{eff}	PRT SPC	
1/x	ϕ_{MAX}		ϕ	:	$\Delta\psi^+$	RTN	
2	ϕ_{EVEN}		η	x^2	$\Delta\psi^-$	OUTPUT	
X	$\eta_{MAX} = \cos^{-1}\frac{1}{y}$		ψ	x^2		β_{TILE}	
fπ			$\psi = \sin^{-1}(y \sin\phi)$	RCL 5	$E = \left(\frac{\phi}{\phi'}\right)^4 t$	t	
X			$\eta = \psi - \phi$	X		ϕ	
RCL C				PRTX E			
X							
PRTX T_o							
RCL 1							
1/x							

*Either η or ϕ must be input as an initial value

APPENDIX G

PERFORMANCE OF APERTURE-SYNTHESIS IMAGERS

APPENDIX G

PERFORMANCE OF APERTURE-SYNTHESIS IMAGERS

G.1 IMAGE SPECTRUM

In Section 6, Part I, of Volume I, "Antennas as Spatial-Frequency Filters," it was shown that the MTF of an interferometer with baseline d is

$$H(F, d) = \Delta D^{-1} \left(1 - \frac{|F - d|}{\Delta d}\right), \quad d - \Delta d \leq F \leq d + \Delta d$$

$$= 0 , \; 0 \leq F \leq d - \Delta d \quad \text{OR} \quad d + \Delta d \leq F < \infty \qquad \text{(G-1)}$$

When $d \gg \Delta d$ (G-1) simplifies to

$$F(F, d) = \delta(F - d) \qquad \text{(G-2)}$$

In Appendix C, "Microwave Radiometer Image Statistics" it was found that the spectrum of the scene expressed as a video power density spectrum is

$$S(F) = \left(\frac{\Delta T^2}{4\pi \sqrt{FOV}}\right) \rho^{-5/2} = \theta \rho^{-5/2}, \; S(F) = G(F)\, G^{*}(F) \qquad \text{(G-3)}$$

Due to linear response to intensity, the RF power spectrum or video voltage spectrum of the radiometer is given by

$$V_I(F, d) = H(F, d)\, G(F) = \delta(F - d)\, G(F) = G(d) \qquad \text{(G-4)}$$

As the baseline length increases from zero to D, the voltage emerging from the radiometer $V_I(d)$ is a perfect replica of the spatial frequency spectrum of scene, S(F). Both S(F) and $V_I(d)$ are complex variables. The Fourier transform of V(d) is a replica of the projection of the radiant temperature distribution of the scene on the baseline.

The video power density spectrum of the radiometer is then

$$I(F) = V_I(F)\, V_I^*(F) = H^2(F, d)\, S(F) = \delta^2(F - D)\, S(F) = \theta\rho^{-5/2} \qquad \text{(G-5)}$$

G.2 BIAS ERROR

In order to find the total system error, we need to find the bias error, as well as error due to noise. Bias is an error which does not go to zero as measurement time becomes infinite. Since the interferometer reproduces G(F) perfectly out to $F = D^*$ and has zero response for $F > D$, the bias power is the integral of the power density spectrum from D to infinity.

That is

$$b^2 = \int_D^\infty \int_0^{2\pi} \theta\rho^{-5/2}(\rho\, d\rho\, d\psi) = -\frac{2\pi\theta}{\left(\frac{1}{2}\right)} \rho^{-1/2}\Bigg|_D^\infty = \left(\frac{\Delta T^2}{\sqrt{FOV}}\right) D^{-1/2} \qquad \text{(G-6)}$$

For the sake of compactness in latter calculations let

$$\frac{\Delta T^2}{\sqrt{FOV}} = f_1 \qquad \text{(G-7)}$$

So that

$$b^2 = f_1\, D^{-1/2} \qquad \text{(G-8)}$$

*Except for bias due to spectrum curvature, which is ignored for the time being.

The above calculation assumes there is no decorrelation of the signal due to differential time-of-flight (DTOF) of the signal to the baseline ends. In fact, such decorrelation occurs. As the baseline length increases $G(\rho)$, the spectrum including the effects of decorrelation, becomes smaller and smaller compared to the thre spectrum $G(F)$.

However, if the system bandwidth is kept small compared to the reciprocal of the DTOF, this bias error will be small and can be neglected. Furthermore, decorrelation also harms the performance of large filled-aperture pencil-beam scanners. Calculations for the effects of lateral decorrelation on a conventional antenna are given by Beran and Parrent.[1] The magnitudes of the effects do not appear to be much different from the aperture-synthesis case.

Therefore, comparison between the two system types is not harmed by neglecting bias due to decorrelation.

It should be pointed out that the spectrum power reduction vs. spatial frequency due to decorrelation is another MTF. It is multiplied into other transfer functions, such as the MTF due to image motion, to obtain the system MTF. Compensation for some of the performance degradation can be removed during image restoration. Correlation bias is treated in more detail in Appendix I.

In this analysis, a conservative value for the system bandwidth will be chosen. This bandwidth will be made one-fourth the reciprocal DTOF computed for the maximum baseline length D and for rays arriving from the edge of the FOV. The geometry for finding the DTOF is shown in Figure G-1 below.

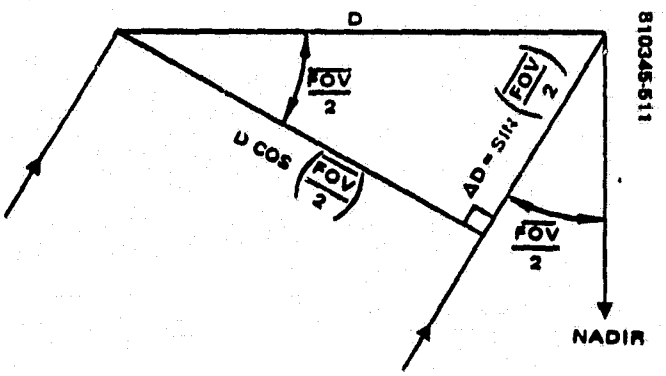


FIGURE G-1. GEOMETRY FOR CALCULATION OF DTOF

(1) M. Berran and G. Parrent, Theory of Partial Coherent, Prentice-Hall, 1964, Sec. 9.2, pp. 133-137.

We see that

$$\Delta D = D \sin\left(\frac{\overline{FOV}}{2}\right) \approx D\left(\frac{\overline{FOV}}{2}\right) \quad \text{for} \quad \overline{FOV} \ll 1 \tag{G-9}$$

So

$$\overline{DTOF} = \frac{D}{C}\frac{\overline{FOV}}{2} = \frac{D\,\overline{FOV}}{2f} = \frac{D\phi}{f} \tag{G-10}$$

NOTE:

$$C \text{ (wavelengths/sec)} = \frac{\overline{C}}{\lambda} = f(HZ) = \text{operating frequency}$$

Then

$$\Delta t = \frac{1}{4\,\overline{DTOF}} = \frac{f}{2D\,\overline{FOV}} = \frac{f}{4\phi D} \tag{G-11}$$

G.3 NUMBER OF SPATIAL-FREQUENCY SAMPLES

This sample number is needed to find the noise variance. For the smoothing time τ is the total observation time available per image buildup divided by the number of samples required. For the time being it is assumed that sample density is great enough to meet the Nyquist criterion. Then aliasing is prevented.

KO[2] gives the value of this Nyquist sample interval, ΔF. This value is

$$\Delta F = \overline{FOV}^{-1} \tag{G-12}$$

---

[2] H. KO, "Coherence Theory of Radio-Astronomical Measurements, IE^3 Trans., Vol. AP-15, No. 1, Sec. IV B, pp. 15-16, January 1967.

The area of a sample cell in the uv-plane is, therefore, $\Delta F^2 = \overline{FOV}^{-2}$. The area of the Fourier disk is $\pi D^2/4$. There the number of spectrum samples for uniform sample distribution is

$$Mv = \frac{\pi D^2}{4\Delta F^2} = \frac{\pi D^2 \overline{FOV}^2}{4} \qquad (G\text{-}13)$$

This would be the end of the sample number calculation, if we could always uniformly sample. However, if the interferometer antennas are moved rapidly back and forth along the baseline while the baseline slowly rotates, oversampling near the disk center results. This is the situation for radio-telescopes which use earth rotation to rotate the baseline. The situation is shown in Figure G-2 on the following page.

The correct number of samples on the half-circle rim in order to meet the Nyquist criterion is $(\pi D/\Delta F)$. Each rim sample corresponds to a baseline position with $(D/\Delta F)$ samples along its length. So the rotating-baseline sample number is

$$M_{ROT\ Baseline} = \frac{\pi D^2}{\Delta F^2} = \pi D^2 \overline{FOV}^2 \qquad (G\text{-}14)$$

The slowly rotating baseline strategy increases the sample number by four to one over the minimum necessary. The best sampling strategy is to rapidly rotate the baseline while its length slowly changes. The antennas move with constant velocity outward along spiral paths producing uniform sampling.

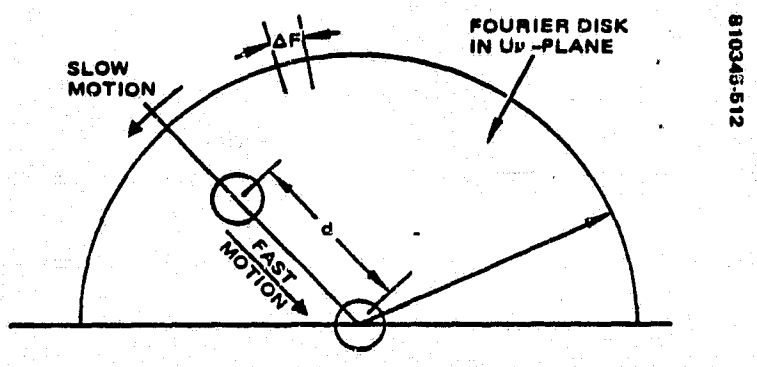


FIGURE G-2. GEOMETRY OF SPATIAL-FREQUENCY SPECTRUM SAMPLING (SLOWLY ROTATING BASELINE CASE)

We are now in a position to write τ as

$$\tau = \frac{t}{M_u} = \frac{4t}{\pi D^2 \overline{FOV}^2} \qquad \text{(G-15)}$$

where t is the observation time per image and M_u is number of samples for uniform sampling density.

G.4 NOISE ERROR

Next we must trace the generation of the video noise power density spectrum. This noise originates in the RF section. Let it have a power spectral density of N. This power density is converted to a video power density of N^2. The total video noise power, after bandwidth reduction, is δT^2. [Remember that we suppress (kB) in all expressions! So the true total noise power is $(k\delta TB)^2$.]

Since there are two antennas and the system is fully coherent $K = 1/\sqrt{2}$. Either the in-phase or quadrature channel (real or imaginary portion of the spectrum sample) will exhibit a noise variance of

$$\delta T_S^2 = \frac{T_{SYS}^2}{2\Delta f \tau} \qquad \text{(G-16)}$$

The transform process adds the I and Q noise powers to obtain the noise power in a pixel. The result is

$$\delta T_p^2 = \frac{T_{SYS}^2}{\Delta f \tau} \qquad \text{(G-17)}$$

Substituting Δf from (G-11) and from (G-15) we get

$$\delta T_p^2 = \left(\frac{\pi T_{SYS}^2 \overline{FOV}^3}{2ft}\right) D^3 = f_2 D^3 \qquad \text{(G-18)}$$

G.5 DETERMINATION OF OPTIMUM DISK DIAMETER AND MINIMUM TOTAL ERROR

Total system error is the sum of thermal and bias noise.

This sum is

$$\varepsilon^2(D) = \delta T_p^2 + b^2 = f_1 D^{-1/2} + f_2 D^3 \qquad \text{(G-19)}$$

This error must be differentiated, equated to zero and solved for D in order to find the optimum disk diameter.

$$\frac{d^2\varepsilon^2(D)}{dD} = -\frac{1}{2} f_1 D^{-3/2} + 3f_2 D^2 = 0 \qquad \text{(G-20)}$$

The result is

$$D = \left(\frac{f_1}{6f_2}\right)^{2/7} \quad \text{wavelengths} \qquad \text{(G-21)}$$

Substituting (G-18) into the expressions for ST_p^2 and b^2 we get

$$\begin{cases} b^2 = f_1\left(\dfrac{6f_2}{f_1}\right)^{1/7} = 6^{1/7} f_1^{6/7} f_2^{1/7} & (°K^2) \\ \delta T_p^2 = f_2\left(\dfrac{f_1}{6f_2}\right)^{6/7} = 6^{-6/7} f_1^{6/7} f_2^{1/7} = b^2/6 & (°K^2) \end{cases} \qquad \text{(G-22)}$$

Then the total mean-square error is

$$\varepsilon^2 = (6^{1/7} + 6^{-6/7}) f_1^{6/7} f_2^{1/7} \ (°K^2) = \frac{7}{6} b^2, \quad \varepsilon = \sqrt{\frac{7}{6}}\, b \qquad \text{(G-23)}$$

APPENDIX H

PERFORMANCE OF SERIES-FED-ARRAY PENCIL-BEAM SCANNERS

APPENDIX H

PERFORMANCE OF SERIES-FED-ARRAY PENCIL-BEAM SCANNERS

H.1 NOISE ERROR

Let the diameter of the circular aperture be D wavelengths. Then

$\Delta\theta = D^{-1}$ (rad) = half-power angular resolution

and

$$\Delta\Omega = \Delta\theta^2 = D^{-2} \text{ (ster)} = \text{pencil beam solid angle} \qquad \text{(H-1)}$$

Let

$$\Omega = \frac{\pi}{4}\overline{FOV}^2 \text{ (ster)} = \text{solid angle subtended by right circular cone scanned by pencil beam} \qquad \text{(H-2)}$$

Then

$$M = \frac{\Omega}{\Delta\Omega} = \frac{\pi D^2\,\overline{FOV}^2}{4} = \text{number of samples per scan cycle}$$

M = number of pixels in image (H-3)

And

$$\tau = \frac{t}{M} = \frac{4t}{\pi D^2 \overline{FOV}^2} = \text{radiometer integration time} \qquad \text{(H-4)}$$

Where t is the image formation time (frame time).

Now

$$\delta T_p^2 = \frac{K^2 T_{SYS}^2}{\Delta f \tau} \ (^\circ K^2) = \text{image noise variance} \qquad \text{(H-5)}$$

In the case of aperture synthesis it was noted that both of the interferometer antennas absorbed thermal power within the solid angle Ω without interfering with each other. However, both correlator channels must obtain power from the same pixel in the pencil beam scanner case. Therefore, the power per channel in the present case is half the power per channel for aperture synthesis. The radiometer measures only temperature. That means we are using only the I-channel output at zero time delay. For this case K = 1.

In Appendix F the question of system bandwidth was discussed. There it was pointed out that the bandwidth restrictions due to coherence loss are about the same for both filled-aperture arrays and synthetic-aperture cases, although some of the details differ. In the filled-aperture case, decorrelation uniformly reduces the entire SFD. In the synthetic-aperture case the reduction differed for different spatial frequencies. It increased with frequency.

As an illustration of these effects for a filled aperture, the loss in gain vs. coherence length for lateral coherence is shown in Figure H-1. The curve comes from Beran and Parrent's "Theory of Partial Coherence" referred to in Appendix C.

We are interested in longitudinal coherence, not lateral coherence. However, it is easy to see how to compute the effect we are concerned with. If the antenna is a series-fed array, the signal reaching the output port is delayed with respect to the signal entering the array nearest to the port. Delay increases with respect to the reference signal as entry into the array progresses further and further away from the exit port.

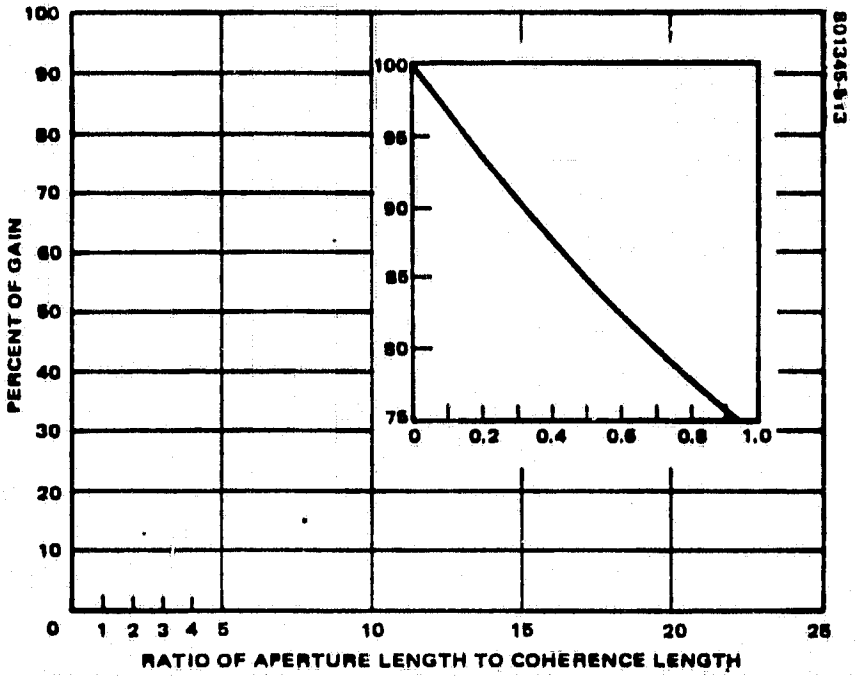


FIGURE H-1. CORRELATION LOSS VS NORMALIZED APERTURE SIZE

Let R(t) be the longitudinal coherence function of the signal, which is the transform of the RF spectrum. It may be regarded as part of a signal, $A^2(t)$, having constant power for all time delays. This signal consists of a coherent portion C(t) and a noncoherent portion I(t). The coherent portion is of the form R(t). Since power level is not a function of t, we have $R(t) + I^2(t) = A^2(t)$ as shown in Figure H-2.

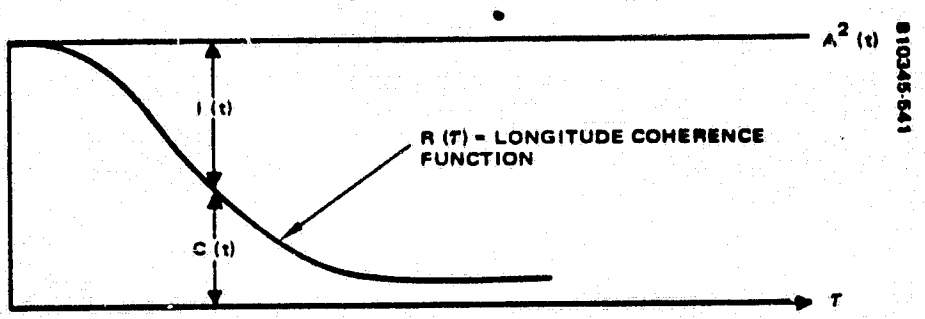


FIGURE H-2. COHERENT AND UNCORRELATED PORTIONS OF SIGNAL

The incoherent signal I(t) will not add amplitude-wise to form an output from the aperture. Its components for differing t's add power-wise to form a noncoherent result which is scattered out of the aperture and lost.

The total power in the signal, $\sqrt{R(t)}$, which adds coherently and is received, is

$$P_c = \int_0^\infty R(t)\, dt \qquad \text{(H-6)}$$

If there had been no decorrelation the power received would be

$$P_{Reference} = \int_0^\infty R(0)\, dt \qquad \text{(H-7)}$$

So the loss in gain vs t is

$$L = R(t)/R(0) = \phi(t) \qquad \text{(H-8a)}$$

The total fractional loss is therefore

$$L_{total} = \int_0^\infty \phi(f)\, dt \qquad \text{(H-8b)}$$

However, (H-8) is of the general form of the gain loss expressions encountered in the aperture synthesis cases.

So we again will use Equation 11, Appendix VI for system bandwidth.

$$\Delta f = \frac{C}{2D\ \overline{FOV}} \qquad \text{(H-9)}$$

Substituting τ in (H-4) and Δf in (H-9) into (H-5) with $K = 1$, we get

$$\delta T_p^2 = \left(\frac{\pi T_{SYS}^2 \overline{FOV}^3}{2ct}\right) D^3 = f_2 D^3 \qquad \text{(H-10)}$$

Note that (H-10) is the same as (G-18), Appendix G in the synthetic aperture case.

H.2 BIAS ERROR

In order to determine resolution bias note that D is the cutoff spatial frequency of a filled aperture.[1] Then the bias power is

$$b^2 = \int_D^{\infty} \int_0^{2\pi} 0\rho^{-5/2} \, (\rho \, d\rho \, d\psi) \; (^\circ K^2) \qquad \text{(H-11)}$$

This is identical to the integral giving the bias power in the synthetic aperture case in (6), Appendix VI.

So again

$$b^2 = f_1 D^{-1/2} \quad \text{where} \quad f_1 = \frac{\Delta T^2}{\overline{FOV}} \qquad \text{(H-12)}$$

In addition to (H-12) there is bias error because the MTF of a filled aperture is not rectangular. However, some of this bias can be removed by processing, so it will be neglected.

H.3 OPTIMUM APERTURE SIZE AND MINIMUM TOTAL ERROR

Since the error parameters, f_1 and f_2, are identical with their values for the aperture synthesis case, the optimum D and minimum error expressions are the same also. They are repeated here for convenience sake.

---

[1] J. Kraus, Radio Astronomy, Sec. 6-9, Eq. 6-65, pg. 170, McGraw-Hill, 1966.

$$D = \left(\frac{f_1}{6f_2}\right)^{2/7} \quad \text{wavelengths} \tag{H-13}$$

$$\begin{cases} b^2 = 6^{1/7} f_1^{6/7} f_2^{1/7} \quad (°K^2) \\ \delta T_p^2 = b^2/6 = 6^{-6/7} f_1^{6/7} f_2^{1/7} \quad (°K^2) \end{cases} \tag{H-14}$$

$$\varepsilon^2 = (6^{1/7} + 6^{-6/7}) f_1^{6/7} f_2^{1/7} \quad (°K^2) \tag{H-15}$$

APPENDIX I

RATE-DISTORTION CALCULATION OF ERRORS DUE TO UNDERSAMPLING

APPENDIX I

RATE-DISTORTION CALCULATION OF ERRORS DUE TO UNDERSAMPLING

I.1 MEAN-SQUARE SIGNAL ERROR VS INFORMATION RATE

The mean square error introduced into a signal as a function of the rate of source coding and information transmission through a communications channel is best expressed in parametric form. For this mean-square-error criteria and for a Gaussian source, this parametric representation is analytically tractable.[1, 2]

The analysis assumes optimum coding. If coding is optimum, then the distortion is an upper bound on the error for all sources with the same spectrum, but with other than Gaussian statistics.

I.1.1 One-Dimensional Case

As a warm-up exercise, let us do the one-dimensional case. From Berger, Equations (4.5.51) and (4.5.52), the parametric representation is

$$\begin{cases} d(\theta) = \dfrac{1}{2\pi}\displaystyle\int_{-\infty}^{+\infty} \mathrm{MIN}[\theta,\ \Phi(\omega)]\ d\omega & \text{(I-1)} \\ R(\theta) = \dfrac{1}{4\pi}\displaystyle\int_{-\infty}^{+\infty} \mathrm{MAX}\left[0,\ \ln \dfrac{\Phi(\omega)}{\theta}\right] d\omega & \text{(I-2)} \end{cases}$$

where ω is radian frequency.

---

(1) T. Berger, "R(D) for Gaussian Sources," pp. 106-130, Sec. 4.5, Rate Distortion Theory, Prentice-Hall, 1971.

(2) A. Viterbi and J. Omara, "Continuous Time Sources and Generalizations," Sec. 8.4.3, pp. 502-513, Principles of Digital Communication and Coding, McGraw-Hill, 1979.

Let the spectrum be one-sided and described by the relation

$$\Phi(\omega) = \omega^{-h} \tag{I-3}$$

If there are M lines in the image, we can write

$$\omega_{MAX} = M\omega_{MIN} \tag{I-4}$$

Furthermore, we can normalize the frequency range by assuming that ω_{min} is unity. This results because this spectrum is self-similar and self-affined. (See Appendix C, C.1.)

The above considerations permit us to write (I-1), (I-2) and the integral for the variance power, σ^2, in the forms.

$$\begin{cases} d(\theta) = \frac{1}{2\pi}\int_{0}^{\infty} MIN(0, \omega^{-h})\, d\omega = \frac{1}{2\pi}\int_{1}^{M} MIN(\theta, \omega^{-h})\, d\omega \\ R(\theta) = \frac{1}{4\pi}\int_{0}^{\infty} MAX\left(0, \ell n \frac{\omega - h}{\theta}\right) d\omega = \frac{1}{4\pi}\int_{1}^{M} MAX\left(0, \ell n \frac{\omega - h}{\theta}\right) d\omega \\ \sigma^2 = \frac{1}{2\pi}\int_{0}^{\infty} \omega^{-h}\, d\omega = \frac{1}{2\pi}\int_{1}^{M} \omega^{-h}\, d\omega \end{cases} \tag{I-5}$$

The variance, which is used to normalize d, becomes

$$\sigma^2 = \frac{1}{2\pi}\int_{1}^{M} \omega^{-h}\, d\omega = -\frac{\omega - (h-1)}{(h-1)}\Bigg|_{1}^{M} = \frac{1 - M - (h-1)}{2\pi(h-1)} \text{ (watts)} \tag{I-6}$$

Now θ becomes greater than ω^{-h} when

$$\theta = \omega_1^{-h} \quad , \quad 1 \leq \omega_1 \leq M \tag{I-7}$$

or

$$\omega_1 = \theta^{-1/h} \quad , \quad M^{-h} \leq \theta \leq 1 \tag{I-8}$$

So the distortion integral times 2π becomes

$$2\pi d(\theta) = \int_1^{\theta^{-1/h}} \theta d\omega + \int_{\theta^{-1/h}}^{M} \omega^{-h} d\omega = \theta(\theta^{-1/h} - 1) + \frac{\theta^{\frac{h-1}{h}} - M^{-(h-1)}}{(h-1)} \tag{I-9}$$

Therefore,

$$d(\theta) = \frac{h\theta^{\frac{h-1}{h}} - (h-1)\theta - M^{-(h-1)}}{2\pi(h-1)} \tag{I-10}$$

Normalize the error to obtain

$$\frac{d(\theta)}{\sigma^2} = h\theta^{\frac{h-1}{h}} - (h-1) \tag{I-11}$$

R(θ) is evaluated as follows

$$4\pi R(\theta) = \int_1^{\theta^{-1/h}} \ell n \frac{\omega^{-h}}{\theta} d\omega = \int_1^{\theta^{-1/h}} (-h \ell n\omega - \ell n\theta) d\omega$$

$$= h(\omega - \omega \ell n\omega) - \omega \ell n\theta \Big|_1^{\theta^{-1/h}}$$

$$= h\theta^{-1/h} + \theta^{-1/h} \ell n\theta - \theta^{-1/h} \ell n\theta - h + \ell n\theta \qquad \text{(I-12)}$$

so

$$R(\theta) = \frac{\ell n\theta - h(1 - \theta^{-1/h})}{4\pi} \text{ nats/UNF} \qquad \text{(I-13)}$$

where UNF means unit normalized frequency.

For θ = 1

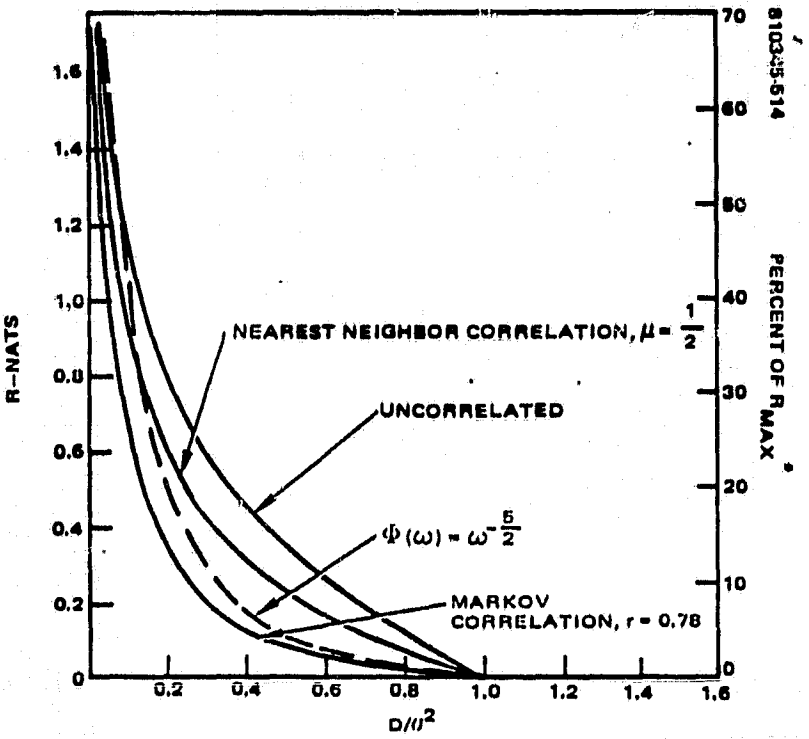
$$\frac{d(0)}{\sigma^2} = \frac{h - h + 1 - M^{-(h-1)}}{1 - M^{-(h-1)}} = 1 \quad \text{and} \quad R(\theta) = \frac{0 - h + h}{4\pi} = 0$$

These are the correct limits. Figure I-1 compares this case with other rate-distortion functions. Quantization also limits information rate as shown in Figure I-2.

I.1.2 Two-Dimensional Case

In the two-dimensional case the differential frequency element in the Fourier plane is of the form $(\rho d\phi d\rho)$, where ρ is radial radian frequency.

C-2



NOTE: COMPARISON OF MSE RATE DISTORTION FUNCTIONS FOR UNCORRELATED DATA, NEAREST NEIGHBOR ($\mu = \frac{1}{2}$), AND MARKOV CORRELATION (r = 0.78), AND $\omega^{-1/2}$ SPECTRUM

*% R_{MAX} SCALE APPLIES ONLY TO $\Psi = \omega^{-5/2}$ SIGNAL SPECTRUM CASE

FIGURE I-1. ONE-DIMENSIONAL CASES

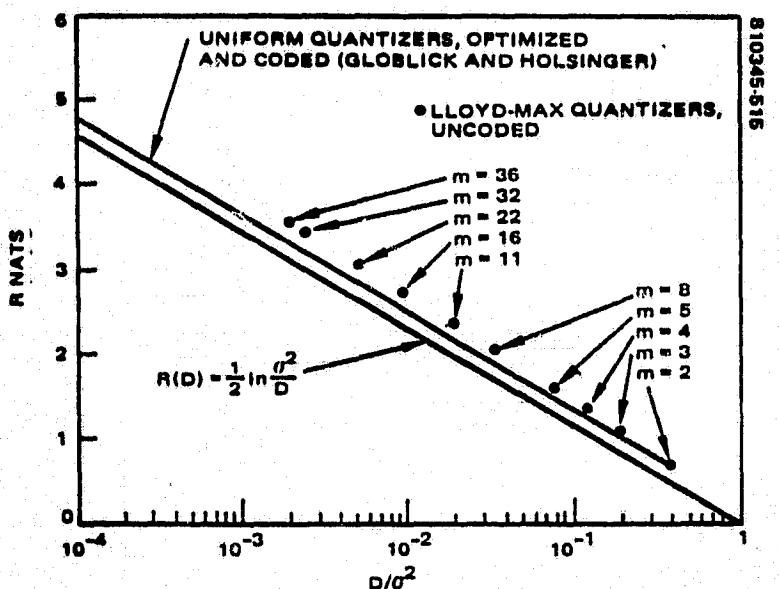


FIGURE I-2. MAXIMUM RATE VS NUMBER OF QUANTITIZATION BITS

Starting with the Cartesian form of the 2-d integrals from Ref. 2, Equations (8.4.45) and (8.4.46), we see that in polar form all integrals are of the form.

$$I = c \int_0^{2\pi} \int_1^M f(\rho, \theta)(\rho \, d\rho \, d\phi) = 2\pi c \int_1^M f(\rho, \theta) \, \rho d\rho \qquad \text{(I-15)}$$

So we have

$$2\pi d(0) = 0 \int_1^{\theta^{-1/h}} \rho d\rho + \int_{\theta^{-1/h}}^M \rho^{-(h-1)} \, d\rho$$

$$= \theta \left. \frac{\rho^2}{2} \right|_1^{\theta^{-1/n}} - \left. \frac{\rho^{-(h-2)}}{(h-2)} \right|_{\theta^{-1/h}}^M$$

$$= \frac{\frac{h}{2} \theta^{\frac{h-2}{h}} - \theta^{\frac{h-2}{h}} - \left(\frac{h}{2} - 1\right) \theta + \theta^{+\frac{h-2}{n}} - M^{-(h-2)}}{2\pi(h-2)} \qquad \text{(I-16)}$$

So

$$d(0) = \frac{\frac{h}{2} \theta^{\frac{h-2}{h}} + \left(1 - \frac{h}{2}\right) \theta - M^{-(h-2)}}{1 - M^{-(h-2)}} \qquad \text{(I-17)}$$

The variance is

$$\sigma^2 = \frac{1}{2\pi}\int_1^M \rho^{-(h-1)}\,d\rho = \frac{\rho^{-(h-2)}}{2\pi(h-2)}\Bigg|_1^M = \frac{1 - M^{-(h-2)}}{2\pi(h-2)} \qquad \text{(I-17)}$$

So the normalized distortion is

$$\frac{d(\theta)}{\sigma^2} = \frac{\frac{h}{2}\theta^{\frac{h-2}{h}} + \left(1 - \frac{h}{2}\right)\theta - M^{-(h-2)}}{1 - M^{-(h-2)}}\, M^{-h} \le M \le 1 \qquad \text{(I-18)}$$

Next evaluate the rate thus

$$4\pi R(0) = \int_1^{\theta^{-1/h}} \rho \ell n \frac{\rho^{-h}}{\theta}\,d\rho = \int_1^{\theta^{-1/h}} (-h\,\rho \ell n\rho - \rho \ell n\theta)\,d\rho$$

$$= h\left(\frac{\rho^2}{4} - \frac{\rho^2}{2}\ell n\rho\right) - \frac{\rho^2}{2}\ell n\theta\Bigg|_1^{\theta^{-1/h}}$$

$$= \frac{h}{4}\theta^{-2/h} + \frac{1}{2}\theta^{-2/h}\ell n\theta - \frac{1}{2} \qquad \text{(I-19)}$$

So

$$R(\theta) = \frac{2\ell n\theta - (h - \theta^{-2/h})}{10\pi} \text{ (nats/UHF)} \qquad \text{(I-20)}$$

I.2 HP-97 RATE-DISTORTION FUNCTION PROGRAM DESCRIPTION

For those with HP calculators, I give the program to calculate $d(\theta)/\sigma^2$ and $R(\theta)$, in the two-dimensional case, below. The symbols used are the same as those in use in this appendix. When M is entered, it is printed. Then $\theta_{min} = M^{-h}$ is found and printed. K multiplies θ by a fixed factor, and by the operator, on each circuit around the loop. I have been using 2. On each circuit θ, $d(\theta)/\sigma^2$, $R(\theta)$ and $R(\theta)/R_{max}$ are printed in that order. When $\theta > 1$ the program stops. The last computation gives R_{max} close enough to unity to permit plotting R_{max} vs number of image lines. The result is shown in Figure I-3. A graph of information rate vs percent distortion is shown in Figure I-4. Information rate is given as a percent of R_{max}. The parameter characterizing each curve is number of image lines.

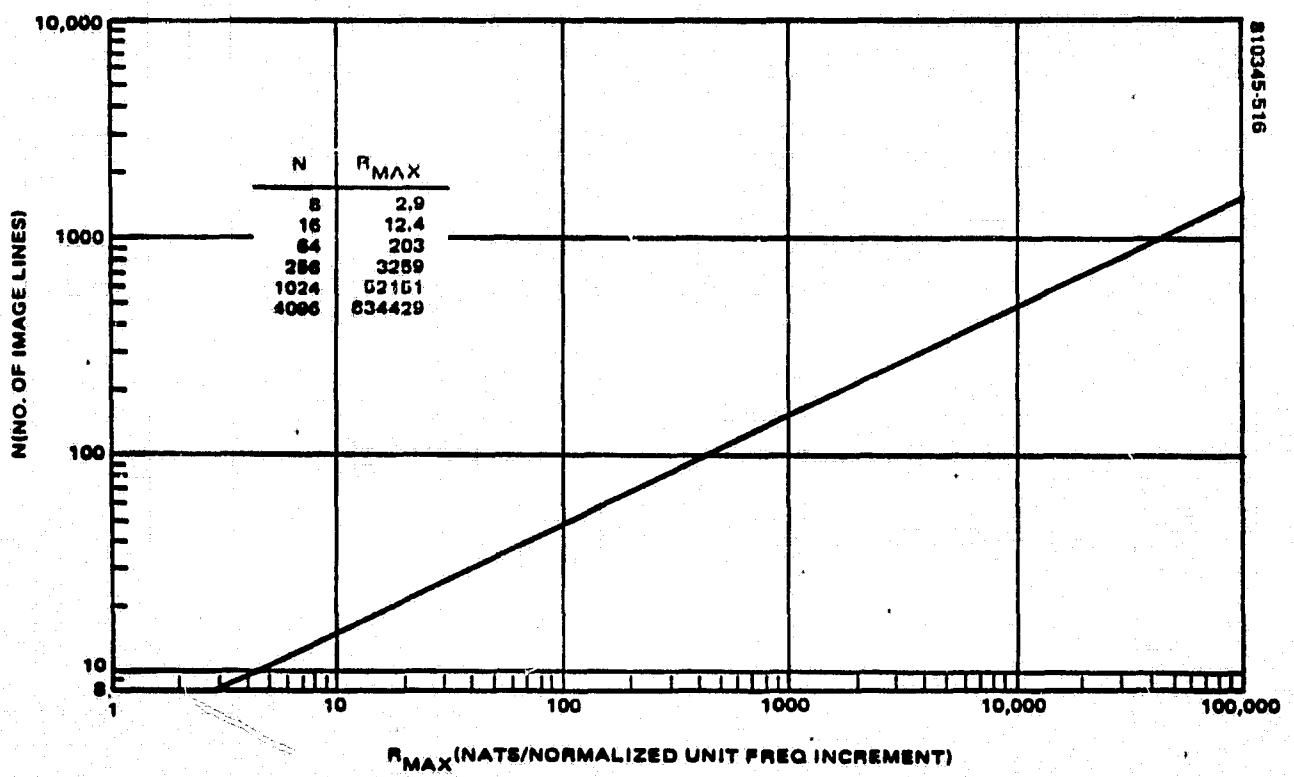


FIGURE I-3. MAXIMUM INFORMATION RATE VS NUMBER OF IMAGE LINES

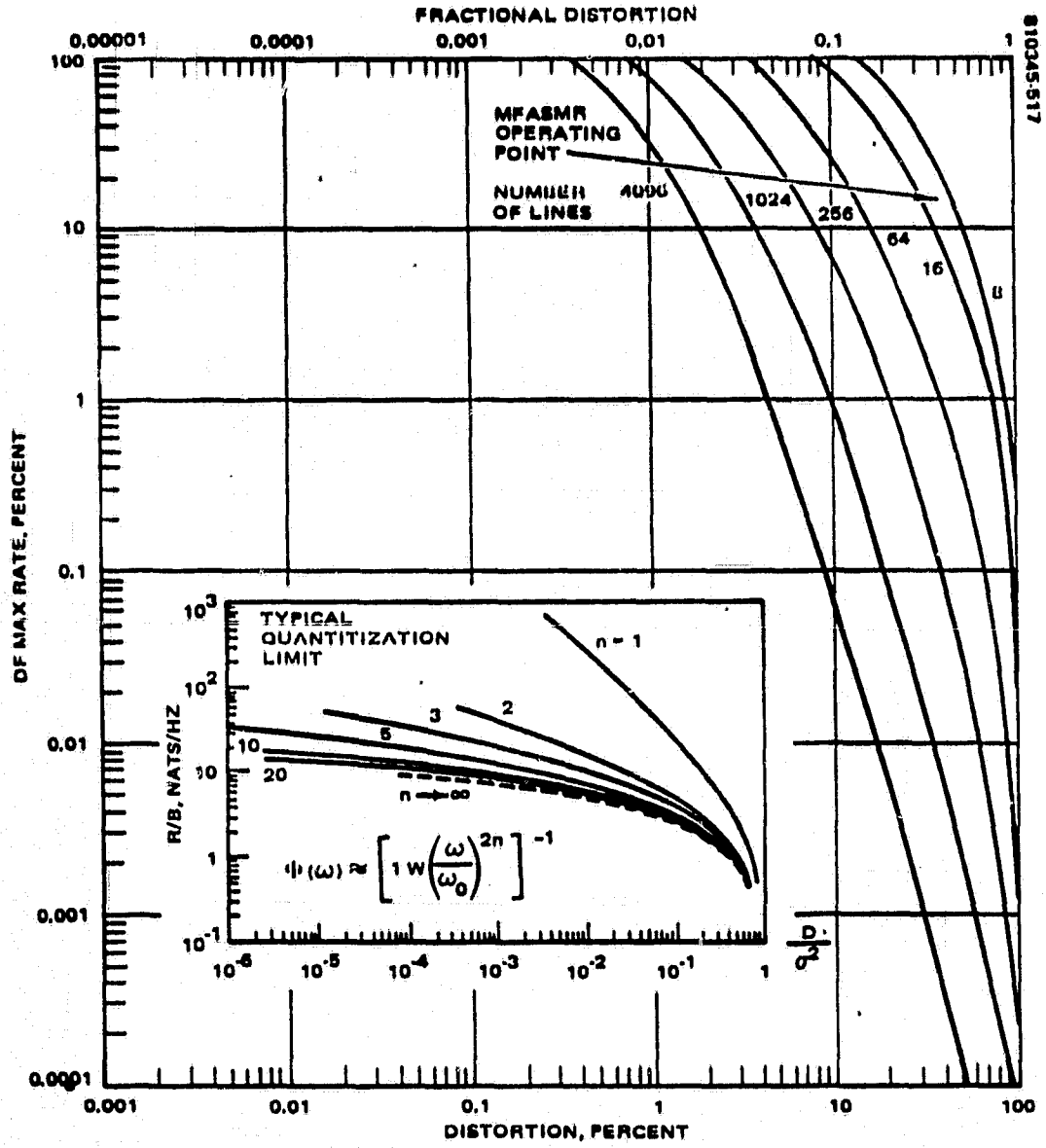


FIGURE I-4. RATE-DISTORTION FUNCTIONS FOR $\phi(\omega) = \omega^{-5/2}$ AND TYPICAL FUNCTION FROM BERGER

TABLE I-1. HP-97 RATE-DISTORTION FUNCTION PROGRAM

REGISTER	Enter 0	Enter 1	2	3	4	5	6	7	8	9
VALUE	h	K	$\frac{h}{2}$	(h-2)	M	0	N	R(0)	$R_{MAX}(0)$	$M^{-(h-2)}$
ENTERM										
LBLA			RCL3			RCL5			STO7	
PRTX # LINES			RCL0			RCL2			1	
STO4			:			1/X			RCLG	
RCL0			y^x			CHS			X ≠ Y?	
CHS			RCL2			y^x			GTO2	
y^x			X			1			RCL7	
STO5			RCL4			-			STO8	
PRTX θ_{MIN}			RCL3			RCL0			LBL2	
RCL0			CHS			X			RCL7	
2			y^x			RCL5			RCL8	
:			STO9			LN			:	
STO2			-			2			EEX	
RCL0			1			X			2	
?			RCL2			+			X	
-			-			fπ			PRTX % UNDER SAMPLE	
STO3			RCL5			:			RCL1	
1			X			4			RCL5	
STO6			+			x^2			X	
PRT SPC			1			:			STO5	
LBL1			RCL9			PRTX R(0)			-1	
PRT SPC			-			STO7			STO+6	
PRT SPC			:						RCL5	
RCL5			PRTX $\frac{d(0)}{\sigma^2}$						1	
PRTX θ									X ≠ Y?	
									GTO1	
									RTN	

APPENDIX K

LIST OF VICAR IMAGE PROCESSING PROGRAMS WITH SHORT PROGRAM DESCRIPTIONS

From DIGITAL IMAGE PROCESSING,
Kenneth R. Castleman, Prentice-Hall, Inc. 1979

IMAGE GENERATION

The following programs are useful for generating digital images.

GEN* is a program that produces images in which gray level increases linearly, modulo 256, from the upper left-hand corner, at specified horizontal and vertical rates.

GRATE generates digital images with vertical or horizontal stripes of specified gray level on a specified background gray level.

SPOT* produces an image containing a spot. The spot may be Gaussian, conical, reciprocal, reciprocal squared, exponential, or dome-shaped. Spot location, size, and contrast may be specified.

LOCUS* generates images containing 5-by-5-pixel plus signs at specified locations.

COHER generates sinusoidal coherent noise images and can also add sinusoidal patterns to other images.

GAUSNOIS* generates random noise images having a Gaussian histogram of specified mean and standard deviation.

POLYNOIS* generates random noise images having a histogram that is specified by the user.

SYNPIC generates an image in complex format having gray level zero. The user supplies the real and imaginary values of a few specified pixels. This program can be used in conjunction with the inverse Fourier transform to produce images containing sinusoidal components.

POINT OPERATIONS

The following programs implement point operations (gray scale transformations) on digital images.

STRETCH* performs general point operations. The transformation may be specified as linear, piecewise linear, or as a cube root or exponential function. It can also produce contour lines at specified intervals.

ASTRTCH2* first computes the gray level histogram of the input image. The program then analyzes that histogram to determine the point operation required to put the histogram into a specified form. The user may specify either a linear point operation or a uniform or Gaussian output histogram. The linear point operation is designed to produce a specified amount of saturation at each end of the gray scale.

MATCH* performs the point operation required to make one image "match" another. The point operation may be based upon specified areas common to the two

pictures. Alternatively, for a single input image, the program will make the mean gray level of specified areas conform to specified values.

ALGEBRAIC OPERATIONS

The following programs perform pixel-by-pixel arithmetic functions.

PICAVE* can average up to 10 input images, which may be linearly displaced with respect to one another.

UNITAVE* divides the input image into a set of contiguous rectangles and averages those rectangles together. In the output image, each rectangle is replaced by the average. This program is useful for noise removal from periodic structures.

DIFFPIC* can add or subtract two images following a linear displacement. A specified linear point operation is also performed on the output image.

F* is a general-purpose pixel arithmetic routine. The function that relates the output image to the two input images has 11 specifiable parameters. These may be chosen to implement addition, subtraction, multiplication, division, exponentiation, and logarithms. The program operates by first generating a 256 × 256 two-dimensional look-up table, which is subsequently used to produce the output image.

F2 performs general arithmetic operations on one or two input images. The arithmetic operation is specified by a FORTRAN-like expression.

PIXC performs complex arithmetic on two complex-valued digital images.

PIXGRAD calculates the magnitude and angle of the gradient vector of an input image. The magnitude is taken as the maximum difference between the current pixel and its eight adjacent neighbors. The angle is specified by an integer from 1 through 8.

PIXH performs addition, subtraction, multiplication, and division on two input images. The output and the two input images may have one or two bytes per pixel. The arithmetic is done in a fractional format.

PIXRMS produces an output image composed of local means or local standard deviations of the input image.

RATIO* is a preprocessor used for comparing two input images. These may be compared on the basis of ratio, log ratio, difference, or log difference. The program generates the parameters required for proper scaling of the output image and fetches program F, which performs the specified operation.

LOCAL OPERATIONS

The following programs perform a variety of specialized local operations on input images. Many of them were designed to remove specific blemishes from images.

SAR* can replace specified rectangular areas with an average of surrounding gray levels. It is also commonly used to copy images from one data set to another.

AUTOSAR locates pixels that deviate from the average of the pixels above and below by more than a specified tolerance. Such pixels are replaced with the average of the pixels above and below.

QSAR* can add or subtract specified values at all pixels within specified rectangular areas of the input image.

PSAR* can add or subtract specified values at all pixels within specified polygonal boundaries.

ERASE* locates and sets to zero the pixels within connected sets having gray level and perimeter below specified thresholds. It is used to remove small objects from an image.

BLEM replaces specified areas in an image by linear interpolation.

ADL* can add or subtract a specified value to all pixels lying along a straight line between any two pixels in the input image. The diagonal line is specified by its end points.

ADESPIKE* replaces the gray level of a pixel if it differs from its four nearest neighbors by more than a specified tolerance. The average of its adjacent neighbors is used for the replacement.

IMAGE MEASUREMENT

The following programs are concerned with extracting and displaying various measurements from an input image.

LIST* can be used to list the gray levels or the histogram of an image on a line printer.

HISTO generates and plots gray level histograms on a line printer.

CLSTR* and HIST2 compute the two-dimensional histogram of a pair of input images.

LPLOT2* produces a graphic plot of the gray levels along a specified diagonal line in an image.

LAVE* can average all the horizontal or vertical lines in an image. The average values are listed on the line printer and output as a one-line image.

PIXSTAT produces output images which represent the local mean, variance, second moment, or standard deviation of an input image.

LITEXFER* is designed to calculate the light transfer characteristics using input images that are flat fields at various intensity levels. For specified rectangular

areas within the set of input images, the program computes and plots the mean and standard deviation of gray level as a function of the input brightness level.

THRESHLD* locates and lists areas of an image containing points above a specified gray level threshold. When such a point is located, a 30 by 50-pixel area surrounding that point is listed on the line printer and set to zero gray level in the image. This program is useful for automatically locating small objects, such as stars, in an image.

GRIDLOCA* and **GRIDLOCB*** are used together to locate the intersections in an image containing a rectilinear grid network. These programs are helpful in the geometric calibration of image digitizers.

DRECK locates line segments in digitized line drawings.

ANNOTATION AND DISPLAY

The following programs are concerned with adding various types of annotation to images or with effecting image display.

ARROW writes arrows into an image at specified locations.

MARK* superimposes rectangular marks centered at specified coordinates in an image.

SCRIBE* places rectangles around specified areas in an image.

GRID* overlays a rectilinear grid network into a digital image.

MAPGRID* overlays an alternating black and white grid (dashed lines) into an image.

OVERLAY* superimposes a latitude-longitude grid onto cartographic projection images produced by program MAP2.

MASK* adds gray scales, pixel coordinate reference marks, label annotation, and a gray level histogram to an image in preparation for display. Most of the digital images in this book were produced using program MASK.

SHADY introduces shading and contour lines into an image. The shaded image is actually a partial derivative image taken in a specified direction. This assists visualization of slowly varying (low-frequency) images.

DNSYMBOL replaces each pixel with a square multi-pixel black and white symbol that represents gray level. When the image is displayed, the user can read the gray level of each pixel by examining the symbol.

DISPLAY* produces images by printing on a line printer. Overprinting of characters is used to achieve up to 64 gray levels.

PRINTPIX also prints images on a line printer. However, it includes facilities for demagnification, variable aspect ratio, and a gray scale transformation.

The following programs are designed for general user specified geometric operations.

GEOM* transforms a specified control grid of contiguous quadrilaterals into a specified rectangular grid. This program is particularly efficient when vertical displacements are small.

LGEOM* performs the same transformation as GEOM but is more efficient when vertical displacements are large. Both programs use bilinear interpolation of displacement and gray levels in the transformation.

GEOMA* performs geometric operations where the transformation is specified as a mapping of quadrilaterals into quadrilaterals rather than into rectangles. This more general format may afford a more convenient specification in some cases. For example, it is possible to degenerate the quadrilaterals into triangles and specify the geometric transformation as a mapping between control grids composed of contiguous triangles. In general, GEOMA runs approximately one-third longer than GEOM on comparable transformations.

POLARECT projects a rectangular image into a sector of a circle, and vice versa.

Rotation and Magnification

The following programs are designed to perform rotation and magnification on digital images.

ROTATE* performs 90° clockwise or counterclockwise rotation.

FLOT* can perform plus or minus 90° rotation or reflection about the horizontal or vertical axis of an image.

ROTATE2* rotates an image through a specified angle about a specified point and places the center of rotation at a specified point in the output image. This program generates the necessary parameters (control grids) and fetches GEOM or LGEOM to perform the rotation.

ANGLE provides the specified rotation and translation necessary to bring two images into registration.

MAG* can generate the parameters necessary to magnify or reduce an image, alter its aspect ratio, or skew the top of the image right or left with respect to the bottom. This program fetches GEOM to perform the operation.

SIZE* can also magnify or reduce an image or change its aspect ratio.

RESAMP* reduces the size of an image by skipping lines and samples.

APAVG* reduces the size of an image by averaging rectangular arrays of pixels to form the new gray level values.

EXPAND* enlarges an image by a specified factor *N* by repeating each pixel in an *N* by *N* array.

INTERP* magnifies an image by a factor of 2 using bilinear interpolation.

CLASP changes the linear aspect ratio of an image by magnification in the horizontal direction. The program can use either linear or cubic spline interpolation.

Image Combination

The following programs may be used to combine images to form larger ones.

CONCAT* can combine up to ten input images of the same size. The images are concatenated side by side and one above the other.

INSECT* can combine two images of unequal size. Specified rectangular areas from each of the two input images are placed at specified locations in an output image of specified size. Where the two areas overlap, the second input image prevails. Areas of the output image where neither input image falls are set to 255.

MOSAIC* is useful for combining multiple overlapping views of an area into one composite output image. The program assumes that the input images are all the same size as the desired output and that the smaller images have already been placed in proper position on a background of zero gray level. Where nonzero portions of images overlap, the program takes values from the input images in a specified order of priority.

HISTLOC generates an output image from a specified two-dimensional look-up table. The gray levels of two input images provide the address into the table. This program is useful for identifying which areas of an image correspond to different clusters in a two-dimensional histogram.

LYNX joins the top half of one image to the bottom half of another. Translation is provided to bring overlapping areas into registration.

Map Projection

The following programs are designed for producing map projections of aerial or spacecraft images.

MAP2* is a general cartographic projection program. The program generates the necessary parameters to project an image from the camera coordinate system into a standard cartographic projection map. The six cartographic projection options are Mercator, Lambert conformal conic, oblique orthographic, polar orthographic, oblique stereographic, and polar stereographic. The program uses LGEOM to perform the actual projection.

MAPTRANS can be used to transform a projected image from one cartographic projection to another. The program uses LGEOM to perform the necessary geometric transformation.

MERCATOR generates LGEOM parameters to transform an image into a mercator projection.

PROJECT generates LGEOM parameters to transform an image from mercator to orthographic projection or vice versa.

GEOTRAN* performs geometric transformation to effect projection from a sphere onto a cylinder or a plane. It can also perform orthographic projections to produce the image that would be obtained if the camera were moved to a specified different position.

CORRELATION AND CONVOLUTION

The following programs perform tasks that are implemented with digital correlation or convolution.

CROSS* compares specified areas within two input images to determine the translation required to register the two images. It computes and prints the sum of squared differences between a stationary rectangle in one image and a moving rectangle in the second image and the relative displacement that results in minimum sum-squared difference. The size of the rectangle and the area of search can be specified by the user. It is useful in many cases requiring cross-correlation or autocorrelation.

REGISTER* generates the GEOM parameters required to bring one picture into registration with another. Specified rectangular areas of the images are compared by first removing low-frequency information, then determining the translation that results in minimum mean square difference, and finally fitting a polynomial surface through the required displacements. A GEOM control grid that spans the image is computed from the polynomial surface.

FILTER* performs general two-dimensional convolution. The input image is convolved with a specified rectangular impulse response. This program assumes that the impulse response has four-quadrant symmetry. It permits either specified or automatic scaling of the output gray scale.

AFILTER* is similar to FILTER except that no assumptions are made about symmetry in the point spread function.

FILTER2* computes the point spread function of a specified two-dimensional transfer function. The program then fetches FILTER to perform the convolution. Thus the user may perform convolution filtering where the filter is specified in the frequency domain.

GFILTER accepts parameters that describe a two-dimensional transfer function having elliptical cross sections and Gaussian profile. It then fetches FILTER2, which computes the corresponding psf and, in turn, fetches FILTER to carry out the convolution. This program makes it convenient to implement lowpass filters of Gaussian profile in the frequency domain.

BOXFLT* convolves an input image with a flat-topped rectangular point spread function. The user specifies the size of the psf. Because of the constrained psf, this program executes much faster than FILTER.

BOXFLT2 is similar to BOXFLT except that it can also produce a highpass filtered output image. This is obtained by subtracting the lowpass filtered image from the input.

FASTFIL2* can produce a highpass and a lowpass filtered version of an input image. The lowpass psf is a two-dimensional rectangular pulse. The highpass filtered image is obtained by subtracting a specified fraction of the lowpass filtered image from the original. Options to reduce ringing at discontinuities are included.

FASTFIL1* is a one-dimensional (horizontal) version of FASTFIL2.

MEDIAN produces an output image in which the gray level represents the median value of surrounding pixels within the input image. The program is limited to one-dimensional (line-by-line) processing.

POLYFILT implements convolution with a spatially variant point spread function. Up to 10 different psfs may be supplied. A second input image specifies the areas in which each of the psfs are to be applied.

SMEAR73* is designed for Wiener deconvolution of linear motion blur. The user specifies the direction and amount of motion blur, and the program calculates the psf of the Wiener deconvolution filter assuming white signal and noise. The program estimates the signal and noise power spectra from the image and fetches AFILTER to implement the convolution.

FOURIER TRANSFORM COMPUTATION

The following programs perform tasks that involve computation of the Fourier transform.

FFT1* computes the forward and inverse one-dimensional complex Fourier transform on a line-by-line basis.

FFT1PIX* can be used to display the complex transforms produced by FFT1.

FFT2* computes the direct or inverse two-dimensional complex Fourier transform of a digital image.

FFTPIC* produces a digital image from the complex data set produced by FFT2. This permits display of the amplitude and phase of the Fourier transform as an image.

POWER* computes the one-dimensional power spectrum of each line in a digital image and displays the root-mean-square power spectrum of all lines on the line printer.

MTF can be used to compute the modulation transfer function of a camera system from digitized images of sine wave test targets.

The following programs implement linear filtering in the frequency domain.

FREQMOD* multiplies the complex spectrum produced by FFT2 by a user-specified transfer function. The resulting complex spectrum can be inverse transformed by FFT2 to implement linear filtering. The user specifies the profile of an elliptically symmetrical, real, nonnegative transfer function.

FFTFIT can either multiply a complex spectrum by an input picture or make the amplitude of the spectrum proportional to the input picture. In either case, phase is unaltered. This program permits frequency domain filtering with unrestricted real transfer functions.

FFTFIL* performs one-dimensional frequency domain filtering on a line-by-line basis. Each line is transformed, multiplied by a specified transfer function, and inverse transformed. The transfer function may be a bandpass or notch filter with user-specified frequency bands. The program can also modify the spectrum by interpolating across specified frequency bands. This program is useful for removing coherent noise from images. Program POWER may be used to determine the unknown noise frequency.

UNSHADE* implements highpass filtering in the frequency domain. It multiplies a complex spectrum (from FFT2) by a transfer function that is unity everywhere except near the vertical and horizontal frequency axes. The transfer function goes to zero at the frequency axes with a negative Gaussian profile. This removes low-frequency information (shading) from the inverse transformed image.

SPIKMASK multiplies the complex spectrum by a transfer function that is unity everywhere except in specified small rectangular regions. In these regions, it takes on the value zero. This program may be used to remove spikes in the frequency domain produced by periodic noise in the image.

OUTSPIKE* removes spikes from the spectrum of an image containing periodic noise. The program first locates spikes in the amplitude spectrum by searching for local maxima above a specified severity. It then removes the spike by interpolation of the surrounding values. The phase is not altered.

APODIZE* modifies an image near the edges so as to reduce the effects of truncation when the Fourier transform is computed. The program places a quarter cycle of a sine function at each end of each line and column of pixels so that making the image periodic in two dimensions does not produce discontinuities. This prevents the introduction of artifact along the frequency axes in the two-dimensional Fourier transform.

OTF1* can compute the horizontal component of a blurring transfer function given an image containing a degraded vertical edge. It can also compute the horizontal component of a transfer function, given the line spread function. The program differentiates each line through a degraded vertical edge to obtain an estimate of the

line spread function. These estimates are averaged together and inverse transformed to determine a component of the transfer function.

OTF2* implements two-dimensional frequency domain image restoration by Wiener deconvolution. The user supplies the complex spectrum of both the input image and the degrading psf. The program assumes white signal and noise spectra, estimates their amplitude, and computes a Wiener deconvolution transfer function. It produces an output spectrum that is the product of the input spectrum and the restoration transfer function. The restored image may be obtained by inverse transforming the output spectrum.

STEREOMETRY PROGRAMS

The following programs are concerned with stereometry and stereoscopic display.

RANGE computes a range image, given a stereo pair of input images.

ELEVMAP generates a topographic map of a surface, given a stereoscopic pair of input images.

VPROFILE* produces a vertical profile plot of a surface in a range image.

STEREO* produces a stereo pair, given a brightness image and a range image. The user supplies the right eye image and the range image, and the program computes the left eye image.

APPENDIX L

AN ITERATIVE PROGRAM FOR MAXIMUM ENTROPY FOURIER SYNTHESIS

AN ITERATIVE PROGRAM FOR MAXIMUM ENTROPY FOURIER SYNTHESIS

Stephen J. Wernecke* and C. John Grebenkemper

Stanford Radio Astronomy Institute Technical Memo # 582

Stanford, California 94305

*Present address: ESL Inc.
495 Java Drive
Sunnyvale, CA 94086

December 7, 1976

TABLE OF CONTENTS

1. INTRODUCTION

This report contains a detailed description and a source listing of MEPIX, an iterative Fortran program for maximum entropy Fourier synthesis and two-dimensional spectral analysis. The program is distributed to enable interested researchers to begin exploring two-dimensional maximum entropy processing without first undertaking a major software development effort. MEPIX was originally designed for use as a radio astronomy imaging procedure: it produces a nonnegative image from a finite number of noisy, irregularly-spaced samples of an unknown object's Fourier transform (the complex visibility function). Such analysis is analogous to two-dimensional power spectral estimation from autocorrelation measurements. The program listing in the Appendix is sufficiently well documented that those who wish to use the program for these purposes should be able to do so with only cursory reading of the text; however, the usual caveats about uninformed use of contributed software are operative. While there are no known errors in MEPIX, this does not guarantee that there are none, and final responsibility for correct operation remains with the user.

With suitable modification, MEPIX can be used for one-dimensional problems of Fourier spectroscopy, spectral analysis of noisy and/or irregularly-spaced autocorrelation samples, and estimation of line-integrated brightness in radio astronomy. It should also be realized that revision of the program to tailor it to special measurement geometries may be profitable. If, for example, visibility measurements are located at or interpolated onto the vertices of a rectangular lattice, the computation strategy can be altered to reduce memory requirements by

a factor of about three and execution time by a factor exceeding four.

The basic reference for this report is "Maximum Entropy Image Reconstruction," by S. J. Wernecke and L. R. D'Addario. This paper is scheduled to appear in the May 1977 issue of the IEEE Transactions on Computers; in the meantime, preprints are available from the authors. This paper describes the rationale for and mathematical specification of the imaging procedure realized in MEPIX, and it also constitutes one chapter of the senior author's dissertation, "Maximum Entropy Techniques for Image Reconstruction from Interferometer Measurements and Projections," available from University Microfilms, Ann Arbor, Michigan 48106. The following section summarizes the mathematical development contained therein.

II. MATHEMATICAL DEVELOPMENT

Notation

$m_i = |m_i| \exp[j \arg(m_i)]$: the ith complex visibility measurement

(u_i, v_i): spatial frequencies of the ith measurement

M: number of measurements

σ_i^2: variance of the ith measurement error

$\underline{f}$: a nonnegative, but otherwise arbitrary, image vector whose kth component is denoted as f_k

$\underline{f}_{ME}$: the ideal maximum entropy image. The kth component is $[\underline{f}_{ME}]_k$

$\underline{f}^{(n)}$: the image after the nth iteration. The kth component is denoted as $f_k^{(n)}$

N: number of pixels (elements in the image vector).

(x_k, y_k): coordinates of the center of the kth pixel

$\Delta x, \Delta y$: pixel widths in x- and y-directions, respectively

ΔA: pixel area ($= \Delta x \Delta y$)

λ: a positive constant selected and perhaps interactively adjusted by the user

Objective Function

The maximum entropy image $\underline{f}_{ME}$ maximizes

$$J(\underline{f}) = \Delta A \sum_{k=1}^{N} \ell n\ f_k - \lambda \sum_{i=1}^{M} \frac{1}{\sigma_i^2}$$

$$\times \left| m_i - \Delta A \sum_{k=1}^{N} f_k \exp[-j2\pi(u_i x_k + v_i y_k)] \right|^2$$

$$= \Delta A \sum_{k=1}^{N} \ell n\ f_k + \Delta A \sum_{k=1}^{N} d_k f_k$$

$$- \frac{\Delta A^2}{2} \sum_{\ell=1}^{N} \sum_{k=1}^{N} p_{\ell,k} f_k f_\ell - \lambda \sum_{i=1}^{M} \frac{1}{\sigma_i^2} |m_i|^2 \qquad (1)$$

where

$$d_k = 2\lambda \sum_{i=1}^{M} \frac{1}{\sigma_i^2} |m_i| \cos[2\pi(u_i x_k + v_i y_k) + \arg(m_i)] \qquad (2)$$

and

$$p_{\ell,k} = 2\lambda \sum_{i=1}^{M} \frac{1}{\sigma_i^2} \cos 2\pi[u_i(x_\ell - x_k) + v_i(y_\ell - y_k)] \qquad (3)$$

If pixels are arranged on a rectangular grid, there are only about 2N distinct values in the array $\{p_{\ell,k}\}$, and a sum of the form $\sum_{k=1}^{N} p_{\ell,k} f_k$ for $\ell = 1,2,\ldots,N$ is identified as a two-dimensional convolution sum regardless of measurement coverage $\{(u_i, v_i),\ i = 1,2,\ldots,M\}$. These convolution sums are evaluated efficiently by fast Fourier transform (FFT) techniques.

Gradient

$$\nabla J(\underline{f}) = \left[\frac{\partial J}{\partial f_1} \quad \frac{\partial J}{\partial f_2} \cdots \frac{\partial J}{\partial f_N}\right]^T$$

where

$$\frac{\partial J}{\partial f_\ell} = \frac{\Delta A}{f_\ell} + \Delta A\, d_\ell - \Delta A^2 \sum_{k=1}^{N} P_{\ell,k} f_k \tag{4}$$

The maximum entropy image satisfies $\nabla J(\underline{f}_{ME}) = 0$.

Selection of λ

An appropriate choice of λ leads to an image $\underline{f}_{ME}$ satisfying

$$\sum_{i=1}^{M} \frac{1}{\sigma_i^2} \left| m_i - \Delta A \sum_{k=1}^{N} \left|\underline{f}_{ME}\right|_k \exp\left[-j2\pi(u_i x_k + v_i y_k)\right] \right|^2 \simeq M$$

While it is not known how to determine λ to satisfy this requirement in general, successful use has been made of the formula

$$\lambda = \frac{N\Delta A}{2(1 + m_F/\sigma_F)} \tag{5}$$

where m_F is the measured (or estimated) flux density and σ_F is its standard deviation. In some cases this value is too large, and dividing it by the number of measurements is helpful. It should be realized that iterative adjustment of λ may be required to control image-measurement compatibility, larger values of λ leading to increased agreement.

Optimization by Univariate Image Perturbation

Let $\underline{f}^{(n)}$ be the image after the nth iteration. If only one pixel brightness is to be changed in the next iteration, the optimizing step is found by setting the corresponding partial derivative equal to zero. The

equation can be solved analytically to yield

$$f_{\ell}^{(n+1)} = \frac{1}{2\Delta A p_{\ell,\ell}} \left\{ d_{\ell} - \Delta A \sum_{k \neq \ell} p_{\ell,k} f_k^{(n)} + \sqrt{\left(d_{\ell} - \Delta A \sum_{k \neq \ell} p_{\ell,k} f_k^{(n)} \right)^2 + 4 \Delta A \, p_{\ell,\ell}} \right\} \tag{6}$$

where ℓ is the index of the pixel whose brightness is changed in the $(n+1)$st iteration. For all $k \neq \ell$, $f_k^{(n+1)} = f_k^{(n)}$. Since only one brightness is changed in each iteration of a univariate search, it takes at least N iterations to change all brightnesses. One might set up a correspondence between iteration number and adjusted pixel index such as $\ell = n \bmod N + 1$.

Equation (6) always yields a positive brightness value since $p_{\ell,\ell} > 0$, and this iterative strategy makes J a nondecreasing function of iteration number. Although no rigorous proof is offered, the univariate procedure should converge to the ideal maximum entropy image because a) the objective function is unimodal; b) the gradient is continuous (there are no corners in the multidimensional contour diagram of the objective function); and c) there is no artificial damping mechanism that might cause image changes to approach zero prematurely.

However, there is no reason to expect this algorithm to be very fast: successive steps are perpendicular, which causes a zig-zag trajectory just as with steepest ascent. The univariate method is also computationally undesirable for large N since FFT calculation of the convolution sum provides no advantage. The reason for this is that only one element of the convolution sum is needed in each iteration, yet each iteration changes all convolution results.

The attractive aspects of the univariate perturbation scheme are automatic satisfaction of the nonnegativity constraint and elimination of

the need for a search in the perturbation direction (the optimal stepsize is defined analytically). The following multivariate search strategy, the one that is programmed in MEPIX, was developed to retain some of the good features of the univariate approach while taking advantage of FFT efficiency as well. The flow chart in Fig. 1 pertains to this multivariate method.

III. THE ITERATIVE ALGORITHM IN MEPIX

Let J_n be the objective function value and ∇J_n the gradient after the nth iteration. The structure of the multivariate search algorithm in MEPIX is as follows:

0) Compute and store $\{d_\ell\}$ (Eq. 2) and $\{p_{\ell,k}\}$ (Eq. 3).

1) Take the starting image, $\underline{f}^{(0)}$, to be uniformly gray at a level that agrees with the measured (or estimated) flux density. Set $n = 0$.

2) Evaluate and store the convolution sum (using the FFT)

$$c_\ell^{(n)} = \sum_{k=1}^{N} p_{\ell,k} f_k^{(n)}$$

for $\ell = 1,2,\ldots,N$.

3) Compute J_n from Eq. (1).

4) Calculate ∇J_n according to Eq. (4).

5) Calculate all optimal univariate moves by a rearrangement of Eq. (6)

$$s_\ell^{(n)} = \frac{1}{2\Delta A p_{\ell,\ell}} \left\{ d_\ell - \Delta A (c_\ell^{(n)} - p_{\ell,\ell} f_\ell^{(n)}) + \sqrt{[d_\ell - \Delta A (c_\ell^{(n)} - p_{\ell,\ell} f_\ell^{(n)})]^2 + 4\Delta A p_{\ell,\ell}} \right\}$$

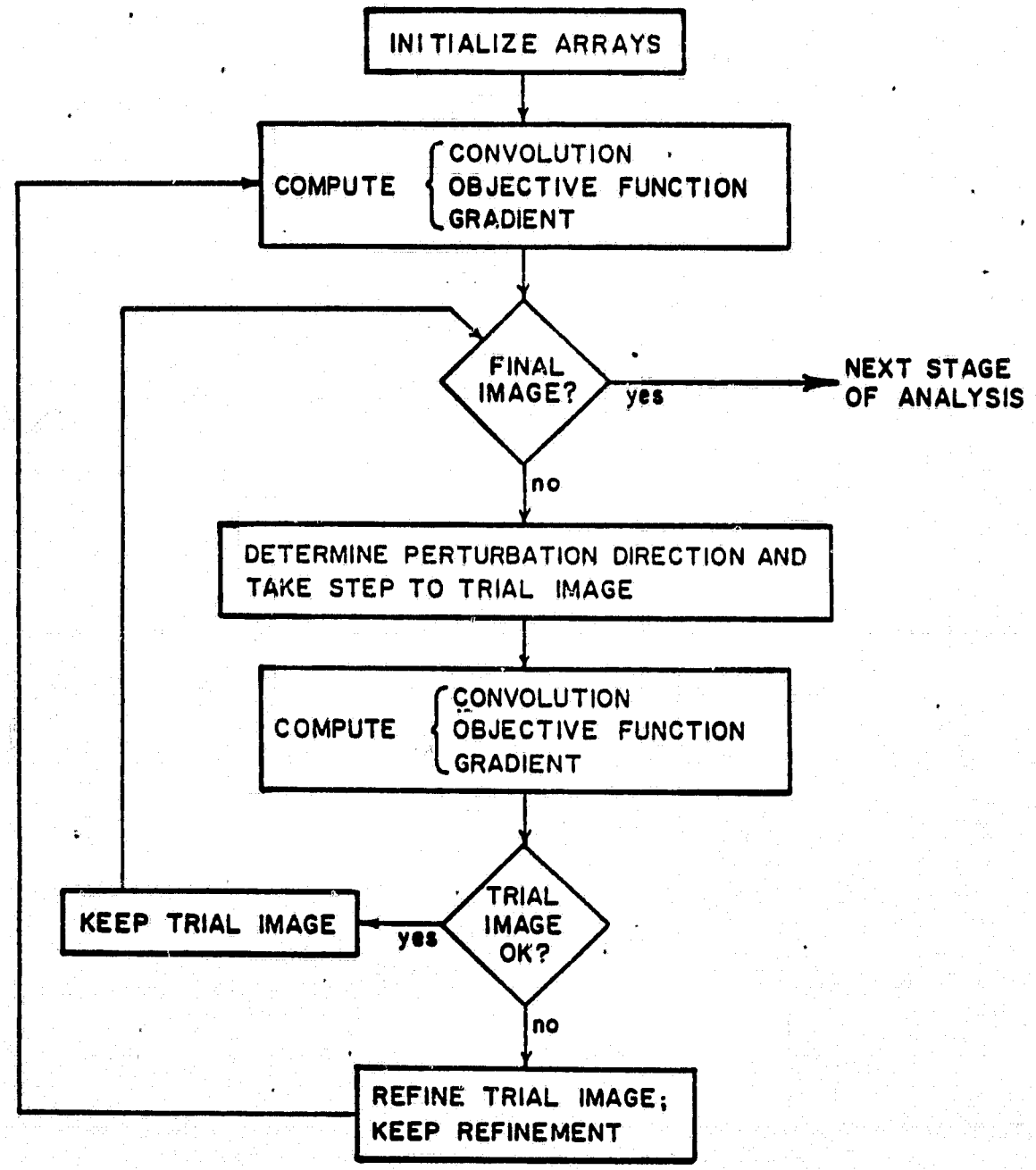


Fig. 1

6) Establish a temporary image $\underline{\tilde{f}}^{(n)}$ according to

$$\underline{\tilde{f}}^{(n)} = \underline{f}^{(n)} + \frac{1}{D}\left[\underline{s}^{(n)} - \underline{f}^{(n)}\right]$$

where D is a positive damping factor. If $D \geq 1$, all elements of $\underline{\tilde{f}}^{(n)}$ are guaranteed to be nonnegative. The discussion below contains comments on this parameter and on the choice of the fraction R introduced in step 10.

7) Store the perturbation vector $\underline{\Delta}^{(n)} = \underline{\tilde{f}}^{(n)} - \underline{f}^{(n)}$ and accumulate the inner product $\langle \nabla J_n, \underline{\Delta}^{(n)} \rangle$.

8. Calculate the convolution sum corresponding to $\underline{\tilde{f}}^{(n)}$. This result can overwrite the calculations of step 2. Calculate the objective function value $\tilde{J}_n = J(\underline{\tilde{f}}^{(n)})$.

9. Approximate objective function behavior in the perturbation direction by a quadratic fit. To second order,

$$J(\underline{f}^{(n)} + \varepsilon\underline{\Delta}^{(n)}) \simeq J_n + \langle \nabla J_n, \underline{\Delta}^{(n)} \rangle \varepsilon + \frac{1}{2} b\varepsilon^2$$

where ε is a scalar and b is a curvature parameter whose value is estimated by

$$b = 2\left[\tilde{J}_n - J_n - \langle \nabla J_n, \underline{\Delta}^{(n)} \rangle\right]$$

The objective function is concave so b is always negative in the absence of numerical errors. The optimal image step $\varepsilon^*\underline{\Delta}^{(n)}$ (relative to $\underline{f}^{(n)}$) is specified by

$$\varepsilon^* = -\langle \nabla J_n, \underline{\Delta}^{(n)} \rangle / b$$

and the corresponding optimal objective function value is predicted to be

$$J^* = J_n + \langle \nabla J_n, \underline{\Delta}^{(n)} \rangle \varepsilon^* + \frac{1}{2} b(\varepsilon^*)^2$$

10) Compute the fractional improvement $(\tilde{J}_n - J_n)/(J^* - J_n)$. If this ratio is greater than the threshold R or if $\varepsilon^* \geq 1$, make the assignments

$$\underline{f}^{(n+1)} = \underline{\tilde{f}}^{(n)}$$

$$J_{n+1} = \tilde{J}_n$$

$$n \leftarrow n + 1$$

and go to step 4. (In words, make the temporary image the image retained after the (n+1)st iteration.) If the fractional improvement is less than R and $\varepsilon^* < 1$, to to step 11.

11) Take the optimal move

$$\underline{f}^{(n+1)} = \underline{f}^{(n)} + \varepsilon^* \underline{\Delta}^{(n)}$$

An alternative computation if step 6 overwrites $\underline{f}^{(n)}$ is

$$\underline{f}^{(n+1)} = \underline{\tilde{f}}^{(n)} + (\varepsilon^* - 1) \underline{\Delta}^{(n)}$$

Make the assignment $n \leftarrow n + 1$ and go to step 2.

Discussion

The current program runs for the number of iterations specified by the user, that is, MEPIX does not contain an automatic stopping criterion that uses the search history to decide when to terminate.

The damping factor D (step 6) is incorporated to increase the numerical stability of the algorithm. Since the optimal move in the perturbation direction $\underline{\Delta}^{(n)}$ is determined by a simple quadratic

approximation, it is important that the step to the temporary image $\tilde{\underline{f}}^{(n)}$ be neither too large nor too small relative to changes in the objective function value. A fixed value of D can be used, but dynamic adjustment of D based on previous experience is expected to give better performance. Such an adaptive scheme is available as an option in MEPIX. D is updated according to

$$D^{(n+1)} = \max\{1,\ (1 - q)\, D^{(n)},\ \min[D^{(n)}/\sqrt{\varepsilon^*},\ (1 + q)\, D^{(n)}]\}$$

This formula has evolved from trial and error, but the basic intent is to increase the damping parameter when the move to the temporary image overshoots ($\varepsilon^* < 1$) the optimal image and to decrease D when the step to $\tilde{\underline{f}}^{(n)}$ falls short ($\varepsilon^* > 1$). The equation ensures that $D^{(n+1)} \geq 1$, which keeps images nonnegative, and it also provides the user with the capability of "hard-limiting" the size of the damping parameter change. $D^{(n+1)}$ will always be within the range $[(1 - q)\, D^{(n)},\ (1 + q)\, D^{(n)}]$ where q is specified by the user. If, for example, $q = .5$, the allowed change to D is at most 50%. If $q = 0$, D remains constant at the original value dictated by the user.

The fractional improvement threshold ratio R (step 10) determines when the temporary image is discarded in favor the the optimal move. If $R = 0$, the temporary image is retained whenever it increases the objective function value (whenever $\tilde{J}_n > J_n$). If $R = 1$, the temporary image is always discarded. $R = \frac{1}{2}$ is recommended--in fact, this value is "hardwired" into MEPIX, but it is easily changed--for the following reason: iterations taking the optimal step require two evaluations of the convolution sum (the primary computational burden of maximum entropy iterations) whereas iterations keeping the temporary image require only one and are thus more

economical. If the temporary image almost achieves the optimal objective function value in the chosen perturbation direction, it is wasteful spending an extra convolution to "fine-trim" the image when the choice of a new perturbation direction will likely lead to an even greater improvement. In practice, this attempt at economy has worked fairly well; savings due to it are on the order of 30%. The key to making this trick work is to select a good temporary image.

The quadratic approximation of step 9 appears to be excellent if the move to the temporary image is large enough. When the step size is too small, the optimal image is estimated by extrapolation rather than interpolation, and this leads to greater approximation errors. A sample follows:

$n = 23 \qquad J_n = 1.7791 \times 10^{-5} \qquad \varepsilon^* = 0.13765$

$D = 5.3 \qquad \tilde{J}_n = -1.2796 \times 10^{-5} \qquad J^* = 1.8591 \times 10^{-5}$

After taking the optimal move, $J_{n+1} = 1.8589 \times 10^{-5}$, which agrees with the prediction within two parts in 10^5. Even in the early iterations when the trial steps are largest, the approximation is accurate:

$n = 1 \qquad J_n = -1.0087 \times 10^{-4} \qquad \varepsilon^* = 0.41370$

$D = 11.7 \qquad \tilde{J}_n = -1.4562 \times 10^{-4} \qquad J^* = -5.6508 \times 10^{-5}$

After taking the optimal step, $J_{n+1} = -5.6684 \times 10^{-5}$.

IV. PROGRAMMING DETAILS

A block diagram of software components in MEPIX is presented in Fig. 2. The programming is modular to facilitate modification. Subroutines in MEPIX are divided into five functional groups, each of which

ORIGINAL PAGE IS
OF POOR QUALITY

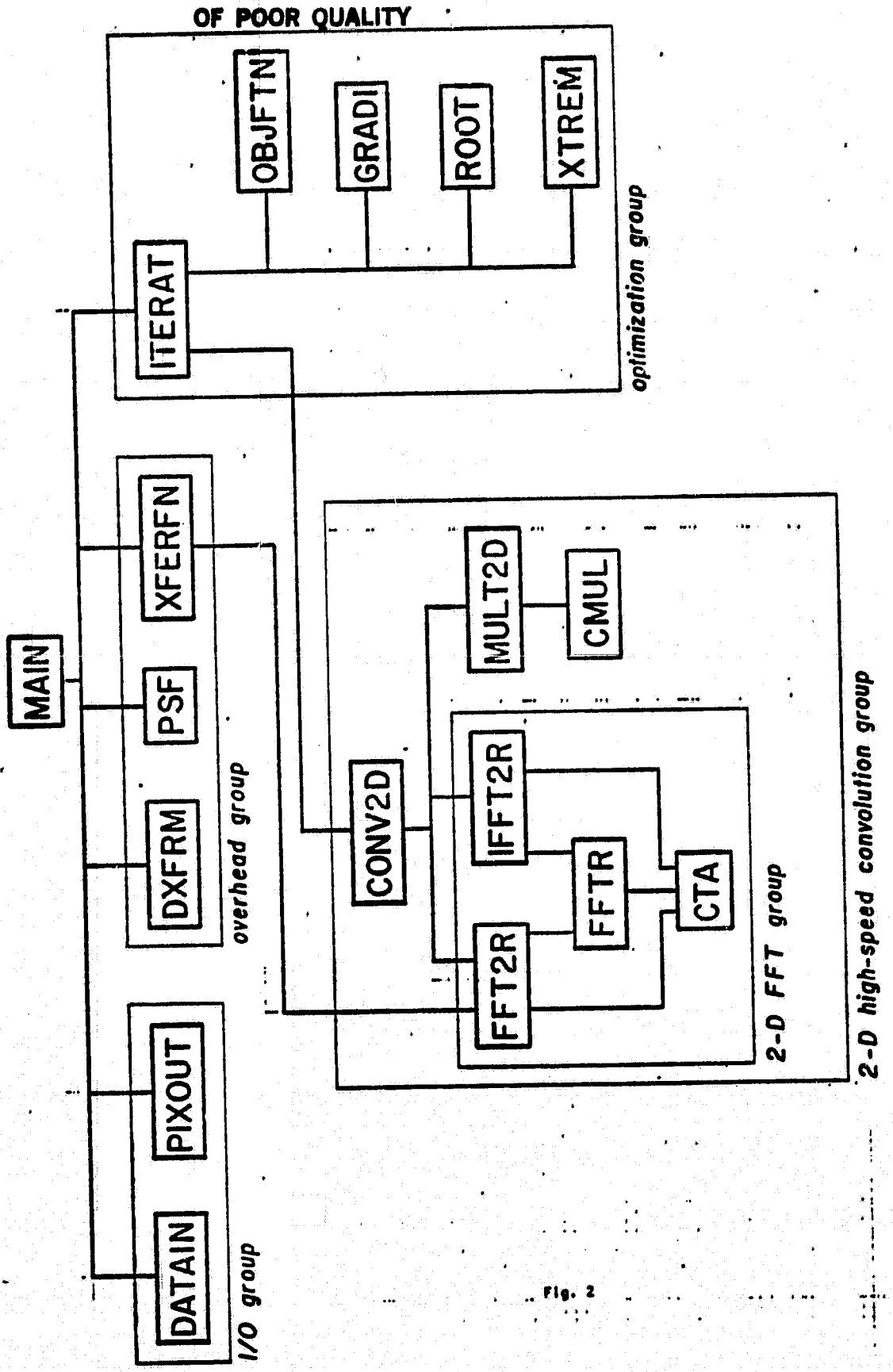


Fig. 2

is described briefly below. Source listings in the Appendix have extensive comments so the discussion here is concerned primarily with the interfaces among software modules and with suggestions for program modification.

Main Program and I/O Group

The main program listing in the Appendix indicates the minimum structure required for maximum entropy reconstruction. It is expected that new users will want to augment this code for documentation on output and error checks on input or to integrate the maximum entropy subroutine calls within more comprehensive software imaging systems. It might also be desirable to permit image input for continued iteration, perhaps with a different value of λ.

There is little computation within the main program itself. The primary purpose of MAIN is to reserve storage for arrays, to oversee I/O, and to call the subroutines in the overhead group before passing control to the main subroutine, ITERAT. Since ITERAT and the subroutines in the I/O and overhead groups are called only once, their code could be inserted in-line; however, this has not been done to take advantage of the object-time dimension capability of Fortran IV. MAIN reserves enough storage to accommodate the largest reconstruction task anticipated, but the actual dimensions of the image are part of the input to the program. This enables the user to produce and save a single load module with the flexibility to handle different image grid sizes, rectangular as well as square. In a multi-user virtual storage environment, this strategy also reduces the number of pages-in and pages-out, because array contents are "packed", and

and this in turn reduces the total job time. Two disadvantages of this structure are the need for line printer output within subroutine ITERAT and fairly long argument lists in subroutine calls.

The two-dimensional image array is stored in column-sequential format, and its correspondence with the physical x,y-coordinate system is shown in Fig. 3. Note that the outer pixels have their centers shifted half a pixel width from the edge of the field of view. The number of pixels in the image and the boundaries of the field of view are input by MAIN. The number of rows and columns need not be equal, but each must be a power of two.

The two subroutines in the I/O group must be supplied by the user. PIXOUT outputs the final image returned by ITERAT. If intermediate images are needed, they can be obtained by inserting additional calls to PIXOUT within ITERAT. Printout in ITERAT supplies the user with information about the numerical behavior of the optimization algorithm. This information is described in a later section.

DATAIN passes visibility measurements and their locations and variances to the main program. MEPIX knows that the visibility function is Hermitian; therefore, it is not necessary to indicate separate measurements at conjugate spatial frequencies. MAIN assumes, for purposes of computing λ (by Eq. 5) and the initial uniformly gray image, that the first visibility corresponds to $(u,v) = (0,0)$. This convention is not essential, but if it is to be changed, an additional parameter pointing to the flux density measurement is required. Note, a flux density value (either measured or estimated) should always be included to avoid possible divergence. No provision is made in the present software for interactive adjustment of λ.

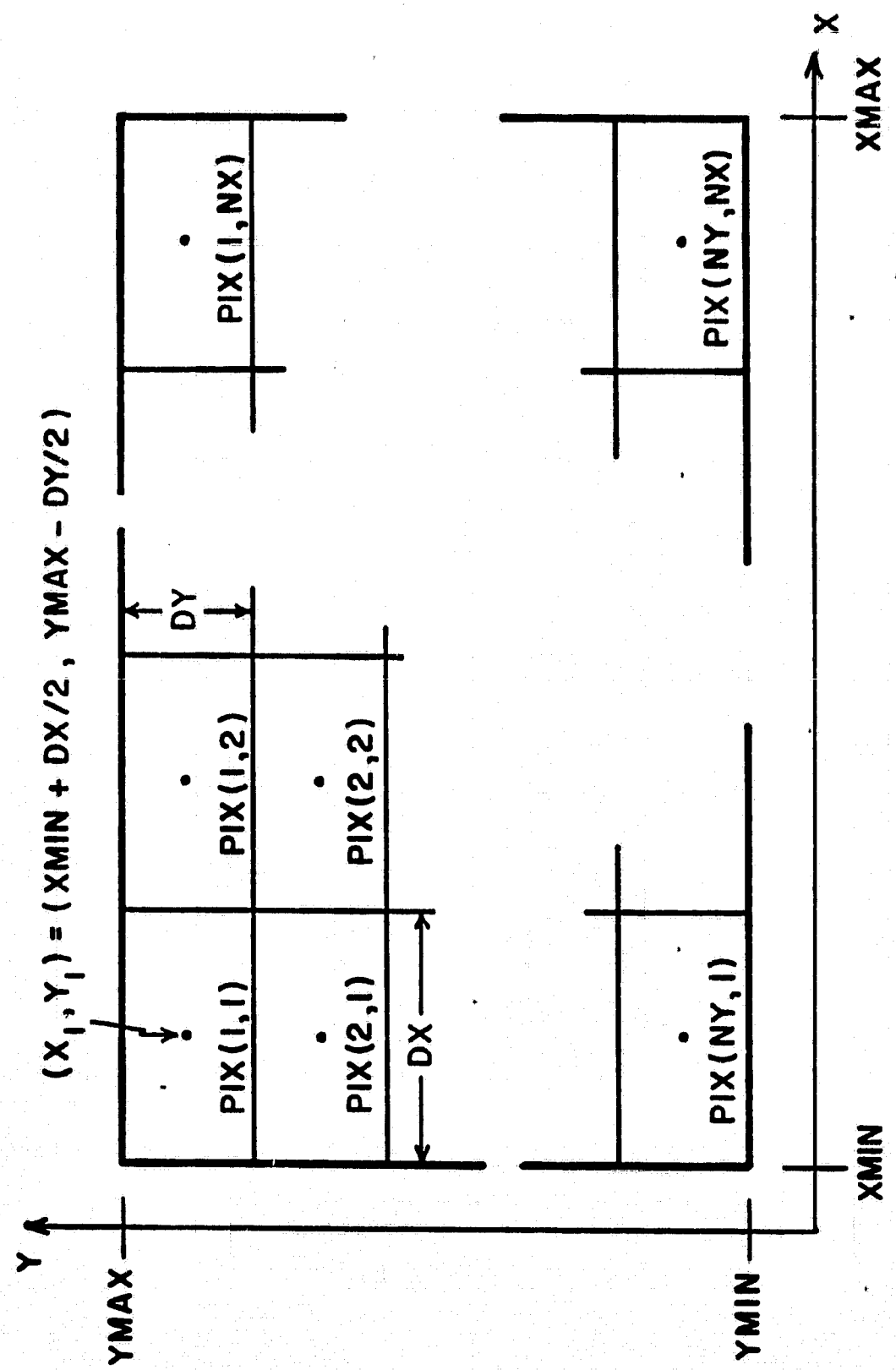

X
XMAX
XMIN
Y
YMAX
YMIN
PIX(1,NX)
PIX(NY,NX)
PIX(1,2)
PIX(2,2)
PIX(1,1)
PIX(2,1)
PIX(NY,1)
DY
DX
$(X_1, Y_1) = (XMIN + DX/2, YMAX - DY/2)$

Fig. 3

Overhead Group

DXFRM is responsible for computing the values $\{d_k, k = 1,2,\ldots,N\}$ (Eq. 2). The implementation of this equation in DXFRM employs some trigonometric identities for faster execution. The fact that the image is a rectangular array of pixels enables use of these identities. No assumptions are made about the locations of visibility measurements. Modification of DXFRM to exploit regularities in measurement coverage for increased speed is possible.

PSF computes the distinct values of $\{p_{\ell,k}\}$ (Eq. 3) corresponding to

$$(x_\ell - x_k) = n_x \Delta x \qquad 0 \leq n_x \leq C - 1$$

$$(y_\ell - y_k) = n_y \Delta y \qquad -(R - 1) \leq n_y \leq (R - 1)$$

where R and C are, respectively, the numbers of rows and columns in the image. Trig identities are again used for speed. The results of these calculations are stored in row-sequential format; the first element of the output array is associated with $(x_\ell - x_k) = 0$, $(y_\ell - y_k) = (R - 1)\Delta y$ and the last with $(x_\ell - x_k) = (c - 1)\Delta x$, $(y_\ell - y_k) = -(R - 1)\Delta y$.

XFERFN rearranges values computed by PSF and performs a two-dimensional FFT of this array. The result is retained for use by programs in the high-speed convolution group. Pre-computing the FFT of this array reduces convolution costs by about 30%.

Two-dimensional FFT Group

Much of the source code contained in the Appendix is associated with FFT software. The FFT program, CTA (Cooley-Tukey algorithm), is transportable only in its Fortran version, which is significantly slower

than an assembly language implementation of the same algorithm. As many computer installations have FFT subroutines that are locally optimized for speed, modification of MEPIX to replace CTA with a faster FFT realization should be considered. Provided the comments on the CTA source listing are studied carefully, such a substitution is not difficult.

It should be noted that CTA cannot compute the FFT of a sequence of length less than 16. Therefore, use of MEPIX is restricted to images with at least 8 rows and 8 columns, and the present program cannot be applied to one-dimensional problems. Those involved in one-dimensional Fourier analysis are encouraged to revise the subroutines in the overhead and convolution groups. Components in the optimization group need not be altered.

FFTR arranges for efficient transformation when either input or output data arrays are real. FFT2R implements the FFT of a two-dimensional real array as a sequence of one-dimensional transformations. IFFT2R inverts this operation, transforming an array of complex numbers with implicit Hermitian symmetry into a two-dimensional real array.

Two-dimensional High-speed Convolution Group

CONV2D oversees the high-speed convolution operation. Its basic function is to take the inverse FFT of the product of the output of XFERFN and the FFT of the image. To avoid certain wraparound edge effects, padding arrays with zeroes is required, and the actual transformation operations are on arrays with 2R rows and 2C columns. CONV2D takes care of these details automatically.

MULT2D and CMUL are utility programs. The function CMUL implements complex multiplication, and MULT2D performs array multiplication as required

by the high-speed convolution algorithm. The storage format for data processed by FFT2R and IFFT2R is rather complicated (see comments attached to FFT2R), which accounts for the length of MULT2D. This data structure could be simplified considerably.

Optimization Group

The behavior of MEPIX is fundamentally controlled by ITERAT, the subroutine that embodies the iterative search strategy. Although the operations performed by the other subroutines are complex, each operation is well defined, and an improvement that speeds the implementation slightly or that cleans up the code will have relatively little effect on the character of the total maximum entropy imaging package. The bulk of the source code is contained in the convolution and overhead groups, yet these calculations, assuming they are done correctly, are really perfunctory.

Improvements to the optimization strategy contained in ITERAT can, on the other hand, have a profound impact on the program. The iterative scheme described in Section III has evolved gradually over the last 18 months. Experimentation has suggested that this method converges much faster than a steepest ascent algorithm, at least when applied to radio astronomical objects; however, very little is known about why this is so. It is expected that substantial improvements to the present algorithm await discovery. Thus, the user should not hesitate to incorporate in ITERAT changes that appear desirable. This version of ITERAT has worked well for us, but no stronger claims are made.

OBJFTN and GRADI are utility routines that evaluate the objective function value and gradient, respectively. ROOT calculates the optimal

univariate moves as described in Step 5 of Section III. XTREM is not called in the present version of ITERAT.

Each iteration results in several lines of printout from ITERAT, an annotated sample of which is contained in Fig. 4. Much of this information is of interest only to those concerned with the behavior of the iterative search algorithm. The most important parameters for the casual user are the fractional image change, which might be used as a stopping criterion, and the rms reconstruction error

$$\sqrt{\sum_{i=1}^{M} \frac{1}{\sigma_i^2} \left| m_i - \Delta A \sum_{k=1}^{N} f_k \exp[-j2\pi(u_i x_k + v_i y_k)] \right|^2 \Big/ M} \qquad (7)$$

which describes the measurement compatibility of the reconstruction. This parameter, which should be approximately one for the final image, can be used to assess the adequacy of the chosen value of λ. If the reconstruction accuracy is not sufficient, the value of λ should be increased. Conversely, if the reconstruction accuracy is too good, spurious detail may develop in the image, and λ should be reduced.

V. RUN TIME CONSIDERATIONS

MEPIX has been compiled by the Fortran H-Extended compiler (OPT=2), and a load module is stored on disk. The execution time for running this load module is a function of the number of pixels N, the number of measurements M, and the number of iterations I. The function is approximately

$$t = \alpha MN + \beta NI \log_2(4N)$$

where

$$\alpha \simeq 1.4 \times 10^{-5} \text{ sec} \quad \text{and} \quad \beta \simeq 3.7 \times 10^{-5} \text{ sec}$$

```
0=ITERATION -1.2451E-04=OBJECT   7.9994E-05=ENTROPY  5.4313E+00=RMSERR   1.0591E-16=RMSGRAD
4.0=DAMPING -2.0446E-02=OBJTEMP -3.5499E-05=OBJSTAR    9.0613E-01=FRACSTR    27.6=GRADPHI
TAKE OPTIMAL MOVE:   9.2653E+09=DXRMS    1.1346E+09=PRMS      81.56% CHANGE      90.0=DELPHI

1=ITERATION -7.5740E-05=OBJECT   8.1517E-06=ENTROPY  3.1259E+00=RMSERR   6.4119E-17=RMSGRAD
16.1=DAMPING -3.0420E-05=OBJTEMP -3.0236E-05=OBJSTAR    8.8452E-01=FRACSTR    76.7=GRADPHI
KEEP THIS IMAGE:   4.3369E+09=DXRMS    1.4424E+09=PRMS      29.89% CHANGE      45.5=DELPHI

2=ITERATION -3.0420E-05=OBJECT   8.1372E-06=ENTROPY  2.9298E+00=RMSERR   6.6394E-17=RMSGRAD
17.5=DAMPING -2.5537E-05=OBJTEMP -1.9763E-05=OBJSTAR    3.8818E-01=FRACSTR    60.6=GRADPHI
KEEP THIS IMAGE:   8.2201E+07=DXRMS    1.3582E+09=PRMS      6.05% CHANGE      141.7=DELPHI
```

ITERATION: iteration number

OBJECT: objective function for current image

ENTROPY: entropy component of objective function

RMSERR: rms reconstruction error (Eq. 7)

RMSGRAD: rms gradient

DAMPING: damping factor

OBJTEMP: objective function for temporary image

OBJSTAR: predicted maximum objective function in perturbation direction

FRACSTAR: predicted optimal step size

GRADPHI: "angle" between gradient and perturbation direction

DXRMS: rms brightness change from current image to new image

PRMS: rms brightness of new image

CHANGE: ratio of DXRMS to PRMS

DELPHI: "angle" between current and previous perturbation directions

KEEP THIS IMAGE means that the temporary image is retained as the result of the current iteration. TAKE OPTIMAL MOVE indicates quadratic approximation in perturbation direction is used to determine new image.

Fig. 4

are appropriate constants for an IBM 370/168 installation. These parameters have been determined experimentally. β is subject to some fluctuation since it depends on the effectiveness of the temporary image moves, that is, how often the temporary image is retained. Each optimal step costs an additional FFT convolution. The temporary image was retained in 72% of the iterations for the test case that determined the cited value of β.

With arrays dimensioned to handle 4096 pixels and up to 1000 visibility measurements, MEPIX needs 360 K bytes of memory storage. This figure could be reduced somewhat by judicious use of equivalence statements. In its most core-efficient realization, MEPIX requires array storage to accommodate about 11N floating point values.

VI. ACKNOWLEDGMENT

This work was supported by the National Science Foundation under grants DCR 75-15140 and MCS 76-12456, and by the Air Force Cambridge Research Laboratories under Grant F19628-74-C-0053.

APPENDIX M

MEM FORTRAN PROGRAM CODE

```
 LEVEL 2.2 (SEPT 76)                    OS/360  FORTRAN H EXTENDED          DATE 77.031/19.20.17

REQUESTED OPTIONS: NOMAP

OPTIONS IN EFFECT: NAME(MAIN) NOOPTIMIZE LINECOUNT(58) SIZE(MAX) AUTODBL(NONE)
                   SOURCE EBCDIC NOLIST NODECK OBJECT NOMAP NOFORMAT GOSTMT NOXREF ALC NOANSF TERM FLAG(I

                C     TWO-DIMENSIONAL MAXIMUM ENTROPY FOURIER SYNTHESIS:  SAMPLE MAIN
                C                                                         PROGRAM
                C
                C     PROGRAM INTENDED FOR USE WITH IRREGULARLY SPACED VISIBILITIES
                C
                C     MINIMUM ARRAY DIMENSIONS WHERE  NX=NUMBER OF IMAGE COLUMNS
                C                                     NY=NUMBER OF IMAGE ROWS
                C                                     NPIXL=NUMBER OF PIXELS (NX*NY)
                C                                     NMEAS=NUMBER OF VISIBILITIES
                C
                C     HOLD,XFER:  4*NPIXL ELEMENTS
                C     PIX,GRAD,DIRECT,PMOVE:  NPIXL ELEMENTS
                C     VREAL,VIMAG,U,V,VAR:  NMEAS ELEMENTS
                C     XWORK,YWORK,XXWORK,YYWORK:  MAX0(NX,NY) ELEMENTS
                C     IWORK:  MAX0(NY,2*NX)/16+1 ELEMENTS
                C     SINE:  MAX0(NY,2*NX)/8 ELEMENTS
                C     WORK:  MAX0(NY,2*NX)+2 ELEMENTS
                C
                C     BOTH NX AND NY MUST BE A POWER OF TWO.
                C
                C                                                         SJW 12/13/76
                C
   ISN 0002           DIMENSION HOLD(16384),XFER(16384)
   ISN 0003           DIMENSION PIX(4096),GRAD(4096),DIRECT(4096),PMOVE(4096)
   ISN 0004           DIMENSION VREAL(1000),VIMAG(1000),U(1000),V(1000),VAR(1000)
   ISN 0005           DIMENSION XWORK(256),YWORK(256),XXWORK(256),YYWORK(256)
   ISN 0006           DIMENSION IWORK(33),SINE(64),WORK(514)
                C
                C     INPUT AND INITIALIZATION:  2**LOGNR ROWS, 2**LOGNC COLUMNS,
                C                                NITER ITERATIONS, DAMPING FACTOR DAMP
                C                                CONTROL FOR ADAPTIVE DAMPING CHDAMP
                C
   ISN 0007         1 READ (5,100) LOGNR,LOGNC,NITER,DAMP,CHDAMP
   ISN 0008           IF (LOGNR .LT. 0) STOP
   ISN 0010           READ (5,101) XMIN,XMAX,YMIN,YMAX
   ISN 0011           NX=2**LOGNC
   ISN 0012           NY=2**LOGNR
   ISN 0013           LOGNC1=LOGNC+1
   ISN 0014           LOGNR1=LOGNR+1
   ISN 0015           NPIXL=NX*NY
   ISN 0016           NX2=2*NX
   ISN 0017           NY2=2*NY
   ISN 0018           DX=(XMAX-XMIN)/NX
   ISN 0019           DY=(YMAX-YMIN)/NY
   ISN 0020           DA=DX*DY
   ISN 0021           XTOPL=XMIN+DX/2.
   ISN 0022           YTOPL=YMAX-DY/2.
                C
                C     INPUT VISIBILITY DATA AND VARIANCES
   ISN 0023           CALL DATAIN (VREAL,VIMAG,VAR,U,V,NMEAS)
```

X-2

```
                C
                C       COMPUTE WEIGHTED SUM OF SQUARES OF VISIBILITY MAGNITUDES.  MAIN
                C       CALCULATES THIS QUANTITY SO INDIVIDUAL VISIBILITIES ARE NOT NEEDED
                C       BY ITERAT.
   ISN 0024             SQAMP=0.
   ISN 0025             DO 20 I=1,NMEAS
   ISN 0026          20 SQAMP=SQAMP+(VREAL(I)**2+VIMAG(I)**2)/VAR(I)
                C
                C       COMPUTE LAMBDA ACCORDING TO "UNIFORM GRAY" RULE ASSUMING FLUX
                C       DENSITY IS FIRST MEASUREMENT.  AN ALTERNATIVE IS TO ALLOW USER
                C       INPUT OF F_MBDA.
   ISN 0027             FLMBDA=.5*DA*NPIXL/(1.+VREAL(1)/SQRT(ABS(VAR(1))))
                C
                C       INITIALIZE IMAGE TO UNIFORMLY GRAY ASSUMING FLUX DENSITY
                C       IS FIRST MEASUREMENT.
   ISN 0028             GRAY=VREAL(1)/(NPIXL*DA)
   ISN 0029             DO 30 I=1,NPIXL
   ISN 0030          30 PIX(I)=GRAY
                C
                C       FORM RATIOS OF LAMBDA TO VARIANCES
   ISN 0031             DO 40 I=1,NMEAS
   ISN 0032          40 VAR(I)=FLMBDA/VAR(I)
                C
                C       COMPUTE DIRECT RECONSTRUCTION ARRAY
   ISN 0033             CALL DXFRM(DIRECT,NY,DY,YTOPL,NX,DX,XTOPL,VREAL,VIMAG,U,V,VAR,
                     &  NMEAS,XWORK,XXWORK,YWORK,YYWORK)
                C
                C       COMPUTE POINT RESPONSE ARRAY AND TRANSFER FUNCTION.  PLL IS
                C       CENTRAL VALUE OF POINT RESPONSE ARRAY AND IS NEEDED BY ITERAT.
   ISN 0034             CALL PSF(HOLD,NY,DY,NX,DX,U,V,VAR,NMEAS,XWORK,XXWORK,YWORK,YYWORK)
   ISN 0035             PLL=HOLD(NX*(NY-1)+1)
   ISN 0036             CALL XFERFN(HOLD,XFER,LOGNRI,LOGNCI,NY2,NX2,IWORK)
                C
                C       ITERATIVE OPTIMIZATION OF MAXIMUM ENTROPY OBJECTIVE FUNCTION
   ISN 0037             CALL ITERAT(HOLD,XFER,LOGNCI,LOGNRI,PIX,PMOVE,GRAD,DIRECT,NX,NY,
                     &    NX2,NY2,F_MBDA,PLL,SQAMP,NITER,DAMP,CHDAMP,NMEAS,DA,IWORK,SINE,
                     &    WORK)
                C
                C       OUTPUT IMAGE AND RETURN FOR NEXT DATA SET
   ISN 0038             CALL PIXOUT(PIX,NX,NY)
   ISN 0039             GOTO 1
                C
   ISN 0040        100 FORMAT (3I10,2F10.3)
   ISN 0041        101 FORMAT (4F10.4)
   ISN 0042             END
```

*OPTIONS IN EFFECT*NAME(MAIN) NOOPTIMIZE LINECOUNT(58) SIZE(MAX) AUTODBL(NONE)

*OPTIONS IN EFFECT*SOURCE EBCDIC NOLIST NODECK OBJECT NOMAP NOFORMAT GOSTMT NOXREF ALC NOANSF TERM FLAG(I)

STATISTICS SOURCE STATEMENTS = 41, PROGRAM SIZE = 22484, SUBPROGRAM NAME = MAIN

STATISTICS NO DIAGNOSTICS GENERATED

****** END OF COMPILATION ****** 130K BYTES OF CORE NOT USED

```
 LEVEL 2.2  (SEPT 76)                    OS/360  FORTRAN H EXTENDED          DATE 77.031/19.23.23

REQUESTED OPTIONS: NOMAP

OPTIONS IN EFFECT: NAME(MAIN) NOOPTIMIZE LINECOUNT(58) SIZE(MAX) AUTODBL(NONE)
                   SOURCE EBCDIC NOLIST NODECK OBJECT NOMAP NOFORMAT GOSTMT NOXREF ALC NOANSF TERM FLAG(I)

   ISN 0002              SUBROUTINE CMUL (A,B,C,D)
                     C   MULTIPLICATION OF COMPLEX NUMBER A + J*B BY C + J*D WITH
                     C   RESULT STORED IN (A,B).
                     C                                                  SJS 5/1/76
                     C
   ISN 0003              HOLD=A
   ISN 0004              A=A*C-B*D
   ISN 0005              B=HOLD*D+B*C
   ISN 0006              RETURN
   ISN 0007              END

*OPTIONS IN EFFECT*NAME(MAIN) NOOPTIMIZE LINECOUNT(58) SIZE(MAX) AUTODBL(NONE)

*OPTIONS IN EFFECT*SOURCE EBCDIC NOLIST NODECK OBJECT NOMAP NOFORMAT GOSTMT NOXREF ALC NOANSF TERM FLAG(I)

*STATISTICS*     SOURCE STATEMENTS =       6, PROGRAM SIZE =      316, SUBPROGRAM NAME = CMUL

*STATISTICS*  NO  DIAGNOSTICS GENERATED

****** END OF COMPILATION ******                                  130K BYTES OF CORE NOT USED
```

```
LEVEL 2.2  (SEPT 76)                    OS/360  FORTRAN H EXTENDED          DATE 77.031/19.23.27

REQUESTED OPTIONS: NOMAP

OPTIONS IN EFFECT: NAME(MAIN) NOOPTIMIZE LINECOUNT(58) SIZE(MAX) AUTODBL(NONE)
                   SOURCE EBCDIC NOLIST NODECK OBJECT NOMAP NOFORMAT GOSTMT NOXREF ALC NOANSF TERM FLAG(I)

   ISN 0002            SUBROUTINE CONV2D (HOLD,XFER,PIX,NYPWR2,NXPWR2,LOGNR,LOGNC,
                      &   NY,NX,SINE,WORK,IWORK)
                C
                C     THIS SUBROUTINE OVERSEES TWO-DIMENSIONAL HIGH-SPEED CONVOLUTION
                C     ARGUMENTS:
                C
                C     HOLD:  ARRAY CONTAINING CONVOLUTION RESULTS ON RETURN
                C     XFER:  TRANSFER FUNCTION ARRAY RETURNED BY XFERFN
                C     PIX:  IMAGE ARRAY
                C     NYPWR2:  TWO TIMES THE NUMBER OF IMAGE ROWS
                C     NXPWR2:  TWO TIMES THE NUMBER OF IMAGE COLUMNS
                C     LOGNR:  LOG(BASE 2) NYPWR2
                C     LOGNC:  LOG(BASE 2) NXPWR2
                C     NY:  NUMBER OF IMAGE ROWS
                C     NX:  NUMBER OF IMAGE COLUMNS
                C     SINE, WORK,IWORK:  WORK ARRAYS FOR THE FFT SUBROUTINE CTA
                C
                C                                                  SJW 9/26/76
                C
   ISN 0003            DIMENSION HOLD(NYPWR2,NXPWR2),XFER(NYPWR2,NXPWR2),PIX(NY,NX)
   ISN 0004            DIMENSION SINE(1),WORK(1),IWORK(1)
   ISN 0005            DO 210 IC=1,NXPWR2
   ISN 0006            DO 210 IR=1,NYPWR2
   ISN 0007          210 HOLD(IR,IC)=0.
                C     TRANSFER IMAGE TO HOLD AND TRANSFORM
   ISN 0008            DO 220 IC=1,NX
   ISN 0009            DO 220 IR=1,NY
   ISN 0010          220 HOLD(IR,IC)=PIX(IR,IC)
   ISN 0011            CALL FFT2R (HOLD,NYPWR2,-NXPWR2,LOGNR,LOGNC,SINE,WORK,IWORK)
                C     MULTIPLY BY TRANSFER FUNCTION
   ISN 0012            CALL MULT2D (HOLD,XFER,NYPWR2,NXPWR2)
                C     INVERSE TRANSFORM
   ISN 0013            CALL IFFT2R (HOLD,NYPWR2,-NXPWR2,LOGNR,LOGNC,SINE,WORK,IWORK)
   ISN 0014            RETURN
   ISN 0015            END

*OPTIONS IN EFFECT*NAME(MAIN) NOOPTIMIZE LINECOUNT(58) SIZE(MAX) AUTODBL(NONE)

*OPTIONS IN EFFECT*SOURCE EBCDIC NOLIST NODECK OBJECT NOMAP NOFORMAT GOSTMT NOXREF ALC NOANSF TERM FLAG(I)

*STATISTICS*      SOURCE STATEMENTS =       14, PROGRAM SIZE =      1004, SUBPROGRAM NAME =CONV2D

*STATISTICS*  NO  DIAGNOSTICS GENERATED

****** END OF COMPILATION ******                                  130K BYTES OF CORE NOT USED
```

M-5

```
LEVEL 2.2  (SEPT 76)                    OS/360  FORTRAN H EXTENDED          DATE 77.031/19.20.35

REQUESTED OPTIONS: NOMAP

OPTIONS IN EFFECT: NAME(MAIN) NOOPTIMIZE LINECOUNT(58) SIZE(MAX) AUTODBL(NONE)
                   SOURCE EBCDIC NOLIST NODECK OBJECT NOMAP NOFORMAT GOSTMT NOXREF ALC NOANSF TERM FLAG(I)

   ISN 0002            SUBROUTINE CTA (NU,A,INV,S,MD)
                 C
                 C     THIS SUBROUTINE FAST FOURIER TRANSFORMS REAL OR COMPLEX DATA.
                 C     ARGUMENTS:
                 C
                 C     A:  DATA ARRAY WITH N REAL VALUES OR N/2 COMPLEX VALUES.
                 C         DIMENSION OF A MUST BE AT LEAST N+2. THE EXTRA TWO
                 C         LOCATIONS NEEDED BY CTA DURING REAL TRANSFORMS.  NOTE:
                 C         N .GE. 16
                 C     NU:  LOG(BASE 2) N
                 C     INV:  INTEGER WORK ARRAY WITH AT LEAST N/16+1 ELEMENTS
                 C     S:  REAL WORK ARRAY WITH AT LEAST N/8 ELEMENTS
                 C     MD:  TYPE OF TRANSFORM DESIRED.  NOTE:  THE FACTOR 1/N NEEDED
                 C          SO FORWARD AND REVERSE TRANSFORMS CANCEL IS CONTAINED IN
                 C          THE FORWARD DIRECTION.
                 C          MD=3 IS FORWARD TRANSFORM OF REAL DATA RETURNING ONLY NON-
                 C               NEGATIVE FREQUENCIES.  TRANSFORM VALUE AT FREQUENCY I/N
                 C               IS  A(2*I+1) + J A(2*I+2)  FOR I=0 TO N/2.
                 C          MD=2 IS FORWARD TRANSFORM OF N/2 COMPLEX VALUES RETURNING
                 C               FREQUENCY 2*I/N AS  A(2*I+1) + J A(2*I+2) FOR
                 C               I=0 TO N/2-1
                 C          MD=1 INITIALIZES INV AND S AND PERFORMS COMPLEX FORWARD
                 C               TRANSFORM
                 C          MD=0 INITIALIZES INV AND S
                 C          MD=-1 INITIALIZES INV AND S AND PERFORMS INVERSE COMPLEX
                 C               TRANSFORM.  CALL WITH MD=1 FOLLOWED BY CALL WITH MD=-1
                 C               RETURNS ORIGINAL DATA.
                 C          MD=-2 INVERSE COMPLEX TRANSFORM
                 C          MD=-3 INVERSE TRANSFORM RETURNING REAL TIME VALUES ASSUMING
                 C               INPUT DATA HAS IMPLICIT HERMITIAN SYMMETRY.  CALL TO
                 C               CTA WITH MD=3 FOLLOWED BY A CALL WITH MD=-3 RETURNS
                 C               ORIGINAL DATA
                 C
                 C                                          EDITED BY SJ# 5/1/76
                 C
   ISN 0003            DIMENSION A(1),INV(1),S(1)
   ISN 0004            DATA TWOPI/6.2831853/
                 C     INITIALIZATION
   ISN 0005            NT=2**NU
   ISN 0006            NT2=NT/2
   ISN 0007            NT4=NT2/2
   ISN 0008            NT8=NT4/2
   ISN 0009            RNT2=1./NT2
   ISN 0010            THETA=TWOPI/NT
   ISN 0011            S2=SIN(THETA)
   ISN 0012            C2=COS(THETA)
   ISN 0013            IF (IABS(MD)-1) 10,10,17
   ISN 0014         10 NT=NU-3
                 C     SINE TABLE
   ISN 0015            THETA=TWOPI/NT2
   ISN 0016            DO 13 I=1,NT8
```

```
ISN 0017          13 S(I)=SIN(I*THETA)
                C      INVERSE TABLE
ISN 0018               MTL=NT2/8
ISN 0019               LM=1
ISN 0020               INV(1)=0
ISN 0021               DO 15 L=1,MT
ISN 0022               DO 14 J=1,LM
ISN 0023               JJ=J+LM
ISN 0024            14 INV(JJ)=INV(J)+MTL
ISN 0025               MTL=MTL/2
ISN 0026            15 LM=LM*2
ISN 0027            17 IF (MD) 1000,1030,30
ISN 0028          1000 IF (MD .LT. -2) GOTO 1002
                C      PHASE FOR INVERSE TRANSFORM
ISN 0030            25 DO 27 I=2,NT,2
ISN 0031            27 A(I)=-A(I)
ISN 0032               GOTO 35
                C      SCALE FOR FORWARD TRANSFORM
ISN 0033            30 DO 31 I=1,NT
ISN 0034            31 A(I)=A(I)*RNT2
                C
                C
                C      MAIN TRANSFORM BEGINS HERE
ISN 0035            35 L=0
ISN 0036               NS=NT
ISN 0037           100 L=L+1
ISN 0038               IF (NU-L) 300,300,101
ISN 0039           101 NS=NS/2
                C      QUICK LOOP FOR PI
ISN 0040               DO 110 I=1,NS,2
ISN 0041               II=I+1
ISN 0042               J=I+NS
ISN 0043               JJ=J+1
ISN 0044               AA=A(I)
ISN 0045               A(I)=AA+A(J)
ISN 0046               A(J)=AA-A(J)
ISN 0047               AA=A(II)
ISN 0048               A(II)=AA+A(JJ)
ISN 0049               A(JJ)=AA-A(JJ)
ISN 0050           110 CONTINUE
ISN 0051               IF (NT2-NS) 100,100,120
ISN 0052           120 N2S=2*NS
ISN 0053               NSTRT=N2S+1
ISN 0054               NSTP=N2S+NS
                C      QUICK LOOP FOR PI/2
ISN 0055               DO 130 I=NSTRT,NSTP,2
ISN 0056               II=I+1
ISN 0057               J=I+NS
ISN 0058               JJ=J+1
ISN 0059               AA=A(J)
ISN 0060               AB=A(JJ)
ISN 0061               A(J)=A(I)-AB
ISN 0062               A(JJ)=A(II)+AA
ISN 0063               A(I)=A(I)+AB
ISN 0064               A(II)=A(II)-AA
ISN 0065           130 CONTINUE
```

```
ISN 0066            IF (NT2-N25) 100,100,140
ISN 0067        140 NSTRT=NSTRT+N25-1
ISN 0068            NSTP=NSTRT+NS
ISN 0069            IF (NT-NSTRT) 100,100,161
ISN 0070        161 IL=NSTRT/NS
ISN 0071            IQUAD=MOD(IL,4)
ISN 0072            IU=IL/4+1
ISN 0073            IK=INV(IU)
ISN 0074            IF (IQUAD) 170,170,180
                C   FIRST QUADRANT
ISN 0075        170 CO=S(NT8-IK)
ISN 0076            SI=S(IK)
ISN 0077            GOTO 210
                C   SECOND QUADRANT
ISN 0078        180 CO=-S(IK)
ISN 0079            SI=S(NT8-IK)
ISN 0080        210 NSTRT=NSTRT+1
ISN 0081            DO 220 I=NSTRT,NSTP,2
ISN 0082            II=I+1
ISN 0083            J=I+NS
ISN 0084            JJ=J+1
ISN 0085            AA=CO*A(J)+SI*A(JJ)
ISN 0086            AB=CO*A(JJ)-SI*A(J)
ISN 0087            A(J)=A(I)-AA
ISN 0088            A(JJ)=A(II)-AB
ISN 0089            A(I)=A(I)+AA
ISN 0090            A(II)=A(II)+AB
ISN 0091        220 CONTINUE
ISN 0092            GOTO 140
                C
                C   RE-ARRANGE THE DATA NOW
ISN 0093        300 DO 340 N=2,NT4,2
ISN 0094            I=8*INV(N/2)+2
ISN 0095            IF (N-I) 305,310,310
ISN 0096        305 AA=A(N)
ISN 0097            A(N)=A(I)
ISN 0098            A(I)=AA
ISN 0099            AA=A(N-1)
ISN 0100            A(N-1)=A(I-1)
ISN 0101            A(I-1)=AA
ISN 0102        310 I=I+2
ISN 0103            NN=N+NT2
ISN 0104            IF (NN-I) 315,320,320
ISN 0105        315 AA=A(NN)
ISN 0106            A(NN)=A(I)
ISN 0107            A(I)=AA
ISN 0108            AA=A(NN-1)
ISN 0109            A(NN-1)=A(I-1)
ISN 0110            A(I-1)=AA
ISN 0111        320 I=I+2
ISN 0112            NN=N+NT4
ISN 0113            IF (NN-I) 325,330,330
ISN 0114        325 AA=A(NN)
ISN 0115            A(NN)=A(I)
ISN 0116            A(I)=AA
ISN 0117            AA=A(NN-1)
```

```
ISN 0118            A(NN-1)=A(I-1)
ISN 0119            A(I-1)=AA
ISN 0120          330 I=I+2
ISN 0121            NN=NN+NT2
ISN 0122            IF (NN-I) 335,340,340
ISN 0123          335 AA=A(NN)
ISN 0124            A(NN)=A(I)
ISN 0125            A(I)=AA
ISN 0126            AA=A(NN-1)
ISN 0127            A(NN-1)=A(I-1)
ISN 0128            A(I-1)=AA
ISN 0129          340 CONTINUE
ISN 0130            IF (MD) 400,1000,1001
ISN 0131          400 DO 450 I=2,NT,2
ISN 0132          450 A(I)=-A(I)
ISN 0133         1001 IF (MD .LE. 2) GOTO 1030
                 C
                 C     THIS PART MERGES THE DATA FOR REAL INPUTS
ISN 0135         1002 MM=NT+2
ISN 0136            SI=1.
ISN 0137            CO=0.
ISN 0138            IF (MD .LT. 0) GOTO 1021
                 C     SPECIAL CASES FOR FORWARD TRANSFORM
ISN 0140            A(MM-1)=.5*(A(1)-A(2))
ISN 0141            A(1)=.5*(A(1)+A(2))
ISN 0142            A(MM)=0.
ISN 0143            A(2)=0.
ISN 0144            A(NT2+1)=.5*A(NT2+1)
ISN 0145            A(NT2+2)=-.5*A(NT2+2)
ISN 0146            GOTO 1011
                 C     SPECIAL CASES FOR INVERSE TRANSFORM
ISN 0147         1021 A(2)=A(1)-A(MM-1)
ISN 0148            A(1)=A(1)+A(MM-1)
ISN 0149            A(NT2+1)=2.*A(NT2+1)
ISN 0150            A(NT2+2)=-2.*A(NT2+2)
ISN 0151            A(MM)=0.
ISN 0152            A(MM-1)=0.
ISN 0153         1011 JJ=NT2+1
ISN 0154            ICOMP=1000
ISN 0155         1012 ICOMP=-ICOMP
ISN 0156            JJ=NT2-JJ+4
ISN 0157            IF (MOD(JJ,4) .NE. 1) GOTO 1040
ISN 0159            II=JJ/4
ISN 0160            SI=S(II)
ISN 0161            CC=S(NT8-II)
ISN 0162            GOTO 1041
ISN 0163         1040 SA=CO*CC+SI*S2
ISN 0164            CC=SI*CC-CO*S2
ISN 0165            SI=SA
ISN 0166         1041 J=JJ+1
ISN 0167         1017 MJJ=MM-JJ
ISN 0168            MJ=MM-J+2
ISN 0169            AR=A(JJ)+A(MJJ)
ISN 0170            AI=A(J)-A(MJ)
ISN 0171            E=A(J)+A(MJ)
ISN 0172            EI=A(JJ)-A(MJJ)
```

```
LEVEL 2.2  (SEPT 76)          CTA          05/360  FORTRAN H EXTENDED          DATE 77.031/19.23.35

   ISN 0173                IF (MD) 1015,1015,1014
                     C     FOR*ARD TRANSFORM
   ISN 0174           1014 BR=B*CC-BI*SI
   ISN 0175                BI=-B*SI-BI*CB
   ISN 0176                A(JJ)=.25*(AR+BR)
   ISN 0177                A(J)=.25*(AI+BI)
   ISN 0178                A(MJJ)=.25*(AR-BR)
   ISN 0179                A(MJ)=.25*(BI-AI)
   ISN 0180                GOTO 1016
                           INVERSE TRANSFORM
   ISN 0181           1015 BR=BI*CC-B*SI
   ISN 0182                BI=BI*SI+B*CB
   ISN 0183                A(JJ)=AR-BI
   ISN 0184                A(J)=AI+BR
   ISN 0185                A(MJJ)=AR+BI
   ISN 0186                A(MJ)=BR-AI
   ISN 0187           1016 IF (ICOMP .GE. 0) GOTO 1012
                     C     SWITCH SI AND CO TO JJ THE COMPLEMENT ANGLE
   ISN 0189                ICOMP=-ICOMP
   ISN 0190                IF (JJ .GT. NT4) GOTO 1010
   ISN 0192                SA=SI
   ISN 0193                SI=CB
   ISN 0194                CC=SA
   ISN 0195                JJ=NT2-JJ+2
   ISN 0196                GOTO 1041
   ISN 0197           1010 IF (MD .LT. 0) GOTO 25
   ISN 0199            999 CONTINUE
   ISN 0200           1030 RETURN
   ISN 0201                END

*OPTIONS IN EFFECT*NAME(MAIN) NOOPTIMIZE LINECOUNT(58) SIZE(MAX) AUTODBL(NONE)

*OPTIONS IN EFFECT*SOURCE EBCDIC NOLIST NODECK OBJECT NOMAP NOFORMAT GOSTMT NOXREF ALC NOANSF TERM FLAG(I)

*STATISTICS*      SOURCE STATEMENTS =      200, PROGRAM SIZE =      4880, SUBPROGRAM NAME =   CTA

*STATISTICS*  NO  DIAGNOSTICS GENERATED

*****  END OF COMPILATION *****                                    98K BYTES OF CORE NOT USED
```

```
 LEVEL 2.2  (SEPT 76)                 05/360  FORTRAN H EXTENDED          DATE 77.031/19.22.16

REQUESTED OPTIONS: NOMAP

OPTIONS IN EFFECT: NAME(MAIN) NOOPTIMIZE LINECOUNT(58) SIZE(MAX) AUTODBL(NONE)
                   SOURCE EBCDIC NOLIST NODECK OBJECT NOMAP NOFORMAT GOSTMT NOXREF ALC NOANSF TERM FLAG(I)

   ISN 0002            SUBROUTINE DXFRM( DI,NROW,DELY,YTOPL,NCOL,DELX,XTOPL,FREAL,FIMAG,
                      &    U,V,WT,NMEAS,XCOS,XSIN,YCOS,YSIN )
   ISN 0003            DIMENSION DI(1),FREAL(1),FIMAG(1),U(1),V(1),WT(1)
   ISN 0004            DIMENSION XCOS(1),XSIN(1),YCOS(1),YSIN(1)
   ISN 0005            REAL*8 DCOS,DSIN,DARG,COS1,SIN1,COS2,SIN2
   ISN 0006            DATA TWOPI/6.2831853/
                     C
                     C     THIS SUBROUTINE COMPUTES THE DIRECT FOURIER RECONSTRUCTION OF
                     C     VISIBILITY MEASUREMENTS WITH NORMALIZATION TO SUIT MAXIMUM
                     C     ENTROPY RECONSTRUCTION.  THE FUNCTION EVALUATED IS
                     C
                     C     C(L) = 2*SUM( I=1,NMEAS) WT(I)*AMP(I)*COS(2*PI*(U(I)*XL+V(I)*YL)
                     C                                                 +PHASE(I))
                     C
                     C     WHERE ONE-DIMENSIONAL COLUMN-SEQUENTIAL REPRESENTATION IS USED
                     C     AND (XL,YL) ARE THE COORDINATES OF THE LTH PIXEL.  THIS EQUATION
                     C     IS COMPUTED FOR 1<=L<=NROW*NCOL.  OTHER PARAMETER INFORMATION:
                     C
                     C     DI: ARRAY OF DIRECT RECONSTRUCTION VALUES
                     C     NROW:  NUMBER OF ROWS IN THE IMAGE.
                     C     DELY:  DISTANCE BETWEEN ADJACENT ROWS.
                     C     YTOPL:  Y-COORDINATE OF TOP LEFT PIXEL
                     C     NCOL:  NUMBER OF COLUMNS IN THE IMAGE.
                     C     DELX:  DISTANCE BETWEEN ADJACENT COLUMNS.
                     C     XTOPL:  X-COORDINATE OF TOP LEFT PIXEL.
                     C     FREAL:  VISIBILITY REAL PARTS
                     C     FIMAG:  VISIBILITY IMAGINARY PARTS.
                     C     U,V:  ARRAYS OF MEASUREMENT LOCATIONS IN THE SPATIAL FREQUENCY
                     C         DOMAIN.
                     C     WT:  ARRAY OF RECONSTRUCTION ERROR WEIGHTS (LAMBDA/VARIANCE).
                     C          IF WT(1)<0, ALL WEIGHTS ARE TAKEN TO BE ABS(WT(1)).
                     C     NMEAS:  NUMBER OF COMPLEX VISIBILITY MEASUREMENTS
                     C     XCOS,XSIN:  WORK ARRAY WITH AT LEAST NCOL ELEMENTS IN EACH
                     C     YCOS,YSIN:  WORK ARRAYS WITH AT LEAST NROW ELEMENTS IN EACH
                     C
                     C                                                 SJW 5/1/76
                     C
                     C     MODIFIED OCTOBER 1976 BY CJ GREBENKEMPER FOR FASTER EXECUTION
                     C     BY USING TRIG IDENTITIES IN THE DIRECT FOURIER TRANSFORM.
                     C
                     C        ZERO DIRECT RECONSTRUCTION ARRAY.
   ISN 0007            NPIXL = NROW*NCOL
   ISN 0008            DO 20 K=1,NPIXL
   ISN 0009         20 DI(K) = 0.
                     C
                     C        TAKE EACH MEASUREMENT ONE AT A TIME.
   ISN 0010            DO 70 I=1,NMEAS
                     C
                     C        CALCULATE REAL AND IMAGINARY PARTS OF WEIGHTED VISIBILITY.
   ISN 0011            WEIGHT = -WT(1)
   ISN 0012            IF( WEIGHT .EQ. 0. )  WEIGHT = 1.
```

```
LEVEL 2.2  (SEPT 76)          DXFRM          OS/360  FORTRAN H EXTENDED          DATE 77.031/19.22.16

   ISN 0014              IF( WEIGHT .LT. 0. )  WEIGHT = WT(I)
   ISN 0016              VR = 2.*WEIGHT*FREAL(I)
   ISN 0017              VI = 2.*WEIGHT*FIMAG(I)
                    C
                    C        FILL UP COLUMN SINE/COSINE TABLE.
   ISN 0018              DARG = TWOPI*U(I)*( XTUPL-DELX )
   ISN 0019              CCSI = DCOS(DARG)
   ISN 0020              SINI = DSIN(DARG)
   ISN 0021              DARG = TWOPI*U(I)*DELX
   ISN 0022              COS2 = DCOS(DARG)
   ISN 0023              SIN2 = DSIN(DARG)
   ISN 0024              DO 30 IX=1,NCOL
   ISN 0025              DARG = COSI*COS2-SINI*SIN2
   ISN 0026              SINI = COSI*SIN2+SINI*COS2
   ISN 0027              COSI = DARG
   ISN 0028              XCOS(IX) = COSI
   ISN 0029              XSIN(IX) = SINI
   ISN 0030        30    CONTINUE
                    C
                    C        FILL UP ROW SINE/COSINE TABLE.
   ISN 0031              DARG = TWOPI*V(I)*( YTUPL+DELY )
   ISN 0032              CCSI = DCOS(DARG)
   ISN 0033              SINI = DSIN(DARG)
   ISN 0034              DARG = -TWOPI*V(I)*DELY
   ISN 0035              COS2 = DCOS(DARG)
   ISN 0036              SIN2 = DSIN(DARG)
   ISN 0037              DO 40 IY=1,NROW
   ISN 0038              DARG = COSI*COS2-SINI*SIN2
   ISN 0039              SINI = COSI*SIN2+SINI*COS2
   ISN 0040              COSI = DARG
   ISN 0041              YCOS(IY) = COSI
   ISN 0042              YSIN(IY) = SINI
   ISN 0043        40    CONTINUE
                    C
                    C        DIRECT FOURIER TRANSFORM FOR THE ITH ELEMENT.
   ISN 0044              DO 60 IX=1,NCOL
   ISN 0045              NDXH = (IX-1)*NROW
   ISN 0046              XC = XCOS(IX)
   ISN 0047              XS = XSIN(IX)
   ISN 0048              DO 50 IY=1,NROW
   ISN 0049              NDX = NDXH+IY
   ISN 0050              YC = YCOS(IY)
   ISN 0051              YS = YSIN(IY)
   ISN 0052              DI(NDX) = DI(NDX) + VR*(XC*YC-XS*YS) - VI*(XC*YS+XS*YC)
   ISN 0053        50    CONTINUE
   ISN 0054        60    CONTINUE
   ISN 0055        70    CONTINUE
   ISN 0056              RETURN
   ISN 0057              END

*OPTIONS IN EFFECT*NAME(MAIN) NOOPTIMIZE LINECOUNT(58) SIZE(MAX) AUTODBL(NONE)

*OPTIONS IN EFFECT*SOURCE EBCDIC NOLIST NODECK OBJECT NOMAP NOFORMAT GOSTMT NOXREF ALC NOANSF TERM FLAG(I)

*STATISTICS*      SOURCE STATEMENTS =       58, PROGRAM SIZE =      1934, SUBPROGRAM NAME = DXFRM

*STATISTICS*  NO  DIAGNOSTICS GENERATED
```

```
 LEVEL 2.2  (SEPT 76)                    OS/360  FORTRAN H EXTENDED          DATE 77.031/19.20.52

REQUESTED OPTIONS: NOMAP

OPTIONS IN EFFECT: NAME(MAIN) NOOPTIMIZE LINECOUNT(58) SIZE(MAX) AUTODBL(NONE)
                   SOURCE EBCDIC NOLIST NODECK OBJECT NOMAP NOFORMAT GOSTMT NOXREF ALC NOANSF TERM FLAG(I)

   ISN 0002             SUBROUTINE FFTR (ARRAY,NEL,LOGNEL,SINE,IWORK,MODE)
   ISN 0003             DIMENSION ARRAY(1),SINE(1),IWORK(1)
                C       IF IMODE .GT. 2, THE REAL PACKED ARRAY WITH NEL (=2**LOGNEL)
                C       ELEMENTS IS TRANSFORMED BY CTA.  NOTE, ARRAY MUST BE DIMENSIONED
                C       TO HAVE AT LEAST NEL+2 ELEMENTS.  RESULTS AT NONNEGATIVE
                C       FREQUENCIES ARE STORED IN NEL ELEMENTS OF ARRAY WITH
                C          ARRAY(1)=>DC
                C          ARRAY(2)=>NYQUIST
                C          ARRAY(2*I+1) + J*ARRAY(2*I+2)=>I/(NEL*DELTA T).
                C       SINE AND IWORK ARE WORK ARRAYS WITH AT LEAST NEL/8 AND NEL/16+1
                C       ELEMENTS RESPECTIVELY (SEE CTA DESCRIPTION).
                C       IF IMODE .LT. -2, THE COMPLEX (FREQUENCY) ARRAY WITH FORMAT AS
                C       PREVIOUSLY DESCRIBED IS INVERSE FFT'ED INTO NEL REAL VALUES.
                C
                C                                                       SJW 5/1/76
                C
   ISN 0004             IF (IMODE .GT. 2) GOTO 20
   ISN 0006             ARRAY(NEL+1)=ARRAY(2)
   ISN 0007             ARRAY(2)=0.
   ISN 0008             ARRAY(NEL+2)=0.
   ISN 0009          20 CALL CTA (LOGNEL,ARRAY,IWORK,SINE,IMODE)
   ISN 0010             IF (IMODE .LT. -2) RETURN
   ISN 0012             ARRAY(2)=ARRAY(NEL+1)
   ISN 0013             ARRAY(NEL+1)=0.
   ISN 0014             ARRAY(NEL+2)=0.
   ISN 0015             RETURN
   ISN 0016             END

*OPTIONS IN EFFECT*NAME(MAIN) NOOPTIMIZE LINECOUNT(58) SIZE(MAX) AUTODBL(NONE)

*OPTIONS IN EFFECT*SOURCE EBCDIC NOLIST NODECK OBJECT NOMAP NOFORMAT GOSTMT NOXREF ALC NOANSF TERM FLAG(I)

*STATISTICS*      SOURCE STATEMENTS =       15, PROGRAM SIZE =       560, SUBPROGRAM NAME = FFTR

*STATISTICS*  NO  DIAGNOSTICS GENERATED

****** END OF COMPILATION ******                                  130K BYTES OF CORE NOT USED
```

```
REQUESTED OPTIONS: NOMAP

OPTIONS IN EFFECT: NAME(MAIN) NOOPTIMIZE LINECOUNT(58) SIZE(MAX) AUTODBL(NONE)
                   SOURCE EBCDIC NOLIST NODECK OBJECT NOMAP NOFORMAT GOSTMT NOXREF ALC NOANSF TERM FLAG(I)

ISN 0002            SUBROUTINE FFT2R (DATA,NROW,NCOL,LOGNR,LOGNC,SINE,WORK,IWORK)
ISN 0003            DIMENSION DATA(NROW,1),SINE(1),WORK(1),IWORK(1)
                C   TWO-DIMENSIONAL FORWARD FFT OF REAL DATA(NROW,NCOL) WHERE
                C   NROW=2**LOGNR AND 2**LOGNC=NCOL. WITH PACKED, NON-REDUNDANT
                C   OUTPUT IN DATA.
                C   SINE IS REAL WORK ARRAY WITH MAX(NROW,2*NCOL)/8 ELEMENTS.
                C   WORK IS REAL WORK ARRAY WITH MAX(NROW,2*NCOL)+2 ELEMENTS.
                C   IWORK IS INTEGER WORK ARRAY WITH MAX(NROW,2*NCOL)/16+1 ELEMENTS.
                C   NOTE:  NROW,NCOL .GE. 16
                C   OUTPUT FORMAT ACCORDING TO FREQUENCIES:
                C   ROW 1 IS TRANSFORM ALONG U AXIS (V=0)
                C      DATA(1,1)=DC
                C      DATA(1,2) => U=1/(2*DELTA X)
                C      DATA(1,2*I+1) + J*DATA(1,2*I+2) => U = I/(NCOL*DELTA X)
                C   ROW 2 IS TRANSFORM ALONG V=U/(2*DELTA Y) (VERTICAL NYQUIST FREQ).
                C      U CORRESPONDENCE IS AS FOR ROW 1.
                C   ROWS 2*I+1 AND 2*I+2 TRANSFORM ALONG V = I/(NROW*DELTA Y)
                C      DATA(2*I+1,K) + J*DATA(2*I+2,K) CORRESPONDS TO
                C      U = (K-1)/(NCOL*DELTA X).   K IS IN RANGE (1,NCOL) INCLUSIVE.
                C
                C   IF RIGHT HALF COLUMNS OF DATA ARE ZERO (AS IN ARRAYS PADDED WITH
                C   ZEROES TO AVOID WRAP-AROUND), CALL WITH NCOL=-(ABS(NCOL)
                C   FOR A COMPUTATIONAL SAVINGS APPROACHING 25%.
                C
                C                                                        SJW 5/1/76
                C
                C      CC REAL COLUMN TRANSFORMS
ISN 0004            IHOLD=NCOL
ISN 0005            NCEND=NCOL
ISN 0006            IF (NCOL .GT. 0) GOTO 5
ISN 0007            NCOL=IABS(NCOL)
ISN 0008            NCEND=NCOL/2
ISN 0009          5 CALL CTA (LOGNR,WORK,IWORK,SINE,0)
ISN 0010            DO 30 IC=1,NCEND
ISN 0011            DO 10 JR=1,NROW
ISN 0012         10 WORK(JR)=DATA(JR,IC)
ISN 0013            CALL FFTR (WORK,NROW,LOGNR,SINE,IWORK,3)
ISN 0014            DO 20 JR=1,NROW
ISN 0015         20 DATA(JR,IC)=WORK(JR)
ISN 0016         30 CONTINUE
                C
                C   DO COMPLEX ROW TRANSFORMS
ISN 0017            CALL CTA (LOGNC+1,WORK,IWORK,SINE,0)
                C   ROWS 1 AND 2 ARE SPECIAL...DO LATER
ISN 0018            DO 60 IR=3,NROW,2
ISN 0019            DO 40 JC=1,NCOL
ISN 0020            WORK(2*JC-1)=DATA(IR,JC)
ISN 0021         40 WORK(2*JC)=DATA(IR+1,JC)
ISN 0022            CALL CTA (LOGNC+1,WORK,IWORK,SINE,2)
ISN 0023            DO 50 JC=1,NCOL
ISN 0024            DATA(IR,JC)=WORK(2*JC-1)
ISN 0025         50 DATA(IR+1,JC)=WORK(2*JC)
ISN 0026         60 CONTINUE
                C   ROWS 1 AND 2
ISN 0027            DO 70 JC=1,NCOL
ISN 0028            WORK(2*JC-1)=DATA(1,JC)
ISN 0029         70 WORK(2*JC)=DATA(2,JC)
ISN 0030            CALL CTA (LOGNC+1,WORK,IWORK,SINE,2)
ISN 0031            DO 80 JC=1,NCOL
ISN 0032            DATA(1,JC)=WORK(2*JC-1)
ISN 0033         80 DATA(2,JC)=WORK(2*JC)
ISN 0034            IF (IHOLD .GT. 0) RETURN
ISN 0035            NCOL=IHOLD
ISN 0036            DO 90 IR=1,NROW
ISN 0037            DO 90 JC=NCEND+1,NCOL
ISN 0038         90 DATA(IR,JC)=0.0
ISN 0039            RETURN
ISN 0040            END
```

```
LEVEL 2.2  (SEPT 76)          FFT2R          OS/360  FORTRAN H EXTENDED          DATE 77.031/19.21.02

   ISN 0026          50 DATA (IR+1,JC)=WORK(2*JC)
   ISN 0027          60 CONTINUE
                      C
   ISN 0028             CALL CTA (LOGNC,WORK,TWORK,SINE,0)
   ISN 0029             DO 70 IC=1,NCOL
   ISN 0030          70 WORK(IC)=DATA(1,IC)
   ISN 0031             CALL FFTR (WORK,NCOL,LOGNC,SINE,TWORK,3)
   ISN 0032             DO 80 IC=1,NCOL
   ISN 0033             DATA(1,IC)=WORK(IC)
   ISN 0034          80 WORK(IC)=DATA(2,IC)
   ISN 0035             CALL FFTR (WORK,NCOL,LOGNC,SINE,TWORK,3)
   ISN 0036             DO 90 IC=1,NCOL
   ISN 0037          90 DATA(2,IC)=WORK(IC)
   ISN 0038             NCOL=IHOLD
   ISN 0039             RETURN
   ISN 0040             END

*OPTIONS IN EFFECT*NAME(MAIN) NOOPTIMIZE LINECOUNT(58) SIZE(MAX) AUTODBL(NONE)

*OPTIONS IN EFFECT*SOURCE EBCDIC NOLIST NODECK OBJECT NOMAP NOFORMAT GOSTMT NOXREF ALC NOANSF TERM FLAG(I)

*STATISTICS*      SOURCE STATEMENTS =       39, PROGRAM SIZE =      1864, SUBPROGRAM NAME = FFT2R

*STATISTICS*  NO DIAGNOSTICS GENERATED

****** END OF COMPILATION ******                                  322K BYTES OF CORE NOT USED
```

```
 LEVEL 2.2  (SEPT 76)                    OS/360  FORTRAN H EXTENDED          DATE 77.031/19.21.15

REQUESTED OPTIONS: NOMAP

OPTIONS IN EFFECT: NAME(MAIN) NOOPTIMIZE LINECOUNT(58) SIZE(MAX) AUTODBL(NONE)
                   SOURCE EBCDIC NOLIST NODECK OBJECT NOMAP NOFORMAT GOSTMT NOXREF ALC NOANSF TERM FLAG(I)

   ISN 0002            SUBROUTINE GRADI(HOLD,NRBIG,PIX,DIRECT,GRAD,NY,NX,DA,GMAG)
                     C
                     C     THIS SUBROUTINE CALCULATES THE OBJECTIVE FUNCTION GRADIENT.
                     C     ARGUMENTS:
                     C
                     C     HOLD:  ARRAY CONTAINING CONVOLUTION RESULTS
                     C     NRBIG:  NUMBER OF ROWS IN HOLD (2*NY)
                     C     PIX:  IMAGE ARRAY
                     C     DIRECT:  DIRECT TRANSFORM ARRAY
                     C     GRAD:  GRADIENT RETURNED HERE
                     C     NY:  NUMBER OF IMAGE ROWS
                     C     NX:  NUMBER OF IMAGE COLUMNS
                     C     DA:  PIXEL AREA
                     C     GMAG:  GRADIENT MAGNITUDE RETURNED HERE
                     C
                     C                                                    SJW 12/13/76
                     C
   ISN 0003            DIMENSION HOLD(NRBIG,1),PIX(NY,1),DIRECT(NY,1),GRAD(NY,1)
   ISN 0004            GMAG=0.
   ISN 0005            DO 250 IC=1,NX
   ISN 0006            DO 250 IR=1,NY
   ISN 0007            GRAD(IR,IC)=DA*(1./PIX(IR,IC)+DIRECT(IR,IC)-DA*HOLD(IR,IC))
   ISN 0008        250 GMAG=GMAG+GRAD(IR,IC)**2
   ISN 0009            GMAG=SQRT(GMAG)
   ISN 0010            RETURN
   ISN 0011            END

*OPTIONS IN EFFECT*NAME(MAIN) NOOPTIMIZE LINECOUNT(58) SIZE(MAX) AUTODBL(NONE)

*OPTIONS IN EFFECT*SOURCE EBCDIC NOLIST NODECK OBJECT NOMAP NOFORMAT GOSTMT NOXREF ALC NOANSF TERM FLAG(I)

*STATISTICS*      SOURCE STATEMENTS =       10, PROGRAM SIZE =       804, SUBPROGRAM NAME = GRADI

*STATISTICS*  NO  DIAGNOSTICS GENERATED

*****  END OF COMPILATION, *****                          130K BYTES OF CORE NOT USED
```

```
 LEVEL 2.2  (SEPT 76)                    OS/360  FORTRAN H EXTENDED          DATE 77.031/19.21.26

REQUESTED OPTIONS: NOMAP

OPTIONS IN EFFECT: NAME(MAIN) NOOPTIMIZE LINECOUNT(50) SIZE(MAX) AUTODBL(NONE)
                   SOURCE EBCDIC NOLIST NODECK OBJECT NOMAP NOFORMAT GOSTMT NOXREF ALC NOANSF TERM FLAG(I)

   ISN 0002            SUBROUTINE IFFT2R (DATA,NROW,NCOL,LOGNR,LOGNC,SINE,WORK,IWORK)
   ISN 0003            DIMENSION DATA(NROW,1),SINE(1),WORK(1),IWORK(1)
                  C    IFFT2R RE-TRANSFORMS PACKED OUTPUT OF FFT2R INTO ORIGINAL
                  C    REAL ARRAY.  SEE FFT2R COMMENTS FOR FORMATS AND ARGUMENTS.
                  C
                  C    IF RIGHT HALF COLUMNS OF RESULT ARE NOT NEEDED CALL WITH
                  C    NCOL=-IABS(NCOL) FOR COMPUTATIONAL SAVINS APPROACHING 25%.
                  C
                  C                                                    SJW 5/1/76
                  C
                  C    CC COMPLEX INVERSE ROW TRANSFORMS
   ISN 0004            IHOLD=NCOL
   ISN 0005            NCEND=NCOL
   ISN 0006            IF (NCOL .GT. 0) GOTO 5
   ISN 0008            NCOL=IABS(NCOL)
   ISN 0009            NCEND=NCOL/2
   ISN 0010          5 CALL CTA (LOGNC,WORK,IWORK,SINE,0)
   ISN 0011            DO 10 IC=1,NCOL
   ISN 0012         10 WORK(IC)=DATA(1,IC)
   ISN 0013            CALL FFTR (WORK,NCOL,LOGNC,SINE,IWORK,-3)
   ISN 0014            DO 20 IC=1,NCOL
   ISN 0015            DATA(1,IC)=WORK(IC)
   ISN 0016         20 WORK(IC)=DATA(2,IC)
   ISN 0017            CALL FFTR(WORK,NCOL,LOGNC,SINE,IWORK,-3)
   ISN 0018            DO 30 IC=1,NCOL
   ISN 0019         30 DATA(2,IC)=WORK(IC)
   ISN 0020            CALL CTA (LOGNC+1,WORK,IWORK,SINE,0)
   ISN 0021            DO 60 IR=3,NROW,2
   ISN 0022            DO 40 JC=1,NCOL
   ISN 0023            WORK(2*JC-1)=DATA(IR,JC)
   ISN 0024         40 WORK(2*JC)=DATA(IR+1,JC)
   ISN 0025            CALL CTA (LOGNC+1,WORK,IWORK,SINE,-2)
   ISN 0026            DO 50 JC=1,NCOL
   ISN 0027            DATA(IR,JC)=WORK(2*JC-1)
   ISN 0028         50 DATA(IR+1,JC)=WORK(2*JC)
   ISN 0029         60 CONTINUE
                  C
                  C    DC COLUMN INVERSE TRANSFORMS (TO REAL)
   ISN 0030            CALL CTA (LOGNR,WORK,IWORK,SINE,0)
   ISN 0031            DO 90 IC=1,NCEND
   ISN 0032            DO 70 JR=1,NROW
   ISN 0033         70 WORK(JR)=DATA(JR,IC)
   ISN 0034            CALL FFTR(WORK,NROW,LOGNR,SINE,IWORK,-3)
   ISN 0035            DO 80 JR=1,NROW
   ISN 0036         80 DATA(JR,IC)=WORK(JR)
   ISN 0037         90 CONTINUE
   ISN 0038            NCOL=IHOLD
   ISN 0039            RETURN
   ISN 0040            END

*OPTIONS IN EFFECT*NAME(MAIN) NOOPTIMIZE LINECOUNT(58) SIZE(MAX) AUTODBL(NONE)
```

```
REQUESTED OPTIONS: NOMAP

OPTIONS IN EFFECT: NAME(MAIN) NOOPTIMIZE LINECOUNT(58) SIZE(MAX) AUTODBL(NONE)
                   SOURCE EBCDIC NOLIST NODECK OBJECT NOMAP NOFORMAT GOSTMT NOXREF ALC NOANSF TERM FLAG(I)

ISN 0002            SUBROUTINE ITERAT( HOLD,XFER,LOGNC,LOGNR,PIX,PMOVE,GRAD,DIRECT,
                   $    NX,NY,NXPWR2,NYPWR2,FLMBDA,PLL,SJAMP,NITER,DAMP,CHDAMP,
                   $    NMEAS,DA,IWORK,SINE,WORK)
                  C
                  C     THIS SUBROUTINE DEFINES THE ITERATIVE STRATEGY FOR MAXIMUM ENTROPY
                  C     RECONSTRUCTION.  ARGUMENTS:
                  C
                  C     HOLD:  WORK ARRAY NEEDED TO RETURN CONVOLUTION RESULTS
                  C     XFER:  TRANSFER FUNCTION ARRAY RETURNED BY XFERFN
                  C     LOGNC:  LOG(BASE 2) (NUMBER OF IMAGE COLUMNS)
                  C     LOGNR:  LOG(BASE 2) (NUMBER OF IMAGE ROWS)
                  C     PIX:  IMAGE ARRAY
                  C     GRAD:  ARRAY FOR OBJECTIVE FUNCTION GRADIENT
                  C     DIRECT:  DIRECT TRANSFORM ARRAY COMPUTED BY DXFRM
                  C     NX:  NUMBER OF IMAGE COLUMNS
                  C     NY:  NUMBER OF IMAGE ROWS
                  C     NXPWR2:  2**NX
                  C     NYPWR2:  2**NY
                  C     FLMBDA:  USER-SELECTED PARAMETER LAMBDA
                  C     PLL:  CENTRAL VALUE OF POINT RESPONSE ARRAY CALCULATED BY PSF
                  C     SQAMP:  SUM OF SQUARED VISIBILITY MAGNITUDES EACH WEIGHTED
                  C             INVERSELY BY CORRESPONDING VARIANCE.
                  C     NITER:  NUMBER OF ITERATIONS
                  C     DAMP:  USER-SPECIFIED DAMPING FACTOR .GE. 1.
                  C     CHDAMP:  PARAMETER TO HARD LIMIT CHANGES IN THE ADAPTIVE SCHEME
                  C              FOR UPDATING DAMP.  CHDAMP=0. TURNS OFF ADAPTATION.
                  C     NMEAS:  NUMBER OF COMPLEX VISIBILITY MEASUREMENTS
                  C     DA:  PIXEL AREA
                  C     IWORK,SINE,WORK:  WORK ARRAYS FOR FFT SUBROUTINES.
                  C
                  C     ITERATIVE MAXIMIZATION OF MAXIMUM ENTROPY OBJECTIVE FUNCTION
                  C     BY A MULTIVARIATE SEARCH METHOD BASED ON OPTIMAL UNIVARIATE
                  C     MOVE CALCULATIONS.
                  C
                  C                                                    SJW 9/26/76
                  C
ISN 0003            DIMENSION HOLD(NYPWR2,NXPWR2),XFER(NYPWR2,NXPWR2),PIX(NY,NX)
ISN 0004            DIMENSION DIRECT(NY,1),GRAD(NY,1),PMOVE(NY,1)
ISN 0005            DIMENSION IWORK(1),SINE(1),WORK(1)
ISN 0006            DAMP = DAMP
ISN 0007            RTDA=.5/(DA*FLL)
ISN 0008            FOURAC=2./RTDA
ISN 0009            ITER=0
ISN 0010            NPIXL=NX*NY
ISN 0011            DELMAG = 1.
ISN 0012            DO 50 IC=1,NX
ISN 0013            DO 50 IR=1,NY
ISN 0014      50    PMOVE(IR,IC) = 0.
                  C
                  C*************************************************************
                  C
```

```
C     ITERATION LOOP BEGINS HERE.  FIRST DO 2-D CONVOLUTION
C
C*****************************************************************
C
ISN 0015      100 CALL CONV2D (HOLD,XFER,PIX,NYPWR2,NXPWR2,LOGNR,LOGNC,
     &    NY,NX,SINE,WORK,IWORK)
C
C     EVALUATE OBJECTIVE FUNCTION
ISN 0016       CALL OBJFTN (HOLD,NYPWR2,PIX,DIRECT,NY,NX,DA,NMEAS,FLMGDA,
     &    SUAMP,OBJN,ENTRPY,RMSERR)
C
C     CALCULATE GRADIENT, OUTPUT SUMMARY AND PREPARE FOR NEXT ITERATION
ISN 0017      120 CALL GRAD1(HOLD,NYPWR2,PIX,DIRECT,GRAD,NY,NX,DA,GMAG)
ISN 0018       GRMS = SQRT( GMAG*GMAG/NPIXL )
ISN 0019       WRITE(6,900) ITER,OBJN,ENTRPY,RMSERR,GRMS
ISN 0020       IF (ITER .GE. NITER) RETURN
ISN 0022       ITER=ITER+1
C
C-----------------------------------------------------------------
C     MULTIVARIATE SEARCH DIRECTION DETERMINED BY OPTIMAL UNIVARIATE
C     STEP CALCULATIONS
C
ISN 0023      200  DMPINV = 1./DMP
ISN 0024       PREMAG = DELMAG
ISN 0025       DELMAG = 0.
ISN 0026       DELXG = 0.
ISN 0027       DELXP = 0.
ISN 0028       GRMS = 0.
ISN 0029       PRMS=0.
ISN 0030       DO 220 IC=1,NX
ISN 0031       DO 220 IR=1,NY
C
C     CALCULATE OPTIMAL MOVE FOR PIX(IR,IC)
ISN 0032       TEMP=ROOT(DIRECT(IR,IC),HOLD(IR,IC),PIX(IR,IC),DA,PLL,RTNDA,
     &    FOURAC)
C
C        CALCULATE CHANGE IN MOVE; DAMP THE MOVE.
ISN 0033       DELTA = TEMP-PIX(IR,IC)
ISN 0034       DELTA = DELTA*DMPINV
C
C        SAVE MOVE, DOT PRODUCTS BETWEEN MOVE AND GRADIENT AND THE
C        MOVE AND THE PREVIOUS MOVE.
ISN 0035       DELXG = DELXG + DELTA*GRAD(IR,IC)
ISN 0036       DELXP = DELXP + DELTA*PMOVE(IR,IC)
ISN 0037       DELMAG = DELMAG + DELTA*DELTA
ISN 0038       PMOVE(IR,IC) = DELTA
ISN 0039       PIX(IR,IC) = PIX(IR,IC) + DELTA
ISN 0040      220  CONTINUE
ISN 0041       DELMAG = SQRT(DELMAG)
C
C     COMPUTE NEW IMAGE OBJECTIVE FUNCTION
ISN 0042       CALL CONV2D (HOLD,XFER,PIX,NYPWR2,NXPWR2,LOGNR,LOGNC,
     &    NY,NX,SINE,WORK,IWORK)
ISN 0043       CALL OBJFTN (HOLD,NYPWR2,PIX,DIRECT,NY,NX,DA,NMEAS,FLMGDA,
     &    SUAMP,OBJTMP,ENTRPY,RMSERR)
C
```

```
LEVEL 2.2  (SEPT 76)          ITERAT          OS/360  FORTRAN H EXTENDED          DATE 77.031/19.21.37

                C     ESTIMATE OPTIMAL OBJECTIVE FUNCTION VALUE IN PERTURBATION
                C     DIRECTION BY QUADRATIC APPROXIMATION
  ISN 0044            B=2.*(OBJTMP-OBJN-DELXG)
  ISN 0045            IF( B .EQ. 0. )  GO TO 240
  ISN 0047            EPSTAR = -DELXG/B
  ISN 0048            OBSTAR = OBJN + EPSTAR*DELXG + 0.5*B*EPSTAR**2
                C
                C        CALCULATE ANGLES BETWEEN GRADIENT AND MOVE AND THIS MOVE AND
                C        THE PREVIOUS MOVE.
  ISN 0049            ARG = DELXG/(DELMAG*GMAG)
  ISN 0050            GPHI = -1.
  ISN 0051            IF( ABS(ARG) .LE. 1. )  GPHI = ARCOS(ARG)*57.2958
  ISN 0053            ARG = DELXP/(DELMAG*PHEMAG)
  ISN 0054            DPHI = -1.
  ISN 0055            IF( ABS(ARG) .LE. 1. )  DPHI = ARCOS(ARG)*57.2958
  ISN 0057            WRITE(6,901) DMP,OBJTMP,OBSTAR,EPSTAR,GPHI
  ISN 0058            IF( B .GT. 0. )  GO TO 240
                C
                C        MODIFY DAMPING BY AT MOST CHDAMP.
  ISN 0060            DMP = AMAX1( 1.-DMP*(1.-CHDAMP),AMIN1( DMP*(1.+CHDAMP),
                     &   DMP/SQRT(EPSTAR) ) )
                C
                C     KEEP CURENT IMAGE IF IT GETS MORE THAN HALF WAY TO TOP
  ISN 0061            IF( OBSTAR-OBJTMP .GT. 0.5*(OBSTAR-OBJN) .AND. EPSTAR .LE. 1. )
                     &   GO TO 270
                C
                C     CURRENT IMAGE GOOD ENOUGH, COMPUTE CHANGE IN THIS ITERATION
  ISN 0063      240 OBJN=OBJTMP
  ISN 0064          DO 250 IC=1,NX
  ISN 0065          DO 250 IR=1,NY
  ISN 0066          PRMS=PRMS+PIX(IR,IC)**2
  ISN 0067      250 DXRMS = DXRMS + PMOVE(IR,IC)**2
  ISN 0068          PRMS=SQRT(PRMS/NPIXL)
  ISN 0069          DXRMS=SQRT(DXRMS/NPIXL)
  ISN 0070          R = DXRMS/PRMS
  ISN 0071          WRITE(6,902) DXRMS,PRMS,R,DPHI
  ISN 0072          GOTO 120
                C
                C     MOVE TO ESTIMATED OPTIMAL IMAGE
  ISN 0073      270 DO 280 IC=1,NX
  ISN 0074          DO 280 IR=1,NY
  ISN 0075          TEMP=EPSTAR*PMOVE(IR,IC)
  ISN 0076          PIX(IR,IC) = PIX(IR,IC) - PMOVE(IR,IC) + TEMP
  ISN 0077      275 DXRMS=DXRMS+TEMP**2
  ISN 0078      280 PRMS=PRMS+PIX(IR,IC)**2
  ISN 0079          PRMS=SQRT(PRMS/NPIXL)
  ISN 0080          DXRMS=SQRT(DXRMS/NPIXL)
  ISN 0081          R = DXRMS/PRMS
  ISN 0082          WRITE(6,903) DXRMS,PRMS,R,DPHI
  ISN 0083          GOTO 100
                C
                C
  ISN 0084      900 FORMAT('0',I,' =ITERATION',IPE12.4,' =OBJECT ',IPE12.4,' =ENTROPY',
                     &IPE12.4,' =RMSERR ',IPE12.4,' =RMSGRAD')
  ISN 0085      901 FORMAT(1X,'  ',F9.1,' =DAMPING   ',IPE12.4,' =OBJTMP',IPE12.4,' =OBJSTAR',
                     &'  ',OPF12.5,' =FRACSTR',OPF12.1,' =GRADPHI')
```

```
LEVEL 2.2  (SEPT 76)          ITERAT          OS/360  FORTRAN H EXTENDED          DATE 77.031/19.21.37

   ISN 0006      902    FORMAT(' KEEP THIS IMAGE:    ',1PE12.4,'=DXRMS  ',1PE12.4,
                       &'=PRMS   ',2PF12.2,'% CHANGE',0PF12.1,'=DELPHI')
   ISN 0007      903    FORMAT(' TAKE OPTIMAL MOVE:  ',1PE12.4,'=DXRMS  ',1PE12.4,
                       &'=PRMS   ',2PF12.2,'% CHANGE',0PF12.1,'=DELPHI')
   ISN 0008             END
*OPTIONS IN EFFECT*NAME(MAIN) NOOPTIMIZE LINECOUNT(58) SIZE(MAX) AUTODBL(NONE)
*OPTIONS IN EFFECT*SOURCE EBCDIC NOLIST NODECK OBJECT NOMAP NOFORMAT GOSTMT NOXREF ALC NOANSF TERM FLAG(I)
*STATISTICS*      SOURCE STATEMENTS =       87, PROGRAM SIZE =      4146, SUBPROGRAM NAME =ITERAT
*STATISTICS*  NO  DIAGNOSTICS GENERATED
******* END OF COMPILATION *******                                  110K BYTES OF CORE NOT USED
```

```
 LEVEL 2.2  (SEPT 76)                    OS/360  FORTRAN H EXTENDED          DATE 77.031/19.21.48

REQUESTED OPTIONS: NOMAP

OPTIONS IN EFFECT: NAME(MAIN) NOOPTIMIZE LINECOUNT(58) SIZE(MAX) AUTODBL(NONE)
                   SOURCE EBCDIC NOLIST NODECK OBJECT NOMAP NOFORMAT GOSTMT NOXREF ALC NOANSF TERM FLAG(I)

   ISN 0002              SUBROUTINE MULT2D (A,B,NROW,NCOL)
                     C
                     C     MULTIPLICATION OF COMPUTED ARRAYS A AND B WHICH ARE OUTPUT OF
                     C     FFT2R.  FOR ARRAY FORMATS, SEE COMMENTS IN FFT2R.  THIS SUB-
                     C     ROUTINE IS USEFUL FOR 2-D HIGH-SPEED FFT CONVOLUTION OF NROW BY
                     C     NCOL ARRAYS.
                     C     USE:  GIVEN TWO REAL ARRAYS, FFT BOTH WITH FFT2R.  THEN CALL
                     C     CMULT2D AND FOLLOW WITH A CALL TO IFFT2R WITH A AS INPUT ARRAY.
                     C     THE RESULT IS THE 2-D CIRCULAR CONVOLUTION OF THE ORIGINAL DATA
                     C     DIVIDED BY NROW*NCOL.
                     C
                     C                                                    SJW 5/1/76
                     C
   ISN 0003              DIMENSION A(NROW,NCOL),B(NROW,NCOL)
   ISN 0004              A(1,1)=A(1,1)*B(1,1)
   ISN 0005              A(2,1)=A(2,1)*B(2,1)
   ISN 0006              A(1,2)=A(1,2)*B(1,2)
   ISN 0007              A(2,2)=A(2,2)*B(2,2)
   ISN 0008              DO 10 I=3,NCOL,2
   ISN 0009              CALL CMUL (A(1,I),A(1,I+1),B(1,I),B(1,I+1))
   ISN 0010           10 CALL CMUL (A(2,I),A(2,I+1),B(2,I),B(2,I+1))
   ISN 0011              DO 20 J=1,NCOL
   ISN 0012              DO 20 I=3,NROW,2
   ISN 0013           20 CALL CMUL (A(I,J),A(I+1,J),B(I,J),B(I+1,J))
   ISN 0014              RETURN
   ISN 0015              END

*OPTIONS IN EFFECT*NAME(MAIN) NOOPTIMIZE LINECOUNT(58) SIZE(MAX) AUTODBL(NONE)

*OPTIONS IN EFFECT*SOURCE EBCDIC NOLIST NODECK OBJECT NOMAP NOFORMAT GOSTMT NOXREF ALC NOANSF TERM FLAG(I)

*STATISTICS*      SOURCE STATEMENTS =       14, PROGRAM SIZE =      1648, SUBPROGRAM NAME =MULT2D

*STATISTICS*  NO  DIAGNOSTICS GENERATED

****** END OF COMPILATION ******                                  130K BYTES OF CORE NOT USED
```

```
 LEVEL 2.2  (SEPT 76)                    OS/360  FORTRAN H EXTENDED          DATE 77.031/19.21.55

REQUESTED OPTIONS: NOMAP

OPTIONS IN EFFECT: NAME(MAIN) NOOPTIMIZE LINECOUNT(58) SIZE(MAX) AUTODBL(NONE)
                   SOURCE EBCDIC NOLIST NODECK OBJECT NOMAP NOFORMAT GOSTMT NOXREF ALC NOANSF TERM FLAG(I)

   ISN 0002             SUBROUTINE OBJFTN (HOLD,NRBIG,PIX,DIRECT,NY,NX,DA,NMEAS,FLMBDA,
                       &    SUAMP,OBJ,ENTRPY,RMSERR)
                     C
                     C      THIS SUBROUTINE CALCULATES THE OBJECTIVE FUNCTION VALUE.
                     C      ARGUMENTS:
                     C
                     C      HOLD: ARRAY CONTAINING CONVOLUTION RESULTS
                     C      NRBIG:  NUMBER OF ROWS IN HOLD (2*NY)
                     C      PIX:  IMAGE ARRAY
                     C      DIRECT:  DIRECT TRANSFORM ARRAY
                     C      NY:  NUMBER OF IMAGE ROWS
                     C      NX:  NUMBER OF IMAGE COLUMNS
                     C      DA:  PIXEL AREA
                     C      NMEAS:  NUMBER OF COMPLEX VISIBILITIES
                     C      FLMBDA:  USER-SELECTED PARAMETER LAMBDA
                     C      SUAMP:  SUM OF SQUARED VISIBILITY MAGNITUDES EACH NORMALIZED
                     C              BY CORRESPONDING ERROR VARIANCE
                     C      OBJ:  OBJECTIVE FUNCTION VALUE RETURNED HERE
                     C      ENTRPY:  ENTROPY VALUE RETURNED HERE
                     C      RMSERR:  IMAGE-MEASUREMENT DISCREPANCY (RMS VALUE OF RESIDUAL
                     C               MAGNITUDE NORMALIZED BY CORRESPONDING ERROR STANDARD
                     C               DEVIATION)
                     C
                     C                                                    SJH 12/13/76
                     C
   ISN 0003             DIMENSION HOLD(NRBIG,1),PIX(NY,1),DIRECT(NY,1)
   ISN 0004             DA2=DA/2.
   ISN 0005             ENTRPY=0.
   ISN 0006             OBJ=-FLMBDA*SUAMP/DA
   ISN 0007             DO 250 IC=1,NX
   ISN 0008             DO 250 IR=1,NY
   ISN 0009             TEMP=ALOG(PIX(IR,IC))
   ISN 0010             ENTRPY=ENTRPY+TEMP
   ISN 0011         250 OBJ=OBJ+TEMP+(DIRECT(IR,IC)-DA2*HOLD(IR,IC))*PIX(IR,IC)
   ISN 0012             OBJ=DA*OBJ
   ISN 0013             ENTRPY=DA*ENTRPY
   ISN 0014             RMSERR=SQRT((ENTRPY-OBJ)/(FLMBDA*NMEAS))
   ISN 0015             RETURN
   ISN 0016             END

*OPTIONS IN EFFECT*NAME(MAIN) NOOPTIMIZE LINECOUNT(58) SIZE(MAX) AUTODBL(NONE)

*OPTIONS IN EFFECT*SOURCE EBCDIC NOLIST NODECK OBJECT NOMAP NOFORMAT GOSTMT NOXREF ALC NOANSF TERM FLAG(I)

*STATISTICS*      SOURCE STATEMENTS =       15, PROGRAM SIZE =      1010, SUBPROGRAM NAME =OBJFTN

*STATISTICS*  NO  DIAGNOSTICS GENERATED

*****  END OF COMPILATION  *****                              130K BYTES OF CORE NOT USED
```

```
LEVEL 2.2  (SEPT 76)          OS/360  FORTRAN H EXTENDED          DATE 77.031/19.22.20

REQUESTED OPTIONS: NOMAP

OPTIONS IN EFFECT: NAME(MAIN) NOOPTIMIZE LINECOUNT(58) SIZE(MAX) AUTODBL(NONE)
                   SOURCE EBCDIC NOLIST NODECK OBJECT NOMAP NOFORMAT GOSTMT NOXREF ALC NOANSF TERM FLAG(I)

ISN 0002            SUBROUTINE PSF( P,NROW,DELY,NCOL,DELX,U,V,WT,NMEAS,XC,XS,YC,YS )
ISN 0003            DIMENSION P(1),U(1),V(1),WT(1),XC(1),XS(1),YC(1),YS(1)
ISN 0004            REAL*8 CARG,DCOS,DSIN,COS1,SIN1,COS2,SIN2
                    C
                    C
                    C     THIS SUBROUTINE CALCULATES THE POINT RESPONSE OF A DIRECT FOURIER
                    C     RECONSTRUCTION FROM VISIBILITIES WITH NORMALIZATION TO SUIT ME
                    C     RECONSTRUCTION.  THE FUNCTION EVALUATED IS
                    C
                    C     P(L) = 2*SUM( I=1,NMEAS) WT(I)*COS(2*PI*(U(I)*CL*DELX
                    C                                +V(I)*RL*DELY))
                    C
                    C     WHERE ONE-DIMENSIONAL NOTATION IS USED.  CL AND RL ARE COLUMN
                    C     AND ROW INDICATORS FOR THE LTH ELEMENT OF P.  ONLY THE RIGHT
                    C     HALF OF THE POINT RESPONSE IS STORED SINCE THE FUNCTION HAS
                    C     SYMMETRY ABOUT THE ORIGIN.  THE VALUES CL AND RL RANGE FROM
                    C     0<=CL<=NCOL-1 AND NROW-1>=RL>=1-NROW.  ROW-SEQUENTIAL FORMAT
                    C     IS USED WITH P(1) REPRESENTING CL=0, RL=NROW-1.  OTHER PARAMETER
                    C     INFORMATION:
                    C
                    C     P:  POINT RESPONSE ARRAY WHICH MUST CONTAIN AT LEAST
                    C         (2*NROW-1)*NCOL ELEMENTS.
                    C     NROW:  NUMBER OF ROWS IN THE IMAGE.
                    C     DELY:  DISTANCE BETWEEN ADJACENT ROWS IN THE IMAGE.
                    C     NCOL:  NUMBER OF COLUMNS IN THE IMAGE.
                    C     DELX:  DISTANCE BETWEEN ADJACENT COLUMNS IN THE IMAGE.
                    C     U,V:  ARRAYS WITH MEASUREMENT LOCATIONS IN THE SPATIAL FREQUENCY
                    C         DOMAIN.
                    C     WT:  ARRAY OF RECONSTRUCTION ERROR WEIGHTS (LAMBDA/VARIANCE).
                    C         IF WT(1) < 0, ALL WEIGHTS ARE TAKEN TO BE ABS(WT(1)).
                    C     NMEAS:  NUMBER OF VISIBILITY MEASUREMENTS
                    C     XC,XS,YC,YS:  WORK ARRAYS.  XC AND XS EACH NEED AT LEAST NCOL-1
                    C         ELEMENTS.  YC AND YS REQUIRE AT LEAST NROW-1 ELEMENTS.
                    C
                    C                                                    SJ 5/1/76
                    C
                    C     MODIFIED OCTOBER 1976 BY CJ GREBENKEMPER FOR FASTER EXECUTION
                    C     BY USING TRIG IDENTITIES IN THE DIRECT FOURIER TRANSFORM.
                    C
ISN 0005            NCOLND=NCOL-1
ISN 0006            NROWND=NROW-1
ISN 0007            NEL=(2*NROW-1)*NCOL
                    C
ISN 0008            DO 10 I=1,NEL
ISN 0009          10 P(I)=0.
                    C
                    C     ACCUMULATE POINT RESPONSE. MEASUREMENT LOOP FIRST
ISN 0010            DO 60 K=1,NMEAS
                    C
ISN 0011            A = -WT(1)
ISN 0012            IF( A .EQ. 0. )  A = 1.
```

```
ISN 0014            IF( W .LT. 0. )  W = WT(K)
ISN 0016            W = 2.*W
                C
ISN 0017            UKDX=6.2831853*U(K)*DELX
ISN 0018            VKDY=6.2831853*V(K)*DELY
                C
                C   USE STORED TRIG FUNCTIONS FOR SPEED
ISN 0019            CCS1 = 1.
ISN 0020            SIN1 = 0.
ISN 0021            DARG = UKDX
ISN 0022            COS2 = DCOS(DARG)
ISN 0023            SIN2 = DSIN(DARG)
ISN 0024            DO 20 L=1,NCOLND
ISN 0025            DARG = COS1*CCS2-SIN1*SIN2
ISN 0026            SIN1 = COS1*SIN2+SIN1*COS2
ISN 0027            CCS1 = DARG
ISN 0028            XC(L) = CCS1
ISN 0029            XS(L) = SIN1
ISN 0030        20  CONTINUE
                C
ISN 0031            CCS1 = 1.
ISN 0032            SIN1 = 0.
ISN 0033            DARG = VKDY
ISN 0034            CCS2 = DCOS(DARG)
ISN 0035            SIN2 = DSIN(DARG)
ISN 0036            DO 25 L=1,NROWND
ISN 0037            DARG = COS1*CCS2-SIN1*SIN2
ISN 0038            SIN1 = COS1*SIN2+SIN1*COS2
ISN 0039            CCS1 = DARG
ISN 0040            YC(L) = CCS1
ISN 0041            YS(L) = SIN1
ISN 0042        25  CONTINUE
                C
                C   ROW LOOP
                C
ISN 0043            KDYEND=2*NROWND+1
ISN 0044            DO 50 KDY=1,KDYEND
ISN 0045            KDY=NROW-KDYU
ISN 0046            I=(NROW-1-KDY)*NCOL
ISN 0047            CKDY=YC(IABS(KDY))
ISN 0048            IF (KDY .EQ. 0) CKDY=1.
ISN 0050            SKDY=YS(IABS(KDY))
ISN 0051            IF (KDY .EQ. 0) SKDY=0.
ISN 0053            IF (KDY .LT. 0) SKDY=-SKDY
                C
                C   COLUMN LOOP
ISN 0055            P(I+1)=P(I+1)*CKDY**2
ISN 0056            DO 40 KDX=1,NCOLND
ISN 0057            CKDX=XC(KDX)
ISN 0058            SKDX=XS(KDX)
ISN 0059            II=I+1+KDX
ISN 0060            P(II) = P(II) + W*( CKDX*CKDY-SKDX*SKDY )
ISN 0061        40  CONTINUE
ISN 0062        50  CONTINUE
ISN 0063        60  CONTINUE
ISN 0064            RETURN
```

```
LEVEL 2.2 (SEPT 76)          PSF          OS/360 FORTRAN H EXTENDED          DATE 77.031/19.22.20
   ISN 0065            END
*OPTIONS IN EFFECT*NAME(MAIN) NOOPTIMIZE LINECOUNT(58) SIZE(MAX) AUTODBL(NONE)
*OPTIONS IN EFFECT*SOURCE EBCDIC NOLIST NODECK OBJECT NOMAP NOFORMAT GOSTMT NOXREF ALC NOANSF TERM FLAG(I)
*STATISTICS*      SOURCE STATEMENTS =       64, PROGRAM SIZE =      1842, SUBPROGRAM NAME =   PSF
*STATISTICS*  NO  DIAGNOSTICS GENERATED
*****  END OF COMPILATION  *****                                  126K BYTES OF CORE NOT USED
*STATISTICS*  NO  DIAGNOSTICS THIS STEP
```

```
 LEVEL 2.2  (SEPT 76)                    OS/360  FORTRAN H EXTENDED          DATE 77.031/19.22.03

REQUESTED OPTIONS: NOMAP

OPTIONS IN EFFECT: NAME(MAIN) NOOPTIMIZE LINECOUNT(58) SIZE(MAX) AUTODBL(NONE)
                   SOURCE EBCDIC NOLIST NODECK OBJECT NOMAP NOFORMAT GOSTMT NOXREF ALC NOANSF TERM FLAG(I)

   ISN 0002              FUNCTION ROOT (D,CONV,PIX,DA,PLL,RTWOA,FOURAC)
                     C   THIS FUNCTION SOLVES THE DERIVATIVE EQUATION TO COMPUTE THE
                     C   OPTIMAL UNIVARIATE CHANGE TO THE BRIGHTNESS IN A PARTICULAR
                     C   PIXEL.
                     C
                     C   ARGUMENTS:
                     C
                     C   D:  DIRECT TRANSFORM VALUE FOR THE PIXEL
                     C   CONV:  CONVOLUTION VALUE OF THE PIXEL
                     C   PIX:  BRIGHTNESS OF THE PIXEL
                     C   DA:  PIXEL AREA
                     C   PLL:  CENTRAL VALUE OF POINT RESPONSE ARRAY
                     C   RTWOA:  .5/(DA*PLL)
                     C   FOURAC:  2./RTWOA
                     C
                     C                                              SJW 5/26/76
                     C
   ISN 0003              ROOT=D-DA*(CONV-PLL*PIX)
   ISN 0004              ROOT=RTWOA*(ROOT+SQRT(ROOT**2+FOURAC))
   ISN 0005              RETURN
   ISN 0006              END

*OPTIONS IN EFFECT*NAME(MAIN) NOOPTIMIZE LINECOUNT(58) SIZE(MAX) AUTODBL(NONE)

*OPTIONS IN EFFECT*SOURCE EBCDIC NOLIST NODECK OBJECT NOMAP NOFORMAT GOSTMT NOXREF ALC NOANSF TERM FLAG(I)

*STATISTICS*      SOURCE STATEMENTS =        6, PROGRAM SIZE =      374, SUBPROGRAM NAME =  ROOT

*STATISTICS*  NO  DIAGNOSTICS GENERATED

****** END OF COMPILATION ******                                  130K BYTES OF CORE NOT USED
```

```
LEVEL 2.2  (SEPT 76)                OS/360  FORTRAN H EXTENDED          DATE 77.031/19.22.10

REQUESTED OPTIONS: NOMAP

OPTIONS IN EFFECT: NAME(MAIN) NOOPTIMIZE LINECOUNT(58) SIZE(MAX) AUTODBL(NONE)
                   SOURCE EBCDIC NOLIST NODECK OBJECT NOMAP NOFORMAT GOSTMT NOXREF ALC NOANSF TERM FLAG(I)

   ISN 0002              SUBROUTINE XFERFN (VECTOR,POINT,LOGNR,LOGNC,NROW2,NCOL2,IWORK)
                     C   GET POINT RESPONSE OUT OF VECTOR FORMAT AND INTO CORNERS
                     C   OF ARRAY THAT WILL BE TRANSFORMED TO HOLD TRANSFER FUNCTION.
                     C   VECTOR CONTAINS PACKED POINT RESPONSE TERMS IN ROW MAJOR FORM.
                     C   POINT(NROW2,NCOL2) IS 2-D ARRAY THAT GETS CORNER VALUES.  POINT
                     C   IS THEN TRANSFORMED AND SCALED TO PROVIDE DISCRETE TRANSFER
                     C   FUNCTION ARRAY.
                     C
                     C   ARGUMENTS
                     C
                     C   VECTOR:  ARRAY OF POINT RESPONSE VALUES CALCULATED BY PSF
                     C   POINT:   ARRAY CONTAINING TRANSFER FUNCTION VALUES ON RETURN
                     C   LOGNR:   1+LOG(BASE 2) (NUMBER OF IMAGE ROWS)
                     C   LOGNC:   1+LOG(BASE 2) (NUMBER OF IMAGE COLUMNS)
                     C   NROW2:   TWO TIMES THE NUMBER OF IMAGE ROWS
                     C   NCOL2:   TWO TIMES THE NUMBER OF IMAGE COLUMNS
                     C   IWORK:   INTEGER WORK ARRAY FOR THE FFT SUBROUTINE
                     C
                     C                                                SJW 9/26/76
                     C
   ISN 0003              DIMENSION POINT(NROW2,NCOL2),VECTOR(1),IWORK(1)
   ISN 0004              NROW=NROW2/2
   ISN 0005              NCOL=NCOL2/2
   ISN 0006              DO 5 IC=1,NCOL2
   ISN 0007              DO 5 IR=1,NROW2
   ISN 0008            5 POINT(IR,IC)=0.
   ISN 0009              NRWMID=NROW2-NROW+2
   ISN 0010              NCLMID=NCOL2-NCOL+2
   ISN 0011              IST=0
                     C   LOWER LEFT CORNER
   ISN 0012              DO 10 IR=NRWMID,NROW2
   ISN 0013              DO 10 IC=1,NCOL
   ISN 0014              IST=IST+1
   ISN 0015           10 POINT(IR,IC)=VECTOR(IST)
                     C   UPPER LEFT CORNER
   ISN 0016              DO 20 IR=1,NROW
   ISN 0017              DO 20 IC=1,NCOL
   ISN 0018              IST=IST+1
   ISN 0019           20 POINT(IR,IC)=VECTOR(IST)
                     C   LOWER RIGHT CORNER
   ISN 0020              IST=NCOL*(2*NROW-1)+1
   ISN 0021              DO 30 IR=NRWMID,NROW2
   ISN 0022              DO 29 IC=NCLMID,NCOL2
   ISN 0023              IST=IST-1
   ISN 0024           29 POINT(IR,IC)=VECTOR(IST)
   ISN 0025           30 IST=IST-1
                     C   UPPER RIGHT CORNER
   ISN 0026              DO 40 IR=1,NROW
   ISN 0027              DO 39 IC=NCLMID,NCOL2
   ISN 0028              IST=IST-1
   ISN 0029           39 POINT(IR,IC)=VECTOR(IST)
```

```
LEVEL 2.2  (SEPT 76)        XFERFN          OS/360  FORTRAN H EXTENDED          DATE 77.031/19.22.10

   ISN 0030          40 IST=IST-1
                     C
                     C     2-D FFT OF POINT SPREAD FUNCTION
   ISN 0031                CALL FFT2R (PCINT,NRO*2,NCOL2,LOGNR,LOGNC,VECTOR,VECTOR(65),
                          &    IWORK)
                     C     SCALE TRANSFER FUNCTION SO 2-D CONVOLUTION IS CORRECTLY NORMALIZED
   ISN 0032                SCALE=NRO**2*NCOL2
   ISN 0033                DO 50 I=1,NCOL2
   ISN 0034                DO 50 J=1,NRO*2
   ISN 0035              50 POINT(J,I)=SCALE*POINT(J,I)
   ISN 0036                RETURN
   ISN 0037                END

*OPTIONS IN EFFECT*NAME(MAIN) NOOPTIMIZE LINECOUNT(58) SIZE(MAX) AUTODBL(NONE)

*OPTIONS IN EFFECT*SOURCE EBCDIC NOLIST NODECK OBJECT NOMAP NOFORMAT DOSTMT NOXREF ALC NOANSF TERM FLAG(I)

*STATISTICS*      SOURCE STATEMENTS =       36, PROGRAM SIZE =      1580, SUBPROGRAM NAME =XFERFN

*STATISTICS*  NO  DIAGNOSTICS GENERATED

*****  END OF COMPILATION  *****                                  130K BYTES OF CORE NOT USED
```