NATIONAL RADIO ASTRONOMY OBSERVATORY
Charlottesville, Virginia

VLA SCIENTIFIC MEMORANDUM #131

The Effects of Various Convolving Functions on Aliasing and Relative Signal-to-Noise Ratios

Eric W. Greisen
December 1979

I. Introduction

VLA maps are normally produced by Fourier transformation only after the observed data samples have been smoothed and then resampled on a regular, rectangular grid. This process causes noise, sidelobes, and real source emission which lie outside the field of view of the map to be reflected ("aliased") into the map. Because it is difficult to produce, at the full VLA resolution, maps which cover the full single-dish field of view, errors caused by the aliasing of interfering sources are likely to be a significant consideration. However, the effect of aliasing can be reduced by a proper choice of the smoothing function. Several of these functions have been investigated recently in Charlottesville.

This memorandum extends the results of a series of memoranda on the effects of convolution in the u-v plane. In Memorandum No. 123 (1976), Greisen developed the basic mathematics and offered some graphical displays which showed the effects of convolution on the apparent brightness temperatures and noise. Clark (Memorandum No. 124, 1976) pointed out the serious effect that the aliasing of the noise has on the signal-to-noise ratios of maps produced by the Fast Fourier transform. In Memorandum No. 129, Schwab presented a number of convolving functions and tabulated their Fourier transforms. In this memorandum, I will list the convolving functions currently available in the Charlottesville mapping programs, derive the formula for noise mentioned by Clark, and discuss graphs of the properties of a variety of the functions.

II. The Functions and Their Implementation

We will use a one-dimensional analysis throughout since we assume that the convolving function is real, symmetric, and separable, i.e., $C(u,v,w) = Cu(u)\ Cv(v)\ Cw(w)$. The actual convolving function used in a mapping program is not a simple, smooth function (e.g., a gaussian) for two practical reasons. First, in order to keep CPU time finite, the actual convolving function must have non-zero value only within a box of small dimension. This is equivalent to multiplying a smooth convolving function, C, by a pill box function, B, of finite radius, CSIZE. Second, since the cost of evaluating the convolving function a significant number of times is also prohibitive, any function evaluation must be done by table lookup. This process is carried out by computing, in advance, the values of C at points separated by f along the axis. For any given data sample, the value of C is taken to be that at the nearest of these pre-computed points.

To express these concepts mathematically, we may write the actual convolving function, Ca, as

$$Ca = P * (II \cdot B \cdot C)$$

where * represents convolution, C is any convolving function, and

$$B(u) = 1 / (2 \cdot CSIZE) \qquad |u| < CSIZE$$
$$B(u) = 0 \qquad |u| > CSIZE$$

$$II = \sum_m \delta(u-mf)$$

$$P(u) = 1 / f \qquad |u| < f/2$$
$$P(u) = 0 \qquad |u| > f/2 .$$

(For convenience, we express the axis coordinate, u, and all parameters in units of cells. In these units, the inverse coordinate, x, has value 0.5 at the edge of the map.)

The map produced by the Fourier transform of the smoothed observations is the product of the "desired" map and the Fourier transform of Ca. The latter is given by

$$\overline{Ca} = \bar{P} \cdot (\overline{III} * \bar{B} * \bar{C})$$

where the bar represents the Fourier transform and

$$\bar{P}(x) = \sin(\pi f x)/(\pi f x)$$

$$\overline{III}(x) = \sum_m \delta(x-m/f)$$

Thus, the effect of the table lookup is to cause $\overline{Ca}$ to repeat at intervals of 1/f weighted down slightly by $\bar{P}$. If f is reasonably small (e.g., 0.01) and $\bar{B} * \bar{C}$ has negligible value for large $|x|$ (e.g., $|x| > 1/2f$), then the fact that table lookup was required may be ignored. It is worth noting that $\overline{Ca}$ may be computed exactly in a reasonable amount of CPU time using the formula

$$\overline{Ca}(x) = \frac{\sin(\pi f x)}{(\pi f x)} \sum_{j=\frac{-CSIZE}{f}}^{CSIZE/f} C(jf) \cos(2\pi x j f)$$

Thus, it is not necessary to select functions C which have analytic expressions for $\bar{C} * \bar{B}$.

The particular functions, C, available at present in the Charlottesville mapping programs are as follows:

(0) Pill box:

$$C(u) = 1.0 \qquad |u| < \text{CPAR1}$$
$$C(u) = 0.5 \qquad |u| = \text{CPAR1}$$
$$C(u) = 0.0 \qquad |u| > \text{CPAR1}$$

defaults: CSIZE = 0.50
CPAR1 = CSIZE

(1) Exponential:

C(u) = exp(-(|u|/CPAR1) ** CPAR2)

defaults: CSIZE = 3.0
CPAR1 = 1.0
CPAR2 = 2.0

(2) Sinc:

$$C(u) = \sin(\pi u/\text{CPAR1}) / (\pi u/\text{CPAR1})$$

defaults: CSIZE = 3.0
CPAR1 = 1.0

(3) Exponential times sinc:

C(u) = sin(πu/CPAR1) / (πu/CPAR1) · exp(-(|u|/CPAR2) ** CPAR3)

defaults: CSIZE = 3.0
CPAR1 = 1.0
CPAR2 = 2.0
CPAR3 = 3.0

(4,5) Kaiser-Bessel:

$$C(u) = A + B\cos(\pi u/\text{CPAR1}) + C\cos(2\pi u/\text{CPAR1})$$

where, if,
$$x1 = \pi \cdot \text{CPAR2}$$
$$x2 = \pi \cdot \text{SQRT}(\text{CPAR2} \cdot \text{CPAR2} - 1)$$
$$x3 = \pi \cdot \text{SQRT}(\text{CPAR2} \cdot \text{CPAR2} - 4)$$

then, for "half Kaiser-Bessel"

$$A = \sinh(x1)$$
$$B = 2 \cdot \sinh(x2)$$
$$C = 2 \cdot \sinh(x3)$$

and. for "full Kaiser-Bessel"

$$A = \sinh(x1) / x1$$
$$B = 2 \sinh(x2) / x2$$
$$C = 2 \sinh(x3) / x3$$

defaults: CSIZE = 2.0
CPAR1 = CSIZE
CPAR2 = 2.5 for half Kaiser-Bessel
CPAR2 = 2.2 for full Kaiser-Bessel

These functions were provided by Ron Harten of the WSRT and are used extensively by the Dutch. Harten has also tested the prolate spheroidal functions. His results suggest these functions are no better than the Kaiser-Bessel function and, hence, are not worth the extra trouble required to evaluate them.

(6,7) Exponential times Kaiser-Bessel:

$$C(u) = (A + B\cos(\pi u/\mathrm{CPAR1}) + C\cos(2\pi u/\mathrm{CPAR1})) \cdot \exp(-(|u|/\mathrm{CPAR3}) ** \mathrm{CPAR4})$$

where A, B, C and the defaults are the same as the Kaiser-Bessel functions and extra defaults: CPAR3 = 1.0
CPAR4 = 2.0

III. Noise

We will attempt to derive an approximate formula for the relative noise on maps produced via the FFT. To reduce the number of integral signs, we will write the equations in only one dimension. The convolution, resampling, and

Fourier transform operation may be written as

$$T \propto \int du\ e^{-2\pi iux} \sum_m \delta(u-m) \int du'\ C(u'-u) \sum_j V_j\ \delta(u'-u_j)\ W_j$$

where the j summation is over the discrete set of observations. Using

$$\sigma^2 = <|\sum_j \frac{\partial T}{\partial V_j}\ \Delta V_j|^2>$$

we obtain

$$\sigma^2 \propto \sum_m \sum_n \sum_j \iint du\ dv\ e^{-2\pi ix(u-v)}\ \delta(u-m)\ \delta(v-n)\ <\ |\Delta V_j|^2\ >\ W_j{}^2$$

$$\cdot \iint du'\ dv'\ \delta(u'-u_j)\ \delta(v'-u_j)\ C(u'-u)\ C(v'-v)\ ,$$

where we have assumed that $< V_j V_k{}^* > = 0$ where $j \neq k$. This assumption is valid only when long term calibration errors do not occur and when the noise in the real and imaginary parts of V is uncorrelated. Performing the integrals on u' and v' and substituting $z = v-u$, we obtain

$$\sigma^2 \propto \int dz\ e^{2\pi izx} \sum_m \sum_n \sum_j \int du\ \delta(u-m)\ \delta(u+z-n)$$

$$<\ |\Delta V_j|^2\ >\ W_j{}^2\ C(u_j-u)\ C(u_j-u-z)\ .$$

Substituting $k = n-m$ and performing the u integral, we obtain

$$\sigma^2 \propto \int dz\ e^{2\pi izx} \sum_k \delta(z-k) \sum_m R(m,z)$$

where

$$R(m,z) = \int du\ <\ |\Delta V(u)|^2\ >\ P^2(u)\ W^2(u)\ C(u-m)\ C(u-m-z),$$

with the sum over j replaced by an integral with a probability function P(u). Let us now assume that the weight function is

$$W_j{}^2 = (\Delta V_o)^2\ /\ (\ <|\Delta V_j|^2>\ <P^2(u)>_j\ )$$

which compensates in the usual way for any variation in noise (e.g., integration time, system temperature) and "averaged" local density of data points. The sum over m may be evaluated by noting, first, that C(u) has value only over a restricted range of u and that all $|u_j|$ are less than some Umax. Thus, $R(m,z) = 0$ for $|m| >$ Umax + CSIZE. If we assume that the data samples are uniformly distributed within the remaining u-space, i.e., that

$$<P^2(u)>_u = P^2(u) \quad ,$$

then R(m,z) is independent of m and the sum becomes simply a measure of the size of the sampled u-space.

Therefore,

$$\sigma^2 \propto \sum_k \int dz\ e^{2\pi izx}\ \delta(z-k) \int du\ C(u)\ C(u-z).$$

or, in words, the variance in the map plane is the Fourier transform of the sampled autocorrelation of the convolving function. Please note that this applies before any correction for the Fourier transform of the convolving function. This is, of course, the result stated by Clark in Memorandum No. 124, but I feel that the derivation is useful in revealing the nature of the assumptions. With the actual convolving functions listed in the previous section, the autocorrelation is easily computed at grid points if 1/f is an integer.

IV. The Plots

In order to evaluate the performance of the many convolving functions, we need to examine two functions both of which are shown in each of the accompanying plots. For x between 0 and 1 map radius, the relative signal-to-noise ratio given by

$$\overline{Ca}(x)\ /\ [\sum_k \int du\ e^{-2\pi iux}\ \delta(u-k) \int dz\ Ca(z)\ Ca(z-u)]^{\frac{1}{2}}$$

is plotted on a linear scale. For x between 1.01 and 10 map radii, a logarithmic scale is used to show the relative amplitude of the aliasing given by

$$\left| \overline{Ca}(x) \;/\; \overline{Ca}(x'(x)) \right|$$

where x'(x) is the position to which brightness at x is aliased. These are the two functions which are of interest when the maps have been corrected for the Fourier transform of the convolving function.

V. Discussion

The pill-box function has two strong virtues. It requires only one row of the u-v plane to be in core at a time and it uses the least CPU time in the convolution. However, Figure 1 reveals that the user pays a heavy price for this computational convenience. Sidelobes within 0.8 map radii of the edge of the map are aliased into the map with weights greater than 10%. Interfering sources are aliased with weights > 3% even at distances exceeding 10 map radii. Because of the large degree of aliasing, the relative signal-to-noise ratio departs significantly from 1.0 even fairly close to the map center. The reader is reminded that, for two-dimensional maps, the actual signal-to-noise ratio is the product of the function in x and the function in y. Thus, the relative signal-to-noise drops to 0.4 at the corners of the map.

The default exponential function, plotted in Figure 2a, requires 7 rows of the u-v plane to be in core at a time and does 36-49 table lookups per point. However, aliasing is less than 1% for all points more than 0.45 map radii outside the map and the signal-to-noise ratio is 1.0 out to 0.75 map radii. Figures 2b-2d illustrate the effects of different parameters. Reducing the support size alone (2b) causes ringing from the sharp edge introduced in Ca. Keeping the ratio of CSIZE and CPAR1 fixed, while varying CSIZE, causes the aliasing region to broaden (2c) or to become so narrow (2d) that unfortunate effects appear within the principal map area. Figure 2d illustrates the fact that seemingly reasonable sets of parameters can produce unhappy consequences. For this reason, I am making a modified version of this plotting program available within the Charlottesville program package.

The ideal convolving function would be the sinc function for which $\bar{C}$ is the pill box just encompassing the map. However, this function is obtained only as CSIZE and CPU time become infinite. The plots for finite CSIZE (Figures 3) show a comparatively sharp initial cutoff at the edge of the map, but high "sidelobes" whose amplitude decreases only slowly with distance. Multiplying the sinc by an exponential (Figures 4) causes the initial cutoff to become wider but can reduce the sidelobes substantially.

The functions provided by Ron Harten, called here the half and full Kaiser-Bessel functions, are designed to be used with CSIZE = 2.0. As illustrated in Figures 5b and 5c, the consequences of altering the default parameters are not necessarily desirable. These functions do have the remarkable, for such a small CSIZE, property of dropping abruptly outside the map and maintaining fairly low sidelobe levels thereafter. In particular, the half Kaiser-Bessel (Figure 5a) has very low sidelobes within the first radius outside the map. Multiplying the Kaiser-Bessel functions by a gaussian widens the inner cutoff which hurts the signal-to-noise ratio but eliminates the sidelobes altogether.

VI. Conclusions

The pill box function provides the fastest execution in the least core.

However, it should not be used when sidelobes of the source are large nor when significant emission outside the map area is probable. Thus, the pill-box funtion should not be used with most sensitive 6-cm and all 2-cm observations. When the full VLA is used with some tapering, sidelobes should not be a problem. In this case, the default gaussian with CSIZE = 3.0 is very effective in suppressing the aliasing of distant confusing sources. If 36-49 table lookups per point are too expensive, then the gaussian times the full Kaiser-Bessel is a good compromise and requires only 16-25 lookups per point. For suppression of nearby confusing sources and sidelobes, the half Kaiser-Bessel function is quite good. This function should be considered for mapping in the snapshot mode with partial arrays, for mapping with high-resolution (inverse taper), and for mosaicing of large sources which extend outside the map area.

Thus, the proper choice of convolving function can greatly reduce the effects of the aliasing of noise, sidelobes, and interfering sources. The reader is reminded, however, that no convolving function can eliminate interfering sources completely. Sidelobes of the interfering source which fall within the map area are not affected by the choice of convolving function.

FIGURE 1

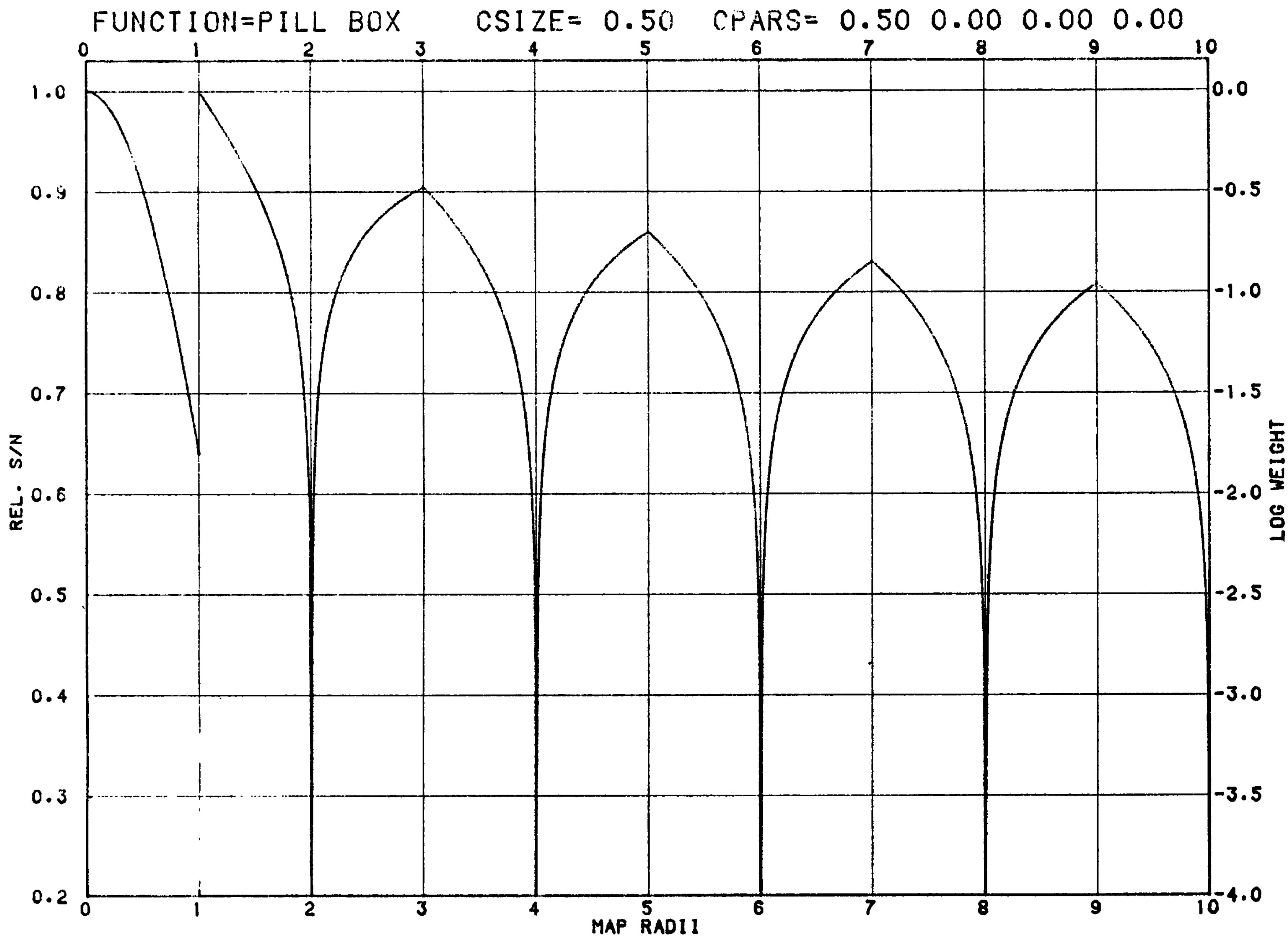

FUNCTION=PILL BOX CSIZE= 0.50 CPARS= 0.50 0.00 0.00 0.00
0 1 2 3 4 5 6 7 8 9 10
1.0 0.9 0.8 0.7 0.6 0.5 0.4 0.3 0.2
REL. S/N
0.0 -0.5 -1.0 -1.5 -2.0 -2.5 -3.0 -3.5 -4.0
LOG WEIGHT
MAP RADII

FIGURE 2a

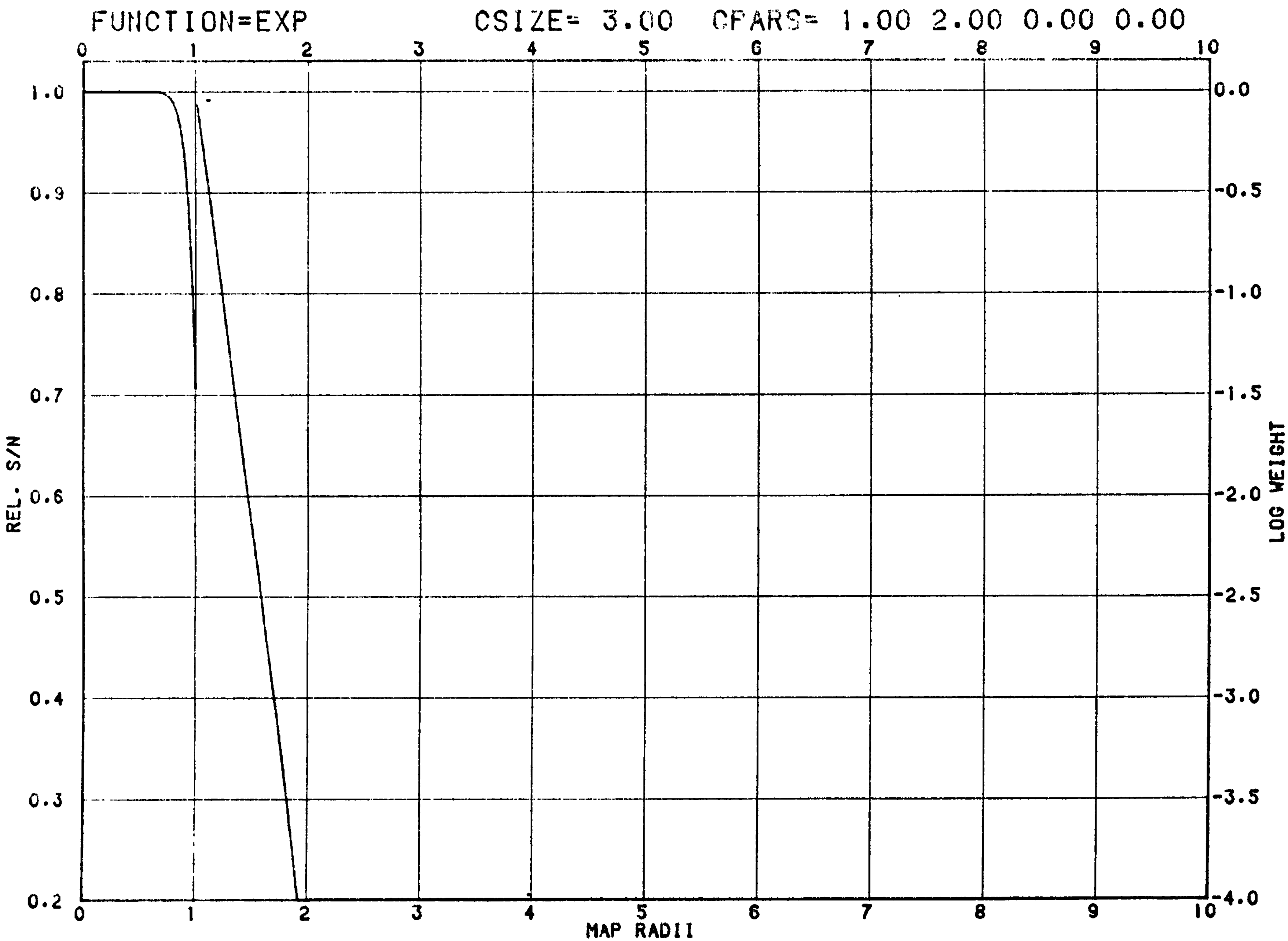

FUNCTION=EXP
CSIZE= 3.00
CPARS= 1.00 2.00 0.00 0.00
0
1
2
3
4
5
6
7
8
9
10
1.0
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
REL. S/N
0.0
-0.5
-1.0
-1.5
-2.0
-2.5
-3.0
-3.5
-4.0
LOG WEIGHT
MAP RADII

FIGURE 2b

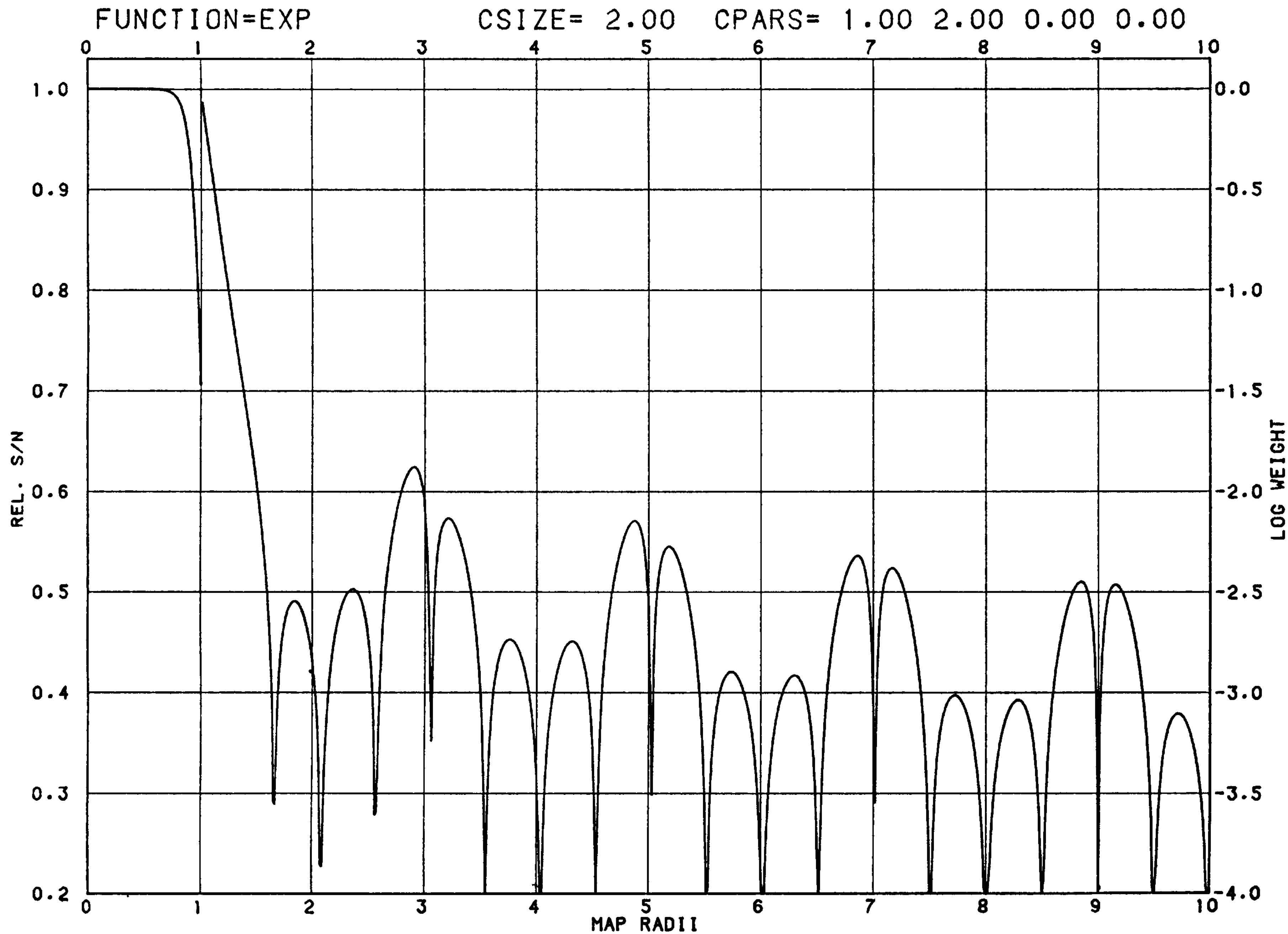

FUNCTION=EXP
CSIZE= 2.00
CPARS= 1.00 2.00 0.00 0.00
0 1 2 3 4 5 6 7 8 9 10
1.0 0.9 0.8 0.7 0.6 0.5 0.4 0.3 0.2
REL. S/N
0.0 -0.5 -1.0 -1.5 -2.0 -2.5 -3.0 -3.5 -4.0
LOG WEIGHT
MAP RADII

FIGURE 2c

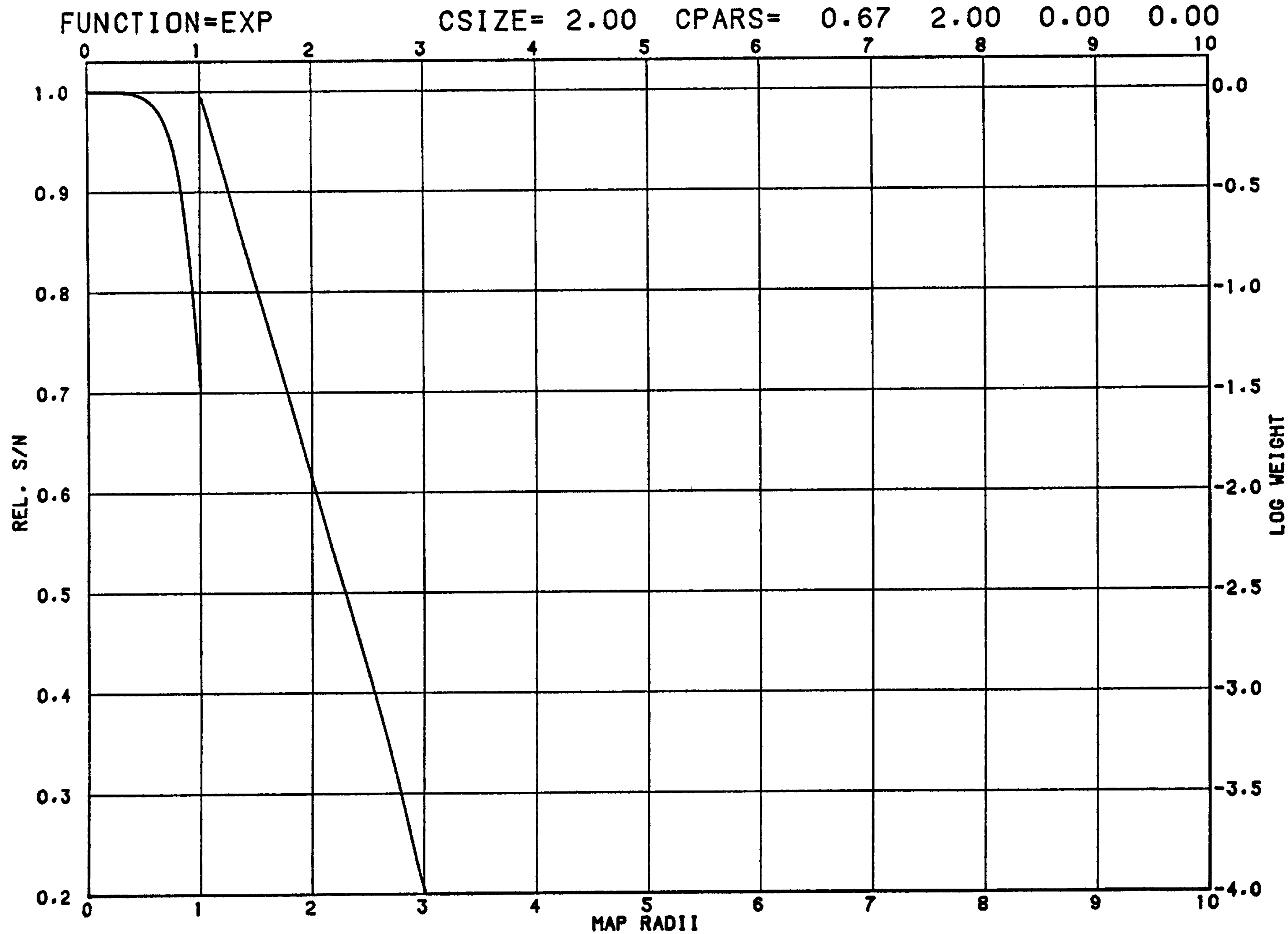
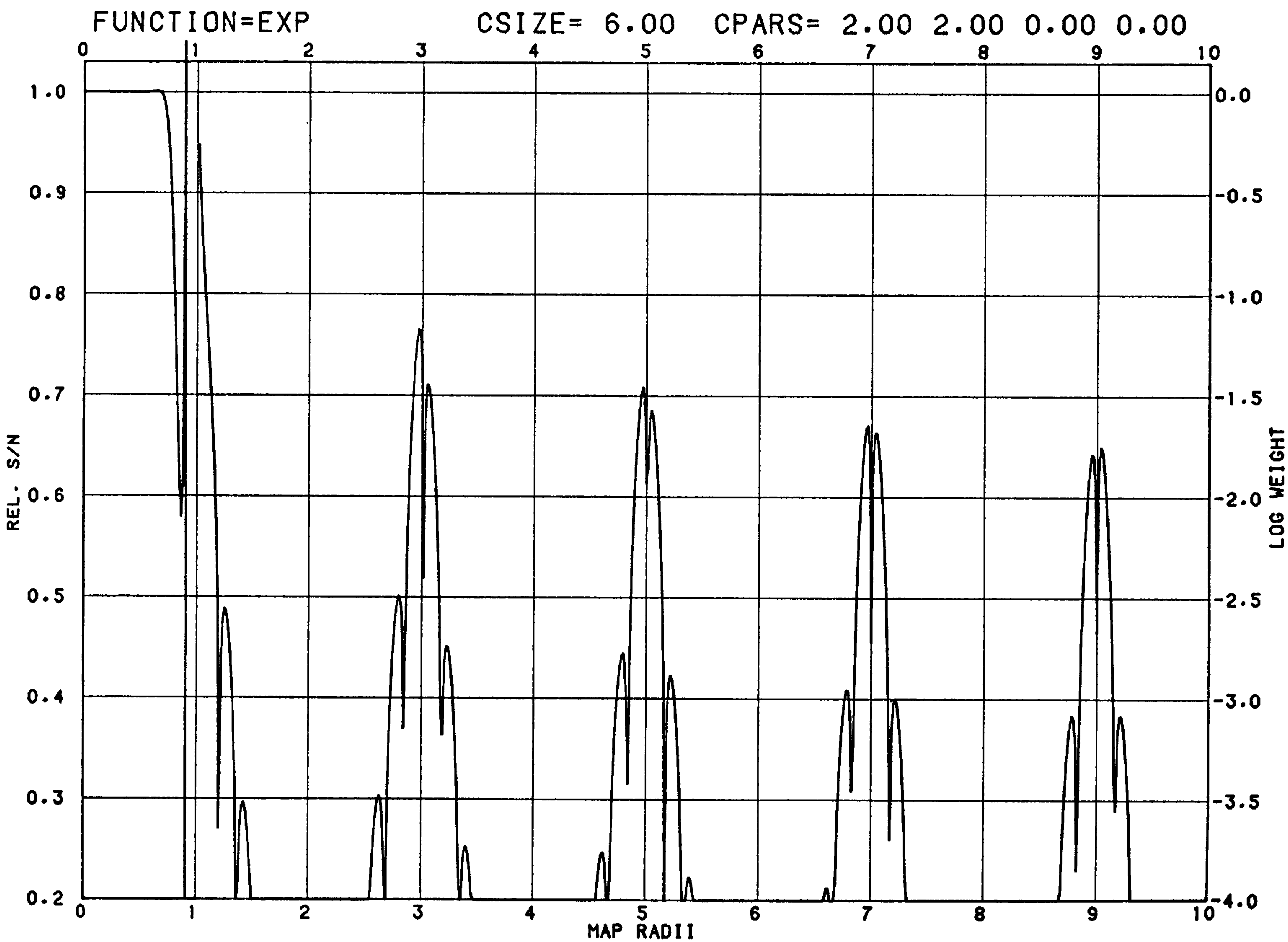

FUNCTION=EXP
CSIZE= 2.00 CPARS= 0.67 2.00 0.00 0.00
0 1 2 3 4 5 6 7 8 9 10
1.0
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
REL. S/N
0.0
-0.5
-1.0
-1.5
-2.0
-2.5
-3.0
-3.5
-4.0
LOG WEIGHT
0 1 2 3 4 5 6 7 8 9 10
MAP RADII

FIGURE 2d



FUNCTION=EXP
CSIZE= 6.00
CPARS= 2.00 2.00 0.00 0.00
0 1 2 3 4 5 6 7 8 9 10
1.0
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
REL. S/N
0.0
-0.5
-1.0
-1.5
-2.0
-2.5
-3.0
-3.5
-4.0
LOG WEIGHT
0 1 2 3 4 5 6 7 8 9 10
MAP RADII

FIGURE 3a

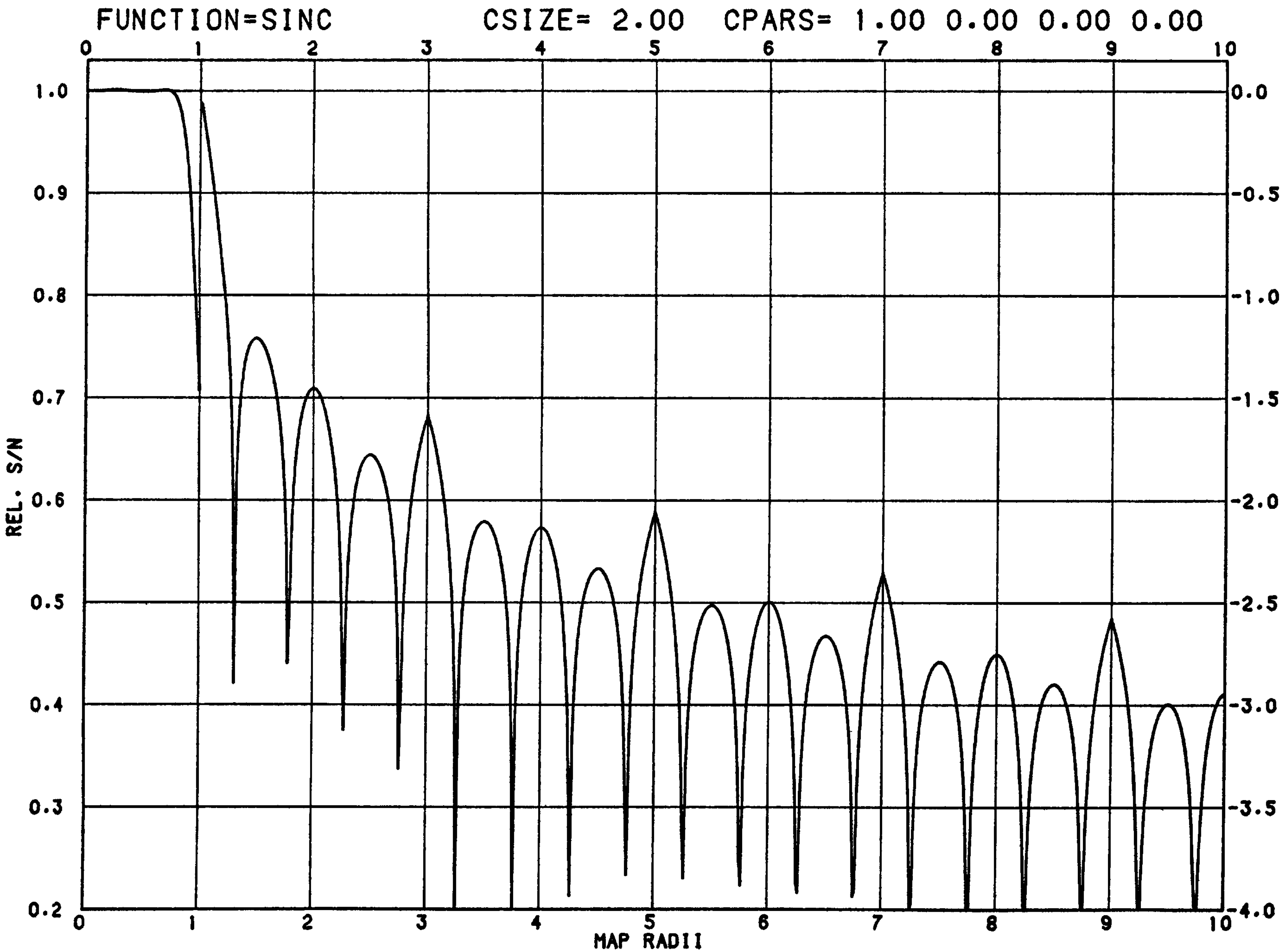
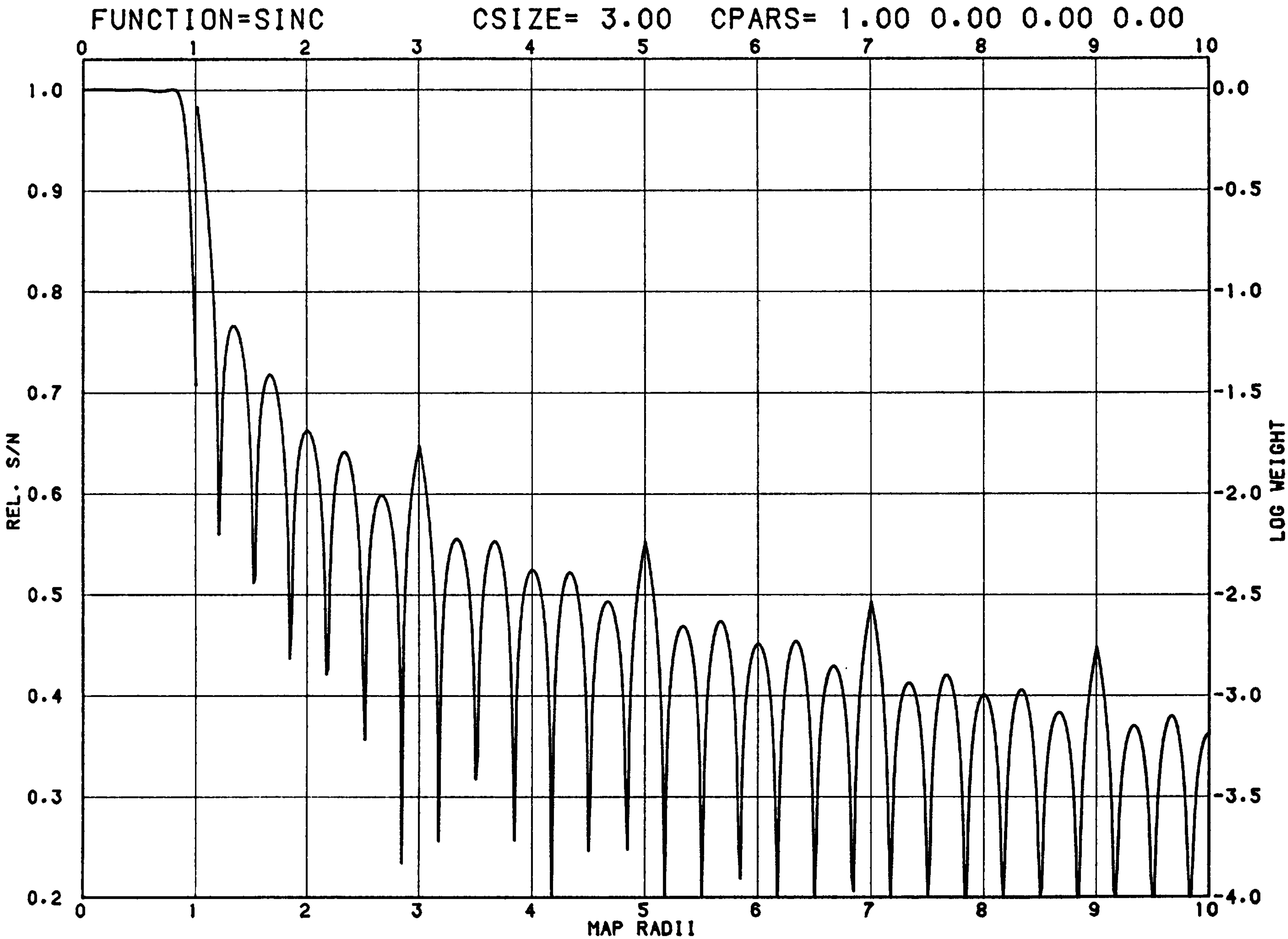
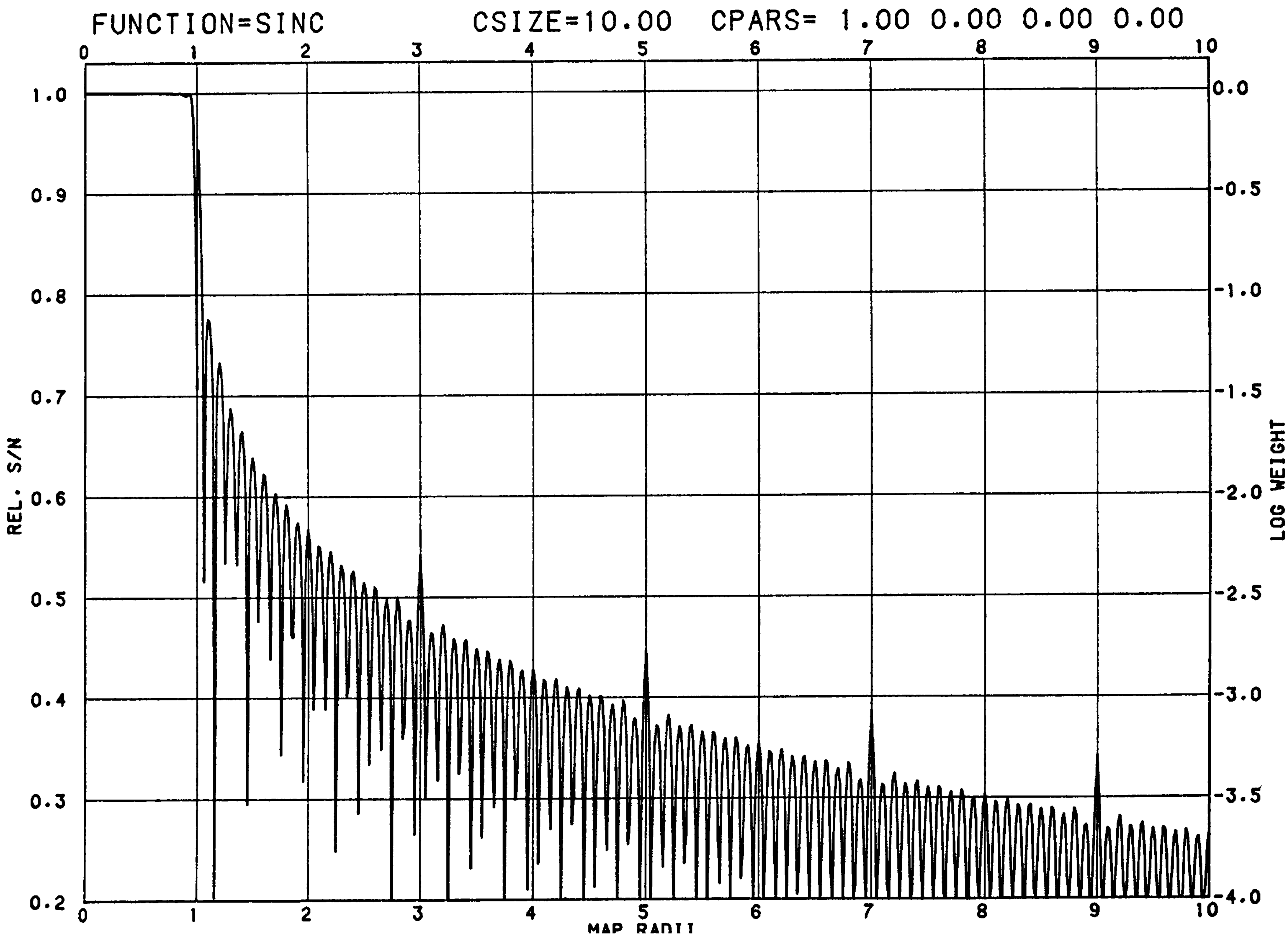

FUNCTION=SINC CSIZE= 2.00 CPARS= 1.00 0.00 0.00 0.00
0 1 2 3 4 5 6 7 8 9 10
1.0
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
REL. S/N
0.0
-0.5
-1.0
-1.5
-2.0
-2.5
-3.0
-3.5
-4.0
LOG WEIGHT
0 1 2 3 4 5 6 7 8 9 10
MAP RADII

FIGURE 3b



FUNCTION=SINC CSIZE= 3.00 CPARS= 1.00 0.00 0.00 0.00
0 1 2 3 4 5 6 7 8 9 10
1.0 0.9 0.8 0.7 0.6 0.5 0.4 0.3 0.2
REL. S/N
0.0 -0.5 -1.0 -1.5 -2.0 -2.5 -3.0 -3.5 -4.0
LOG WEIGHT
MAP RADII

FIGURE 3c



FUNCTION=SINC CSIZE=10.00 CPARS= 1.00 0.00 0.00 0.00
0 1 2 3 4 5 6 7 8 9 10
1.0
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
REL. S/N
0.0
-0.5
-1.0
-1.5
-2.0
-2.5
-3.0
-3.5
-4.0
LOG WEIGHT
0 1 2 3 4 5 6 7 8 9 10
MAP RADII

FIGURE 4a

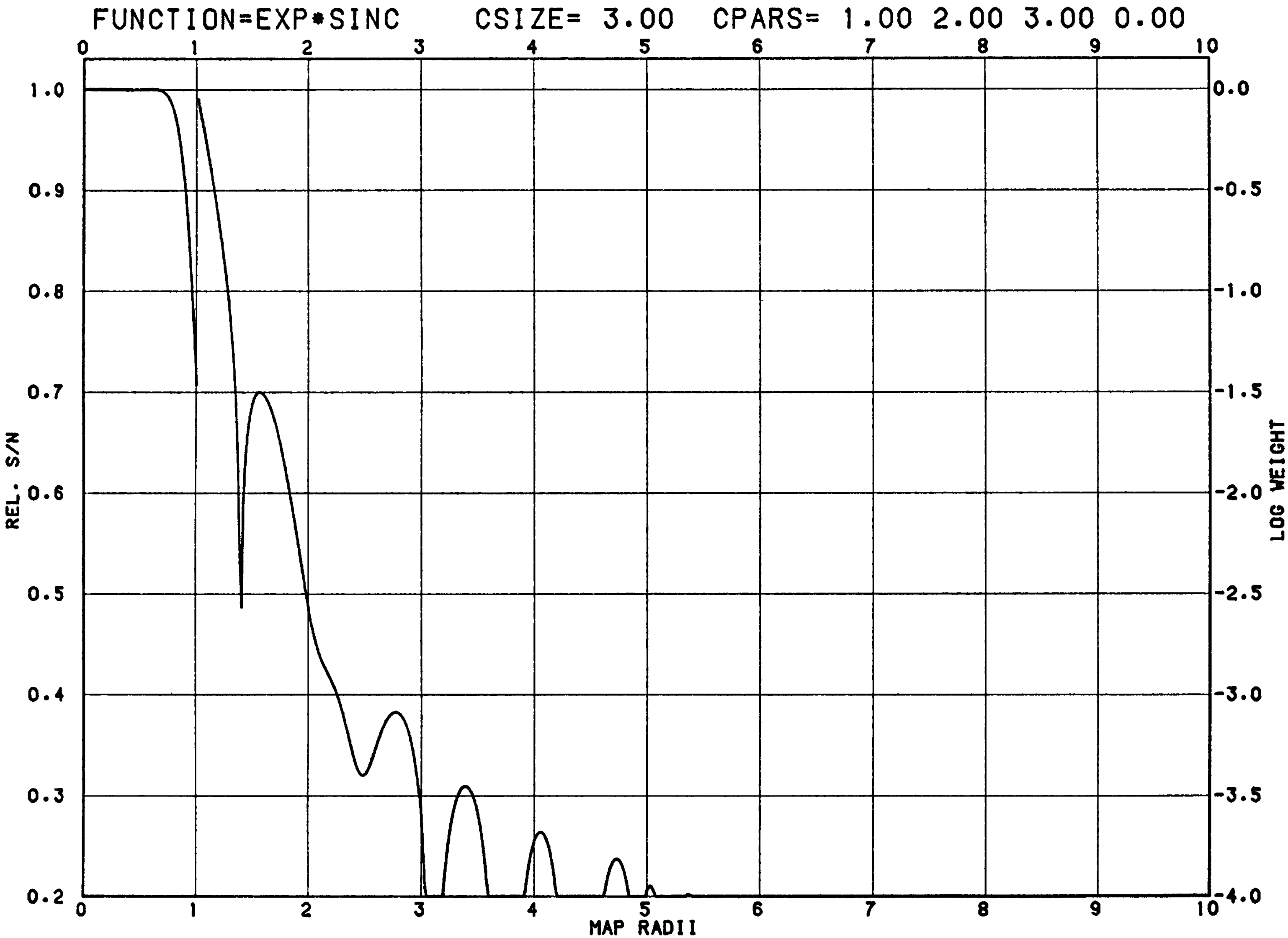

FUNCTION=EXP*SINC CSIZE= 3.00 CPARS= 1.00 2.00 3.00 0.00
0 1 2 3 4 5 6 7 8 9 10
1.0 0.9 0.8 0.7 0.6 0.5 0.4 0.3 0.2
REL. S/N
0.0 -0.5 -1.0 -1.5 -2.0 -2.5 -3.0 -3.5 -4.0
LOG WEIGHT
MAP RADII

FIGURE 4b

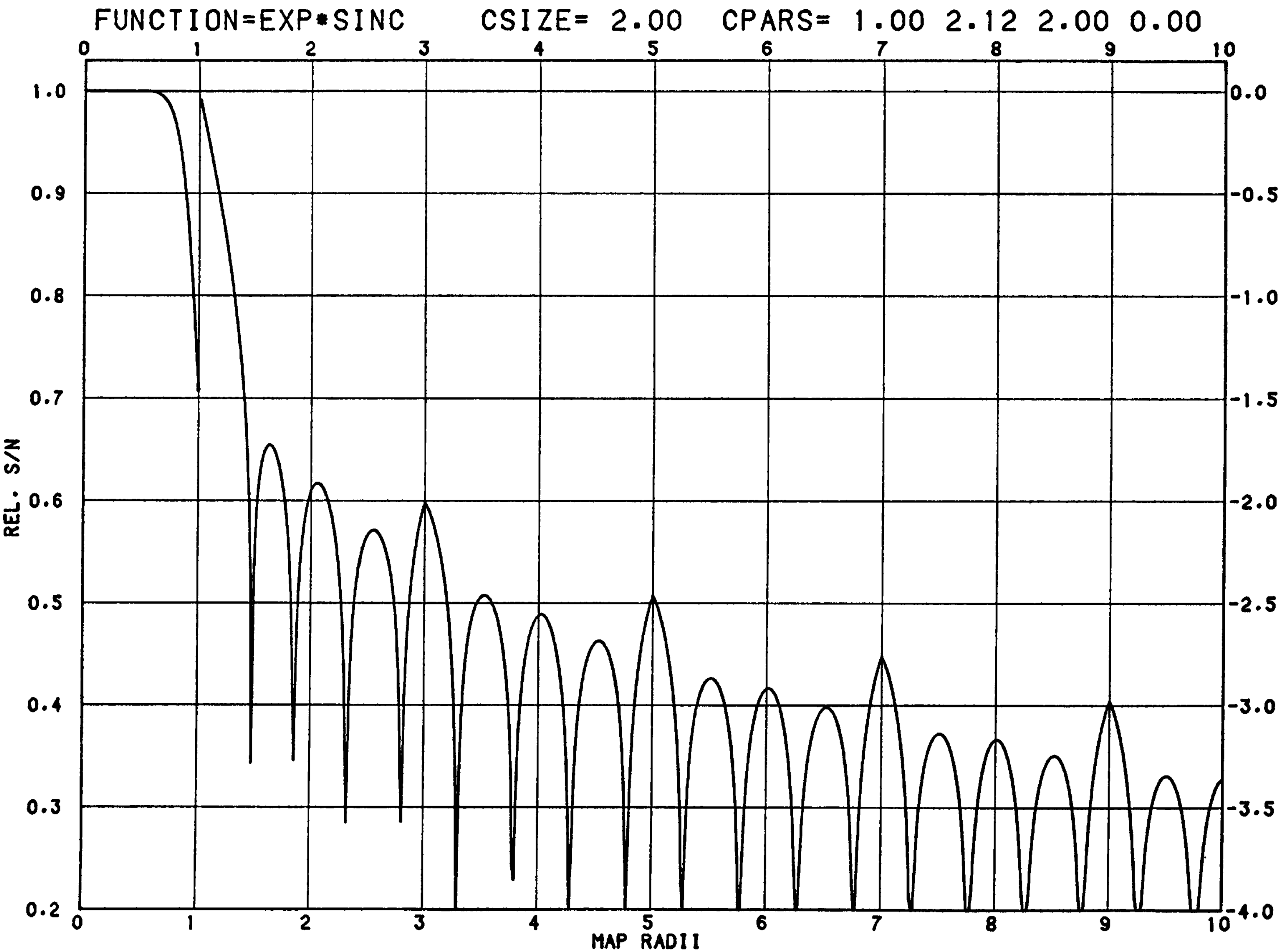

FUNCTION=EXP*SINC CSIZE= 2.00 CPARS= 1.00 2.12 2.00 0.00
0 1 2 3 4 5 6 7 8 9 10
1.0 0.9 0.8 0.7 0.6 0.5 0.4 0.3 0.2
REL. S/N
0.0 -0.5 -1.0 -1.5 -2.0 -2.5 -3.0 -3.5 -4.0
LOG WEIGHT
0 1 2 3 4 5 6 7 8 9 10
MAP RADII

FIGURE 4c

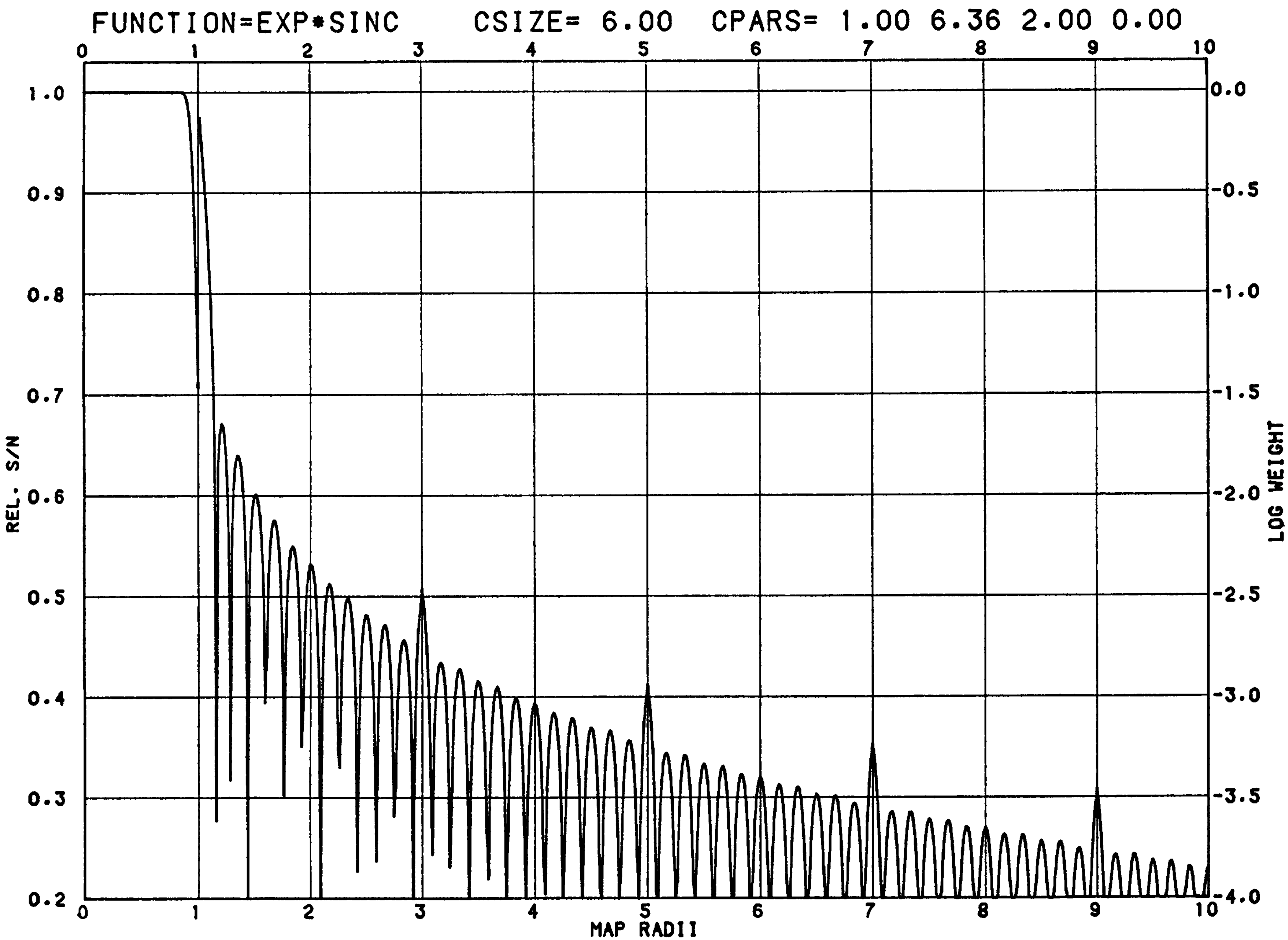

FUNCTION=EXP*SINC CSIZE= 6.00 CPARS= 1.00 6.36 2.00 0.00
REL. S/N
LOG WEIGHT
MAP RADII
0
1
2
3
4
5
6
7
8
9
10
1.0
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.0
-0.5
-1.0
-1.5
-2.0
-2.5
-3.0
-3.5
-4.0

FIGURE 4d

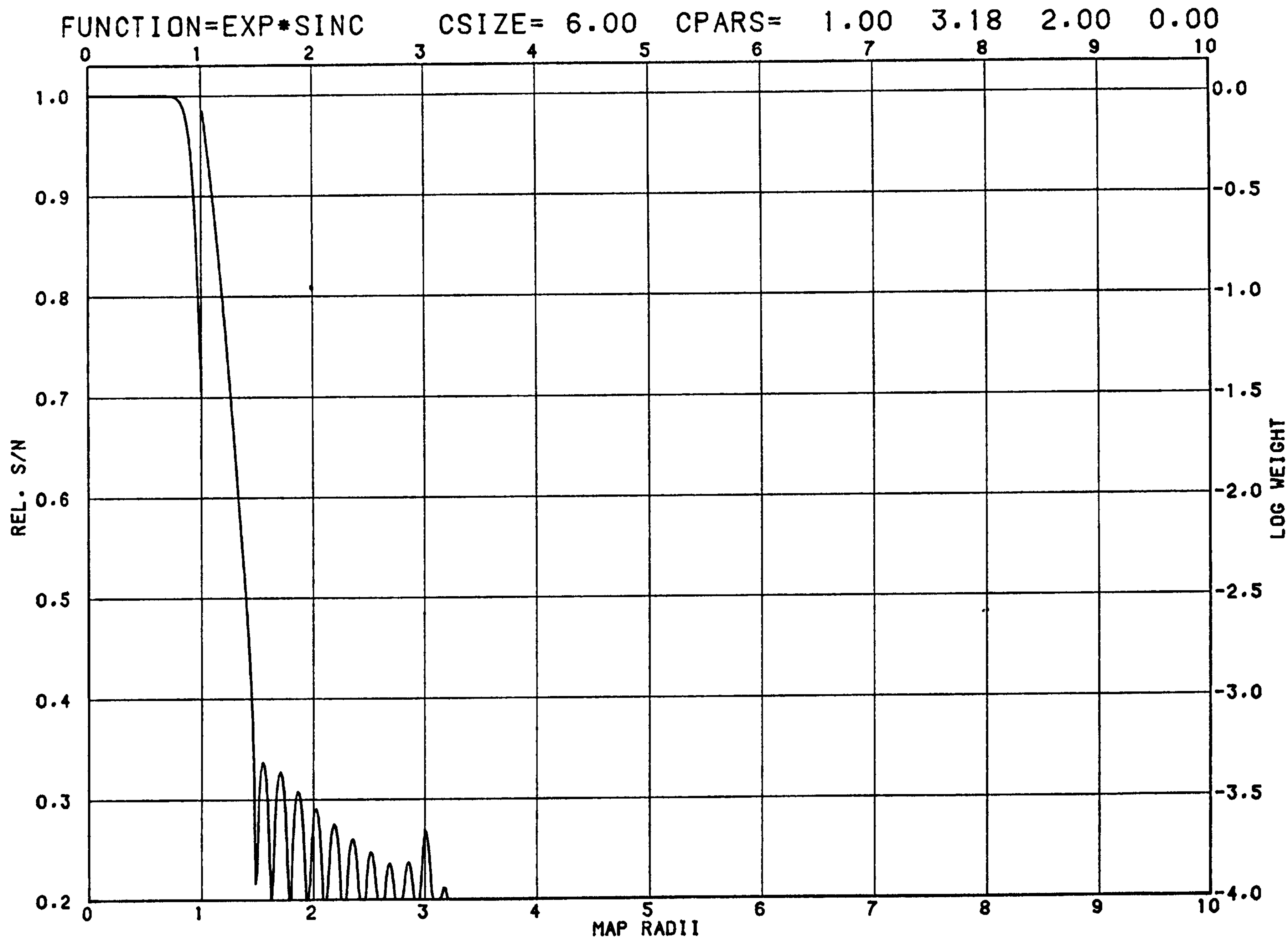
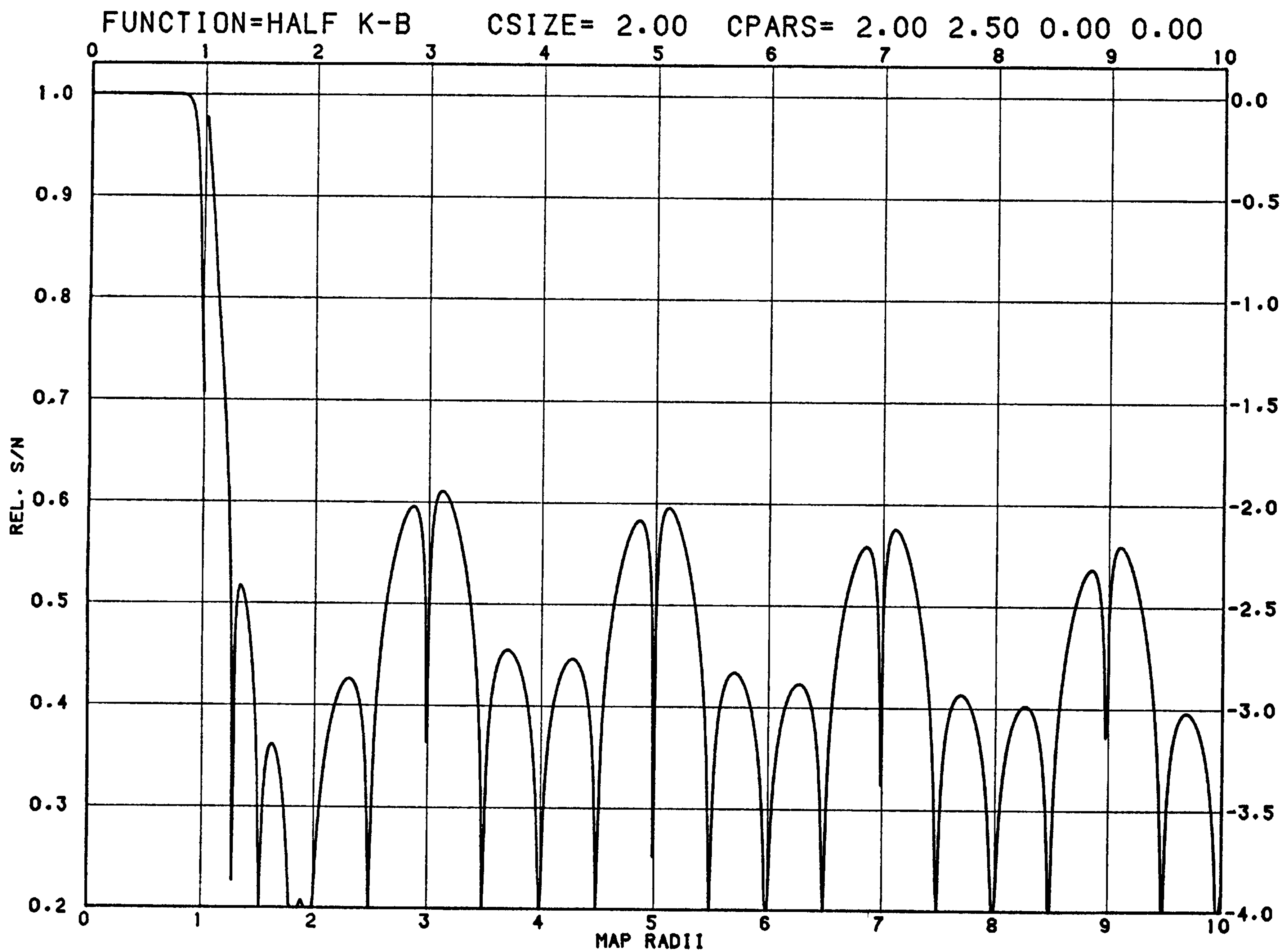

FUNCTION=EXP*SINC CSIZE= 6.00 CPARS= 1.00 3.18 2.00 0.00
0 1 2 3 4 5 6 7 8 9 10
1.0
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
REL. S/N
0.0
-0.5
-1.0
-1.5
-2.0
-2.5
-3.0
-3.5
-4.0
LOG WEIGHT
MAP RADII

FIGURE 5a



FUNCTION=HALF K-B CSIZE= 2.00 CPARS= 2.00 2.50 0.00 0.00
0 1 2 3 4 5 6 7 8 9 10
1.0
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
REL. S/N
0.0
-0.5
-1.0
-1.5
-2.0
-2.5
-3.0
-3.5
-4.0
LOG WEIGHT
MAP RADII

FIGURE 5b

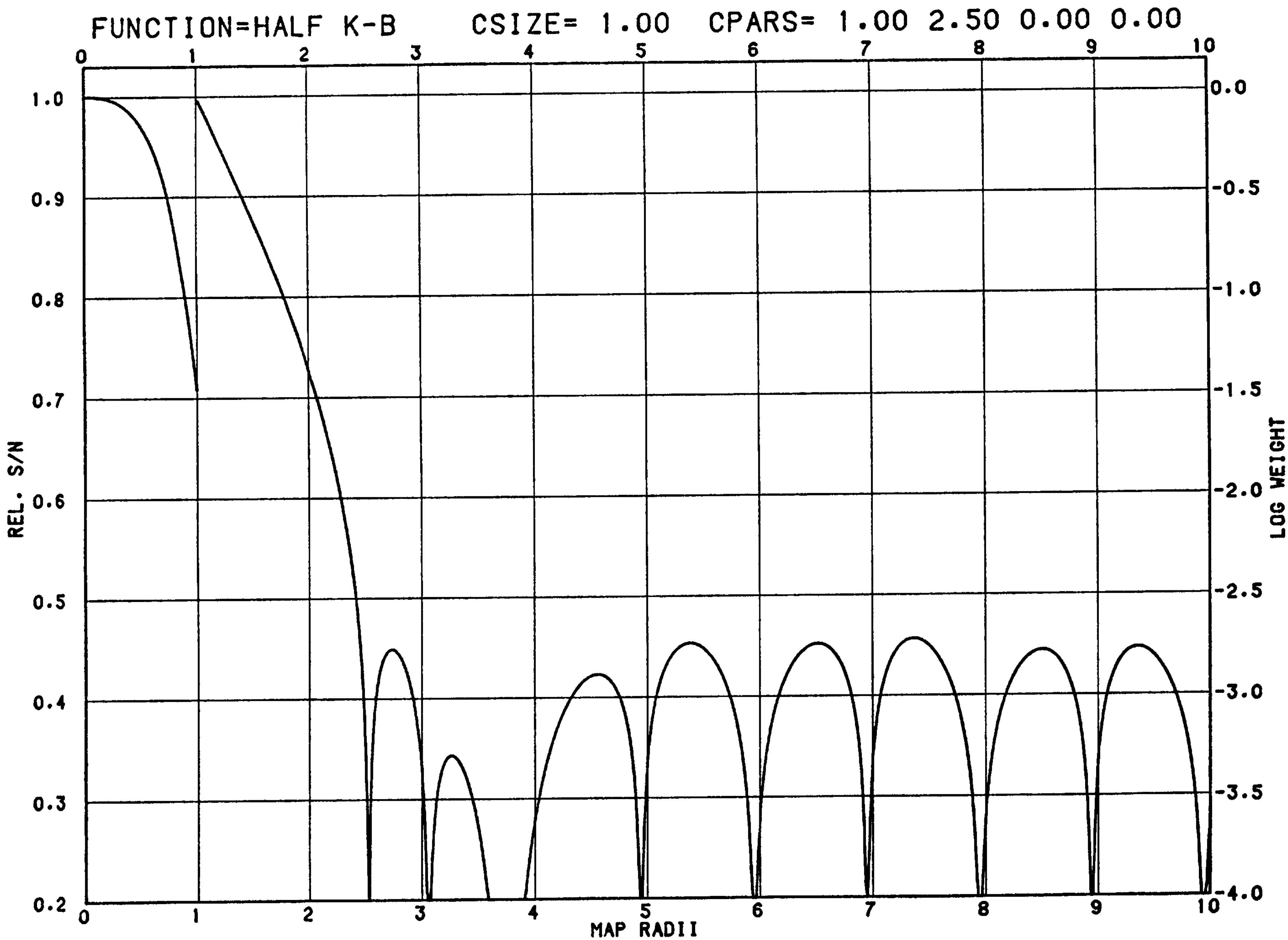

FUNCTION=HALF K-B CSIZE= 1.00 CPARS= 1.00 2.50 0.00 0.00
REL. S/N
LOG WEIGHT
MAP RADII
0 1 2 3 4 5 6 7 8 9 10
1.0 0.9 0.8 0.7 0.6 0.5 0.4 0.3 0.2
0.0 -0.5 -1.0 -1.5 -2.0 -2.5 -3.0 -3.5 -4.0

FIGURE 5c

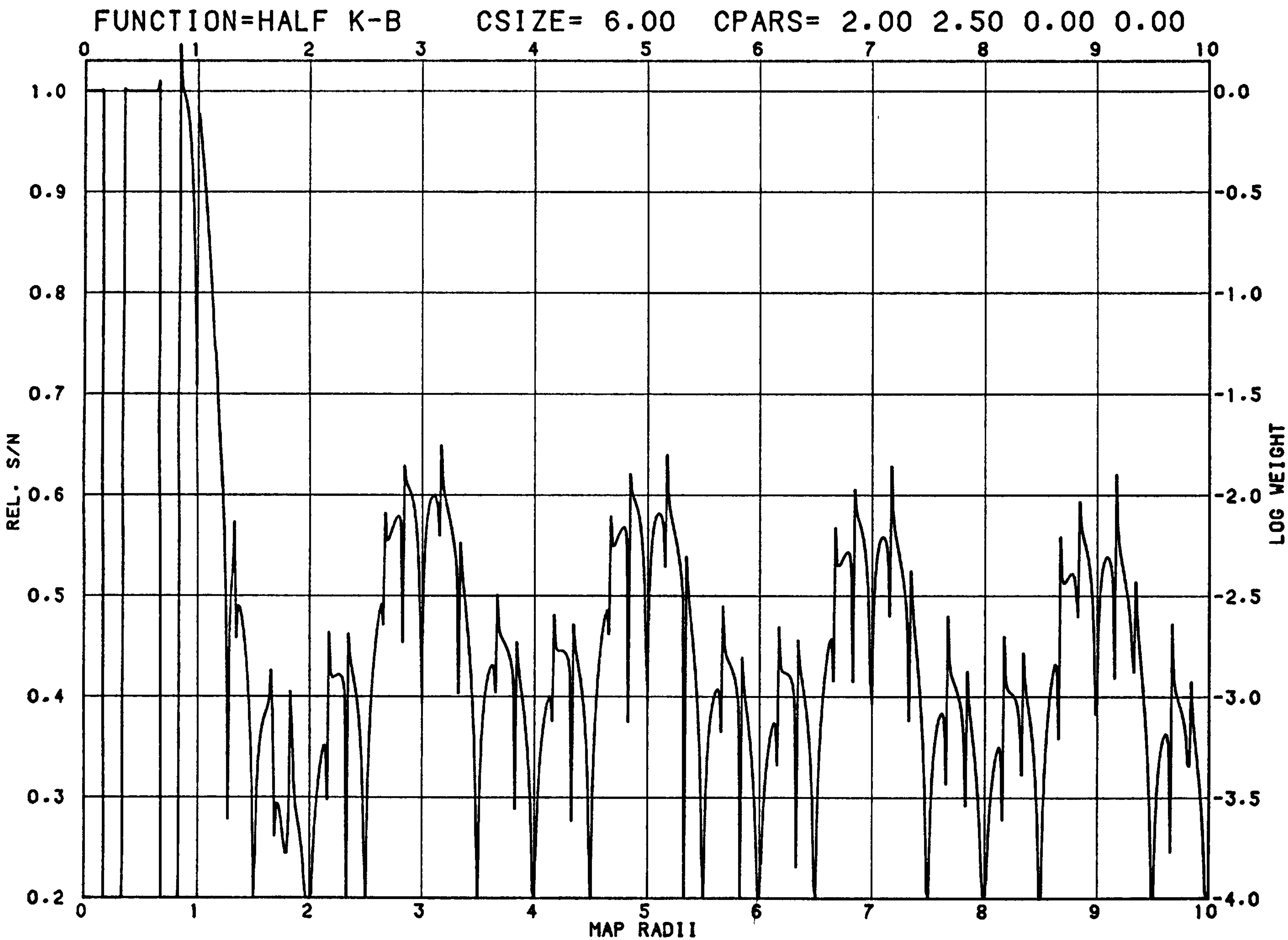

FUNCTION=HALF K-B CSIZE= 6.00 CPARS= 2.00 2.50 0.00 0.00
0 1 2 3 4 5 6 7 8 9 10
1.0
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
REL. S/N
0.0
-0.5
-1.0
-1.5
-2.0
-2.5
-3.0
-3.5
-4.0
LOG WEIGHT
MAP RADII

FIGURE 6

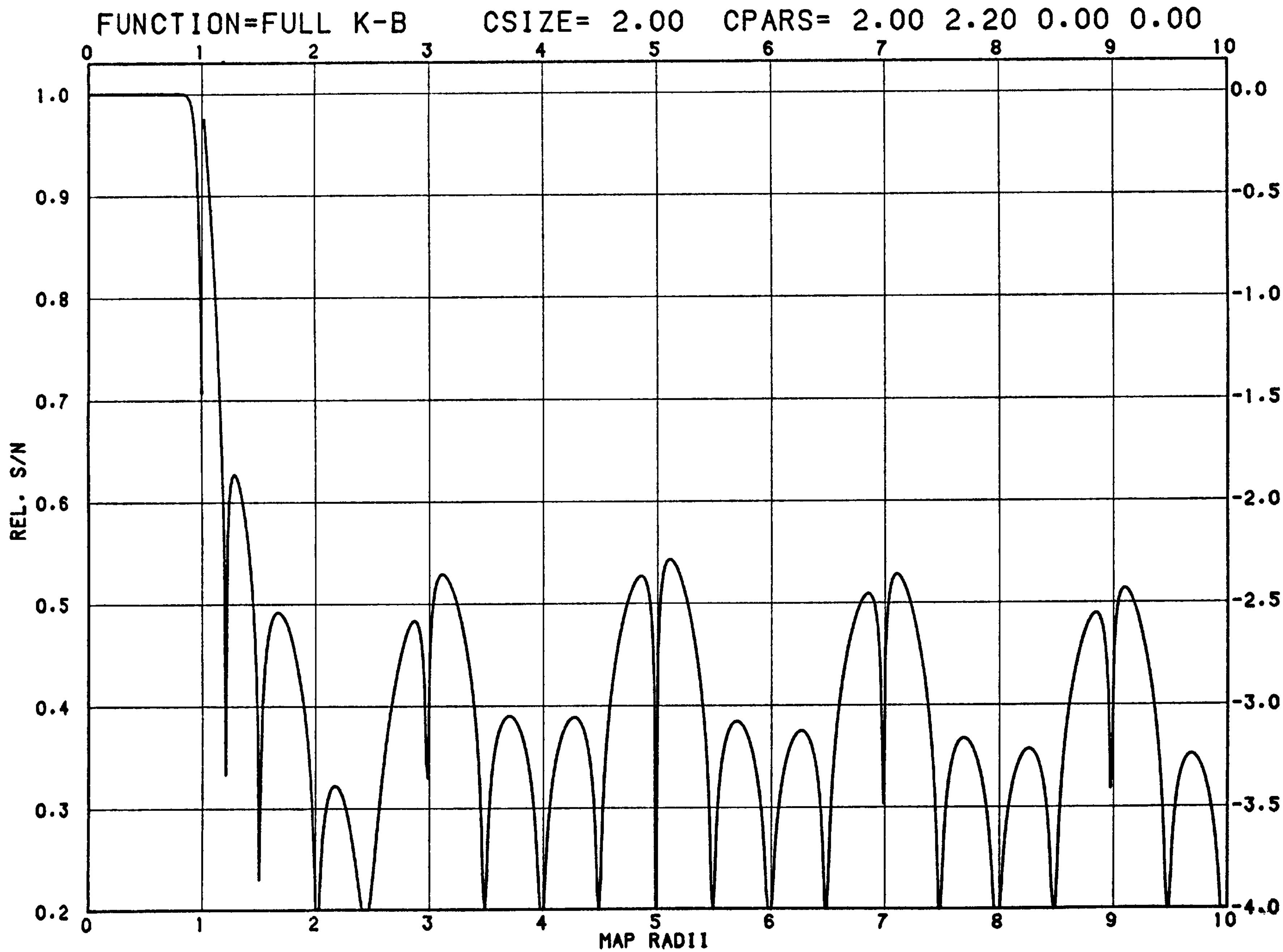

FUNCTION=FULL K-B CSIZE= 2.00 CPARS= 2.00 2.20 0.00 0.00
REL. S/N
LOG WEIGHT
MAP RADII
0 1 2 3 4 5 6 7 8 9 10
1.0 0.9 0.8 0.7 0.6 0.5 0.4 0.3 0.2
0.0 -0.5 -1.0 -1.5 -2.0 -2.5 -3.0 -3.5 -4.0

FIGURE 7

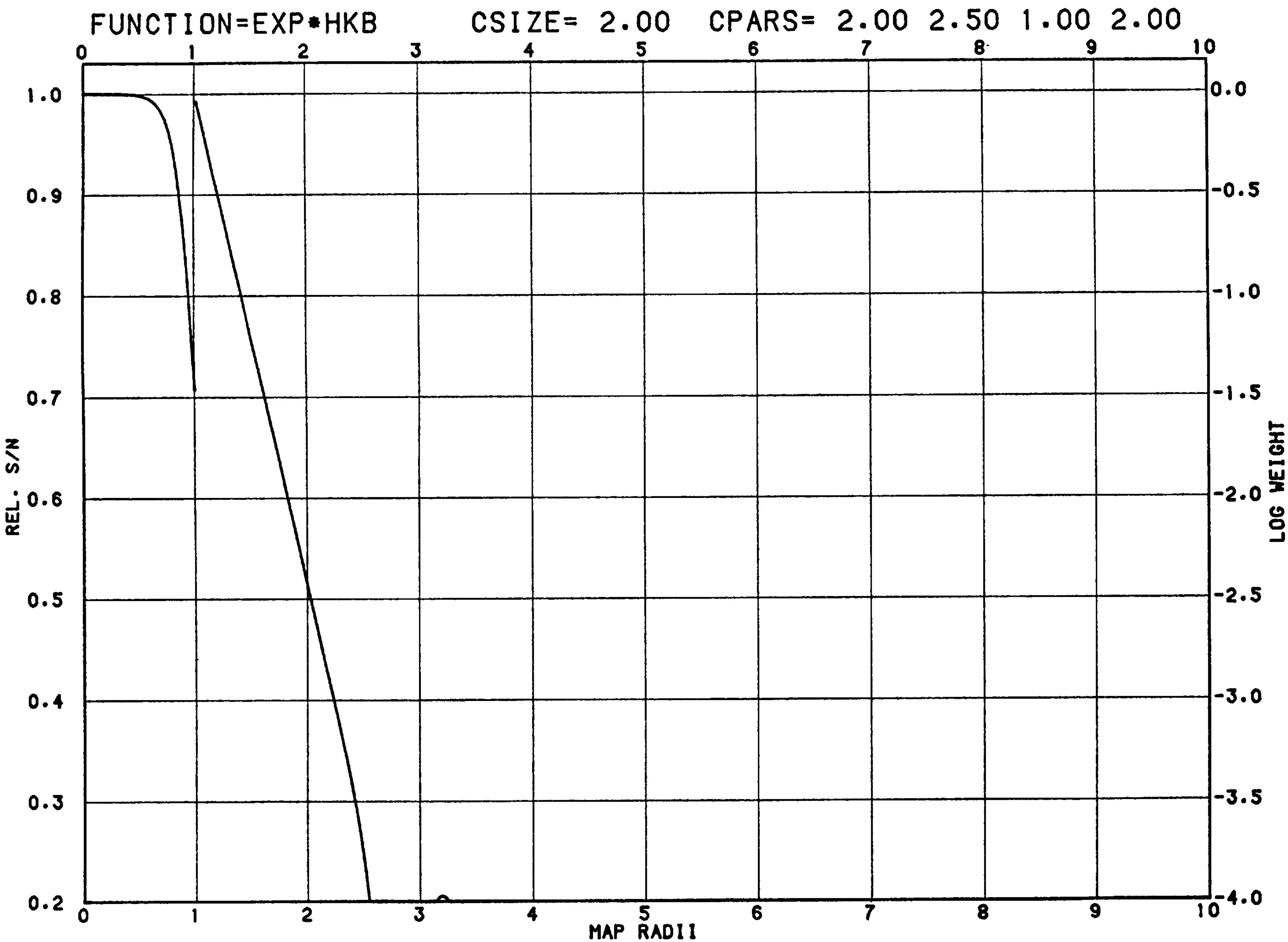
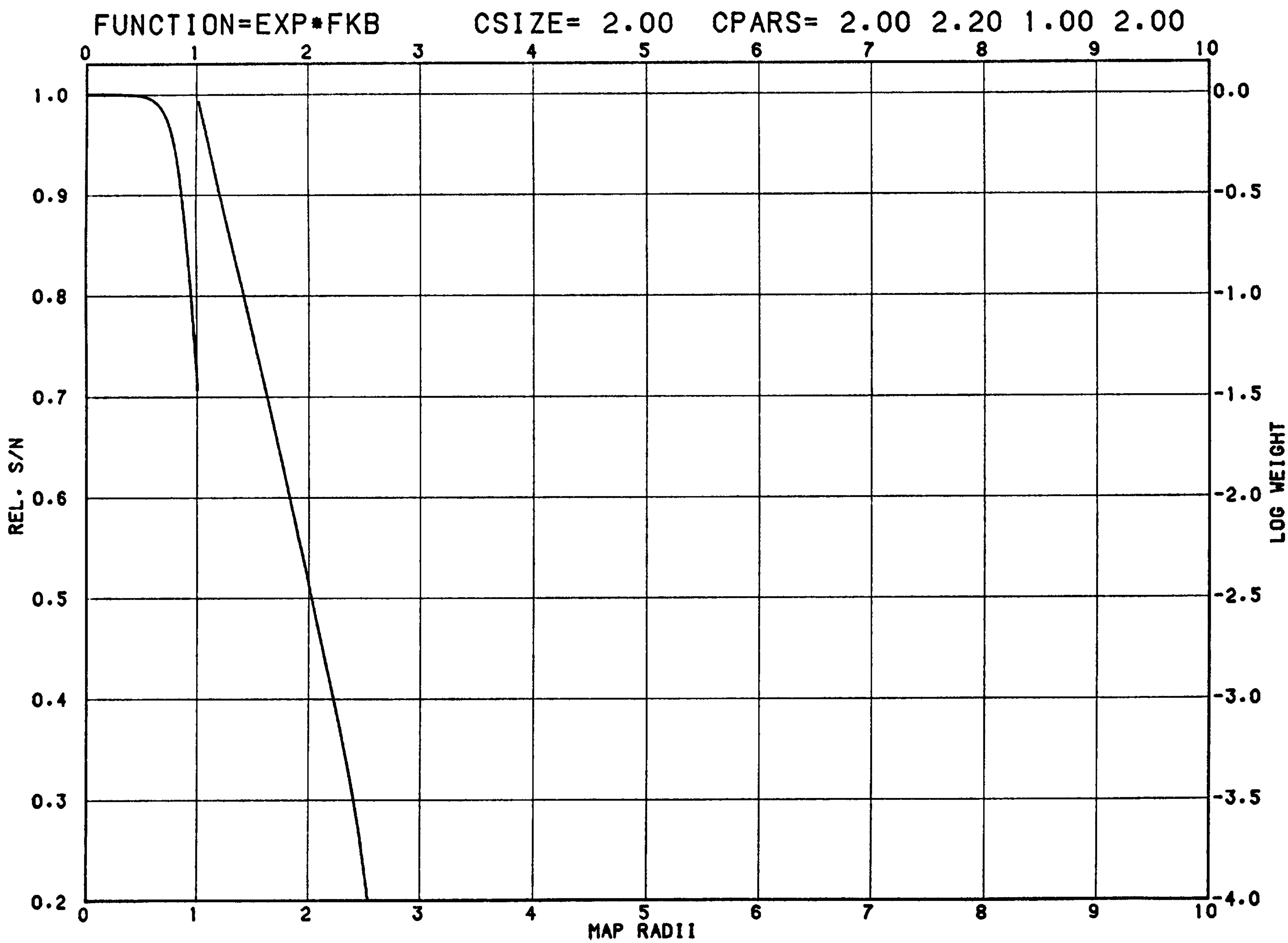

FUNCTION=EXP*HKB CSIZE= 2.00 CPARS= 2.00 2.50 1.00 2.00
0 1 2 3 4 5 6 7 8 9 10
1.0
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
REL. S/N
0.0
-0.5
-1.0
-1.5
-2.0
-2.5
-3.0
-3.5
-4.0
LOG WEIGHT
MAP RADII

FIGURE 8



FUNCTION=EXP*FKB CSIZE= 2.00 CPARS= 2.00 2.20 1.00 2.00
0
1
2
3
4
5
6
7
8
9
10
1.0
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
REL. S/N
0.0
-0.5
-1.0
-1.5
-2.0
-2.5
-3.0
-3.5
-4.0
LOG WEIGHT
MAP RADII