LA--11799-MS

DE91 002435

Transionospheric Propagation Code (TIPC)

Robert Roussel-Dupré
Thomas A. Kelley

Los Alamos National Laboratory
Los Alamos,New Mexico 87545

TRANSIONOSPHERIC PROPAGATION CODE (TIPC)

by

Robert Roussel-Dupré and Thomas A. Kelley

ABSTRACT

The Transionospheric Propagation Code is a computer program developed at Los Alamos National Lab to perform certain tasks related to the detection of VHF signals following propagation through the ionosphere. The code is written in Fortran 77, runs interactively and was designed to be as machine independent as possible. A menu format in which the user is prompted to supply appropriate parameters for a given task has been adopted for the input while the output is primarily in the form of graphics. The user has the option of selecting from five basic tasks, namely (1) Transionospheric propagation, (2) Signal filtering, (3) Signal processing, (4) DTOA study, and (5) DTOA uncertainty study. For the first task a specified signal is convolved against the impulse response function of the ionosphere to obtain the transionospheric signal. The user is given a choice of four analytic forms for the input pulse or of supplying a tabular form. The option of adding Gaussian-distributed white noise or spectral noise to the input signal is also provided. The deterministic ionosphere is characterized to first order in terms of a total electron content (TEC) along the propagation path. In addition, a scattering model parameterized in terms of a frequency coherence bandwidth is also available. In the second task, detection is simulated by convolving a given filter response against the transionospheric signal. The user is given a choice of a wideband filter or a narrowband Gaussian filter. It is also possible to input a filter response. The third task provides for quadrature detection, envelope detection, and three different techniques for time-tagging the arrival of the transionospheric signal at specified receivers. The latter algorithms can be used to determine a TEC and thus take out the effects of the ionosphere to first order. Task four allows the user to construct a table of delta-times-of-arrival (DTOAs) vs. TECs for a specified pair of receivers. In this way, a TEC can be determined by interpolation for any given measurement of DTOA. The fifth task can be used to study the effects of noise and/or scattering on the ability of a particular time-tagging algorithm to measure an accurate DTOA for a given set of receivers. A detailed description of the capabilities and inherent limitations of TIPC is presented. Future enhancements to improve the utility of the code and to address the issue of signal reconstruction are discussed.

I. INTRODUCTION

The purpose of this report is to outline the capabilities and limitations of a computer program (TIPC) developed in the Atmospheric Sciences group at Los Alamos National Lab to perform certain tasks related to the detection of VHF signals following propagation through the ionosphere. The input to this code is menu driven, allowing many options for individual tasks. Once the input is specified the task is executed and the relevant information is plotted both interactively and in a disk file (termed metafile in the Los Alamos computing environment) that is stored and available for post processing. Thus the relevant output in the present version of TIPC is primarily in the form of graphics. For each run a parameter file containing the input data is created. This can be read in for future runs or printed out for a record of TIPC runs. TIPC can also be executed in batch mode by means of command files under the VAX VMS operating system. Although TIPC was originally designed to operate on a VAX, it can easily be converted to run on other systems and more generic routines have been incorporated to facilitate this process.

Many of the algorithms used in TIPC either were written specifically to address a particular aspect of RF propagation or are generic routines extracted from the Common Los Alamos Mathematical Software (CLAMS) library or available through standard math libraries (e.g., IMSL). In addition two computer codes written at other institutions, namely CIRF (Wittwer, Defense Nuclear Agency) and IRI (Rawer *et al.*, 1978a and b; Bilitza, 1986; Rawer and Bradley, 1987) were incorporated into TIPC. A brief discussion of these codes along with appropriate references are included in this report. More recently CIRF has been replaced with a more efficient algorithm that reduces the computing time significantly. The graphics package initially used in TIPC, DISSPLA^{TM}, has been replaced by more readily available software developed at the National Center for Atmospheric Research (NCAR graphics). All of the coding in TIPC is standard Fortran-77. Since its initial release, a number of upgrades in the user interface and in available tasks have been incorporated into the code by the Computer Research and Applications group at Los Alamos. Further planned enhancements are discussed below.

An outline of the tasks performed by TIPC is presented in Section II. A brief discussion of the physical processes inherent to transionospheric propagation of RF signals and the approximations incorporated into TIPC are presented in Section III. A brief statement outlining the tasks performed by each subroutine in TIPC is given in Section IV, while sample code runs with representative output are presented in Section V. A discussion of planned enhancements to the code is given in Section VI.

II. PROGRAM OUTLINE

TIPC was designed to perform five basic tasks related to transionospheric propagation, the detection of RF signals following propagation and signal analysis of pulse waveforms, namely (1) transionospheric propagation, (2) signal filtering, (3) signal processing, (4) DTOA study, and (5) DTOA uncertainty study. Task two utilizes the output of one while task three uses the output of both one and two, and tasks four and five independently use the outputs of tasks one, two, and three. Several supporting calculations that allow the code to treat a broad range of problems of interest are performed under each of these tasks. A detailed description of each is presented below.

TIPC assumes that all input times are specified in microseconds and that the corresponding frequency information is given in MHz. In addition, the propagating electromagnetic field is defined in terms of its electric field component in units of volts/meter. The Fourier transform of the signal has

units of volts/meter · MHz. The output, of course, follows the same convention. Array sizes are fixed in a set of common blocks. Complex arrays with dimension 10,001 points are used to store information in the frequency domain while real arrays of the same dimension are used for corresponding data in the time domain. The restricted size of these arrays leads to inherent limitations on the problems that can be addressed by TIPC. These will be discussed below and in the following section. The default time resolution Δt is taken to be 0.001 μs with a Nyquist frequency of 500 MHz. This choice provides a framework in which TIPC can address a broad range of problems of interest and at the same time maintain reasonable execution times.

A. Transionospheric Propagation

In this section of the code a convolution of a specified signal with the impulse response function of the ionosphere is performed to obtain the transionospheric signal. This is accomplished by taking the fast Fourier transform (FFT) of the input signal, multiplying against the transfer function of the ionosphere (defined in the following section), and taking the inverse transform of the product to yield the transionospheric signal. Thus, we have

$$E_{TI}(t) = (1/2\pi) \int E(\omega) H_I(\omega) \exp(i\omega t) d\omega \quad . \tag{1}$$

where E_{TI} is the transionospheric signal, $E(\omega)$ is the Fourier transform of the input signal $E(t)$, and H_I is the transfer function of the ionosphere. Equation (1) is evaluated numerically by means of an inverse FFT.

Clearly, the first step in this process is the specification of an input signal. The code provides for a number of optional forms, including **(1) Delta function**. This function is created in the time domain by assigning an amplitude equal to $1/\Delta t$ (V/m), where Δt is the specified time step for the problem, at a single point. The resulting FFT has a flat spectrum with amplitude unity (V/m · MHz). In general it is possible to specify an input pulse that consists of many delta functions separated by specified amounts in time. The default case is a single delta function placed at $t = 4{,}000\Delta t$. This functional form is the simplest representation of an impulsive signal that can be used to elucidate the effects of transionospheric propagation on RF signals. It can also be used to check the accuracy of the numerical results by comparing against analytic forms for the impulse response function of the ionosphere. **(2) Double exponential function**. This form has been adopted as a standard for representing nuclear generated EMP and can be written analytically as

$$E(t) = \frac{E_m \left(\exp[-C(t-t_0)] - \exp[-D(t-t_0)]\right)}{(C/D)^{[-C/(C-D)]} - (C/D)^{[-D/(C-D)]}} \quad , \tag{2}$$

where E is the electric field strength in V/m; E_m is the maximum electric field occurring at $t_m = t_0 + ln(C/D)/(C - D)$; $t \geq t_0$, where t_0 is the starting time for the pulse; $C < D$; and $1/(D + C)$ is the rise time of the pulse and $1/C$ the decay time. The default parameters used in TIPC are $C = 80$ MHz, $D = 100$ MHz, $E_m = 500$ V/m, and $t_0 = 0.5$ μs. **(3) Super Gaussian function**. This functional form can be used to model symmetric pulses with flat tops and fast rise and decay times. The analytic representation is written:

$$E(t) = E_m \exp[-[(t - t_0)/\Delta\tau]^n] \quad . \tag{3}$$

where n is even and effectively defines the shape of the function. The higher values of n correspond to flatter tops with faster rise and decay times. The code also allows for five reflections, with specified delays and relative amplitudes, to be added to the direct signal. The default parameters for the direct pulse are $E_m = 1$ V/m, $t_0 = 0.03$ μs, $\Delta\tau = 0.004$ μs and $n = 4$. Two reflection components are also added, with parameters $E_m = 0.3$ V/m, $t_0 = 0.04$ μs, $\Delta\tau = 0.004$ μs, $n = 4$, and $E_m = -0.2$ V/m, $t_0 = 0.05$ μs, $\Delta\tau = 0.004$ μs, $n = 4$. **(4) Modified Gaussian function**. This form has been used to represent the EMP expected from a CPB device. The analytic form can be written:

$$E(t) = \beta I_{net} \sin\theta / [(1 - \beta\cos\theta)\ R] \quad , \tag{4}$$

with

$$I_{net} = I_0 \left\{ \frac{1}{(1 + \exp[a_0(t - t_w)/t_f]} - \frac{1}{1 + \exp\left[\frac{a_0 t}{t_r}\right]} \right\} \quad , \tag{5}$$

and where β is the beam velocity divided by the speed of light, θ is the polar angle of the observer off the beam axis, R is the direct distance from beam origin to observer in meters, I_0 is the initial beam current at exit in kA, t_r is the current 10% to 90% rise time in μs, t_f is the current 90% to 10% decay time in μs, t_w is the pulse width (FWHM) in μs, and a_0 has a fixed value of 4.4. The code also allows for up to five reflections with specified delays and relative amplitudes to be added to the direct signal. The default parameters for this case are $I_0 = 10$ kA, $t_r = 0.01$ μs, $t_f = 0.02$ μs, $t_w = 0.05$ μs, $\theta = 90$ degrees, $\beta = 0.999$, and $R = 250.000$ m. The default pulse does not include reflections. **(5) Specified input pulse**. This option allows the user to designate an input file containing an arbitrary shaped pulse. The first line of the file specifies the format in which the subsequent amplitude and time information will be read (see the TIPC User's Manual for further details). The pulse is read in and interpolated onto the appropriate grid.

As discussed in the following section, in the absence of birefringent effects introduced by the geomagnetic field and to first order, the ionosphere can be specified in terms of the total electron content (TEC, $\equiv \int n_e\ ds$) along the line of sight and its primary effect is to disperse the input signal in time, i.e., the ionosphere introduces a frequency-dependent delay in the arrival of different frequency components. These delays are sufficiently large relative to the duration of signals of interest that the individual frequency components become resolvable in time, with the high frequencies arriving first. Near the cutoff frequency of the ionosphere (maximum plasma frequency), the time delays rapidly approach infinity. This sets an absolute lower limit on the frequency that can be modeled computationally, with a practical lower limit being somewhat above cutoff depending on the particular ionospheric model (i.e., total electron content). The user is therefore asked to specify a particular frequency range of interest. When the time difference of arrival between the upper and lower frequencies exceeds the length of the initial time window, the user is asked to supply a new value for the lower frequency such that the entire frequency range fits into the time window (the minimum frequency allowed by the time window is computed and displayed for the user). A filtering procedure is then implemented in the frequency domain so as to avoid aliasing in the time domain. The filtering in the frequency domain is accomplished with a functional form that is flat with amplitude unity over most of the frequency range and falls off exponentially at both the high- and low-frequency ends with two separate exponential forms that are forced to match at the 50% transmission point. This functional form, shown in Figure 1, preserves the signal amplitude through most of the time domain and minimizes ringing near the edges of the window. Clearly, there exists a tradeoff between the latter effect and aliasing. These problems can be avoided by introducing much larger arrays and versions of TIPC with dimensions of 100,001 points have been written. Unfortunately, the cost in processing time is significant [proportional to dimension · Log (dimension)].

TIPC also treats the birefringence of the ionosphere. In this case the user is asked to specify the percent of extraordinary mode to be mixed in with the ordinary mode (0%, the default, corresponds to a pure ordinary wave) in the transionospheric signal, and to identify the hemisphere in which propagation will occur.

In addition to a deterministic component associated with the mean vertical structure of the ionosphere and parameterized in terms of the total electron content, the impulse response function in general also includes a statistical component produced by the large, random horizontal density gradients observed in the ionosphere and parameterized in terms of a frequency coherence bandwidth. The user is given the option of including this effect and asked to input a value for the frequency coherence bandwidth of the ionosphere. Up to 25 realizations of the impulse response function (IRF) and corresponding transfer function are generated in the replacement algorithm for CIRF. In the case of transionospheric propagation, a single realization is selected and convolved against the deterministic response function to obtain the final transionospheric signal. This procedure can be represented mathematically by simply redefining the ionospheric transfer function in equation (1) to be

$$H_I(\omega) = H_D(\omega) H_S(\omega) \quad , \tag{6}$$

where $H_D(\omega)$ is the deterministic part and $H_S(\omega)$ is a single realization of the scattering part, both of which are defined in the following section.

In addition to specifying the propagation channel, the user is also given the option of adding Gaussian-distributed white noise to the input signal. The amplitude of the noise is determined from the signal-to-noise ratio (SNR) specified by the user and the peak signal power, i.e.,

$$P_N = E_p^2 / SNR \quad , \tag{7}$$

where E_p is the signal peak amplitude and P_N is the noise power (standard deviation squared of Gaussian distribution). The noise, however, is actually added to the signal in the frequency domain. Thus, the noise spectrum is defined as

$$E_N(\omega) = r_1 + i r_2 \quad , \tag{8}$$

where r_1 and r_2 are two independent random variables obtained for each frequency value from a Gaussian distribution with standard deviation squared given by

$$\sigma_N^2 = E_p^2 T \Delta t / SNR \quad , \tag{9}$$

where T is the length of the time window and Δt is the sampling time interval. E_N is simply added to the FFT of the input signal, the result multiplied by the transfer function of the ionosphere and an inverse FFT performed [see equation (1)] to obtain the "noisy" transionospheric signal.

TIPC also allows for "colored" noise. In this case the noise amplitude (σ_N) as a function of frequency is defined by the user and read from a specified input file. The input consists of a header indicating the number of points and format of the data followed by a tabular list of frequencies in MHz, RMS amplitudes of the noise in volts/meter · MHz and standard deviations of the noise amplitudes in volts/meter · MHz (see the TIPC User's Manual for further information). The noise is then added to the signal in the same way as outlined above for white noise. Clearly, the particular noise distribution added to the input signal represents only one statistical realization that may or may not reflect the

mean properties of the Gaussian distribution. This procedure takes on meaning only when a series of realizations of some measured quantity is obtained in order to characterize that quantity's statistical properties. This point will be discussed further in the section on DTOA uncertainties.

The final output of this task is the transionospheric signal over a time span that incorporates the specified frequency range. In the presence of noise or scattering, the transionospheric signal represents only one random statistical realization of potential outcomes. The output takes on the form of plots and the user is given the option of displaying any or all of the following: the input pulse, input pulse spectrum, and the transionospheric signal.

B. Signal Filtering

Signal filtering in this context refers to a complex measurement of both the amplitude and phase of the transionospheric signal over a specified bandwidth. The filtering process is simulated by convolving a given filter response against the transionospheric signal, yielding for the filtered transionospheric signal

$$E_{FTI}(t) = (1/2\pi) \int E(\omega) H_I(\omega) H_F(\omega) \exp(i\omega t) d\omega \quad , \tag{10}$$

where $H_F\ (\omega)$ is the transfer function of the filter.

In the present version of TIPC the user is given the option of either specifying the filter transfer function, i.e., amplitude and phase (or real and imaginary parts; see TIPC User's Manual for details of implementation) as a function of frequency, or selecting from two possible filter shapes, namely a narrowband Gaussian filter parameterized in terms of its full width at half maximum (FWHM) and central frequency or a broadband filter parameterized in terms of the upper and lower frequency values at which the filter achieves 50% transmission. The broadband filter is characterized by a rapid rise in transmission (computed by means of two exponentials spliced together at the 50% transmission point) followed by unit transmission over most of the bandwidth and a rapid decay in transmission (computed in the same way as the rise). Up to some maximum (presently set to four, see the TIPC User's Manual) number of filters can be specified for any given run. The default parameters for this task are a single Gaussian filter at 150 MHz with a 2-MHz bandwidth (FWHM).

The final output of this task is the filtered transionospheric signal. A number of plots can be displayed. The user can choose to have the filter profiles in the frequency domain and/or the filter response to a unit impulse in the time domain plotted. Similarly the filtered transionospheric signal can also be displayed in either the frequency or the time domain or both. TIPC also gives the user the option of plotting the power spectrum of the input pulse overlaid with the peak power measured by each of the receivers. The latter plot is particularly useful in indicating the utility of narrowband amplitude measurements in estimating the power spectrum of the original signal.

C. Signal Processing

Two levels of signal processing are implemented in TIPC. The first offers the choice of performing either quadrature detection or square-law detection of the signal. Quadrature detection simply refers to an algorithm that computes the in-phase (I) and 90° out-of-phase (Q) components of the filtered signal and displays the results. Quite often this form of signal processing also requires beating the

received signal against a local oscillator so that the signal can be digitized at a slower rate. The real, 90° phase shifted component *P(t)* of the filtered transionospheric signal is given by

$$P(t) = (i/2\pi) \int_0^\infty [E_{FTI}(\omega) \exp(i\omega t) - E^*_{FTI}(\omega) \exp(-i\omega t)] d\omega. \quad (11)$$

The final I and Q components beat down by a local oscillator of frequency F_{LO} are then

$$I(t) = E_{FTI}(t) \cos(2\pi F_{LO} t) - P(t) \sin(2\pi F_{LO} t) \quad (12)$$

and

$$Q(t) = E_{FTI}(t) \sin(2\pi F_{LO} t) + P(t) \cos(2\pi F_{LO} t) \quad . \quad (13)$$

The code also computes the amplitude [$\equiv (I^2 + Q^2)^{1/2}/\sqrt{2}$] and the phase [$\equiv \tan^{-1}$(Q/I)] of the filtered transionospheric signal.

For square-law detection the envelope of the received signal in any given channel is obtained by squaring the signal and passing it through a low-pass filter whose bandwidth is specified by the user and defaulted to 10 MHz. The shape of the low-pass filter is identical to that of the broadband filter described in Section II.B. Thus

$$E_{SL}(t) = [(1/2\pi) \int E_s(\omega) H_{LP}(\omega) \exp(i\omega t) d\omega]^{1/2} \quad , \quad (14)$$

where E_s (ω) is the Fourier transform of the filtered transionospheric signal squared and H_{LP} (ω) is the transfer function of the low-pass filter.

The second level of signal processing incorporates three different algorithms designed to extract the ionospheric TEC from the transionospheric signal. In general the TEC can be determined from the chirp rate of the signal (FM technique) or by simply measuring the difference in the time of arrival (TOA) of the signal at a minimum of two frequencies (AM technique). In the latter technique a method for specifying an arrival time for the signal in a given channel must be adopted. Although many different methods have been proposed, we have chosen to work with an algorithm in which the one-third and two-thirds amplitude (relative to the maximum) points on the leading edge of the envelope of the filtered signal are time-tagged to obtain a slope from which a TOA is computed by simple extrapolation to a zero-amplitude baseline. TIPC performs this calculation for each channel given the output envelope computed by the square-law detection algorithm. In the FM technique initially developed at Los Alamos National Lab by R. Massey (unpublished), zero crossings of the filtered transionospheric signal are used to determine the period as a function of time (chirp rate of the signal) for each receiver. A corresponding TOA is obtained from the intercept of a straight-line fit to the measured period squared vs. time. This TOA represents the time of arrival of an infinite frequency component, i.e., the effects of the ionosphere are completely removed independent of the filter used. The third algorithm is designed to yield a delta-time-of-arrival (DTOA) between any two receivers. The output of square-law detection (signal envelope) for a given channel (i) is convolved against that of another channel (j) to yield a cross-correlation amplitude [$C_{i,j}(\tau)$] as a function of the time separation (τ) between channels. Thus, we have

$$C_{i,j}(\tau) = \int E_{SL,i}(t) E_{SL,j}(t - \tau) dt \quad , \quad (15)$$

The time of the peak cross-correlation amplitude is computed and defined to be the DTOA for that pair of receivers. This process is repeated for all possible pairs.

The final output of this task depends on the particular subtask selected. In the case of quadrature filtering, four plots consisting of the I-component, Q-component, amplitude, and phase of the transionospheric signal as a function of time are displayed for each channel. For square-law detection, the envelope of the transionospheric signal is plotted as a function of time. The output of the FMTOA subtask consists of plots of period squared vs. time for all receivers, while for the leading-edge TOA, a plot of the envelope of the filtered transionospheric signal is displayed for each channel. For the cross-correlation subtask, a plot of the cross-correlation function vs. the time separation between channels is produced for each pair of receivers.

D. DTOA Study

For this task the user is asked to specify a range of TECs over which to carry out a series of TOA measurements. TIPC then automatically loops through the first three tasks, given the selected settings for each, and accumulates DTOAs for all possible pairs of channels. The user must of course specify a TOA method (FMTOA or Leading-Edge) or the cross-correlation algorithm in signal processing to generate the DTOAs. The DTOAs obtained with any of the selected methods are plotted vs. TEC for all channel pairs. In this way the user can generate a look-up table that yields the actual TEC for a given DTOA measurement for any combination of time-tagging algorithm and system of receivers. A rough estimate of the TEC can be obtained from the DTOA directly, independent of the look-up table, with the formula

$$TEC = 2\pi DTOA/(\beta_1/f_1^2 - \beta_2/f_2^2) \quad , \tag{16}$$

where TEC is in units of 10^{13} cm^{-2}, DTOA is in μs, f_1 and f_2 are the central frequencies of the two channels in MHz, and the β's are defined as

$$\beta_{1,2} = 8.430 \times 10^4/(1 \pm 0.8 \sin\theta, /f_{1,2}) \quad , \tag{17}$$

where θ is the angle between the wave propagation direction and the geomagnetic field. The FMTOA method represents a special case because in principle the effects of the ionosphere are completely removed for any individual channel. The DTOA should therefore be zero or equal to the difference in receiver delays for any pair of channels. Deviations from these values are an indication of the errors incurred by the particular system under study with the FMTOA technique. The final output of this task is a plot of DTOA vs. TEC for each pair of channels.

E. DTOA Uncertainty Study

This task allows the user to study the effect of Gaussian white noise or of a structured ionosphere on the ability of a specified algorithm and system of receivers to determine the ionospheric TEC. The user is asked to specify a range of signal-to-noise ratios or frequency coherence bandwidths over which the study is to be performed. TIPC then loops through tasks one through three, accumulating DTOAs for all pairs of receivers and for up to 25 realizations of each of up to 5 SNRs or frequency coherence bandwidths. The root-mean-square deviation from the mean for the DTOAs as well as the mean are computed and plotted vs. SNR or coherence bandwidth. In this way the user is able to estimate a specified system's ability to determine TEC under noisy or disturbed conditions.

III. IONOSPHERIC MODEL

Among the effects associated with the propagation of electromagnetic waves through a plasma are dispersion, reflection, refraction, absorption, and scattering. We discuss each of these processes and indicate which are treated in the TIPC ionospheric model.

A. Dispersion

The dispersion of an electromagnetic pulse (EMP) incident on a plasma arises because the different frequency components which make up the temporal structure of the pulse propagate at different speeds defined by the index of refraction of the plasma. The presence of the earth's magnetic field also makes the ionosphere a birefringent medium, separating an incident pulse into two modes (ordinary and extraordinary, corresponding roughly to right-hand and left-hand polarization, respectively, in the northern hemisphere), each propagating at a different speed. For the ionosphere the index of refraction is given by the Appleton-Hartree formula (see e.g., Rishbeth and Garriot, 1969; Davies, 1965), which in the limit where the electron collision frequency is small compared to the frequencies of interest reduces to

$$\mu^2 = 1 - \frac{X(1-X)}{(1-X) - Y_T^2/2 \pm [Y_T^4/4 + (1-X)^2 Y_L^2]^{1/2}} \quad , \tag{18}$$

where $X = \omega_p^2/\omega^2, Y = \omega_H/\omega, Y_T = Y\cos\theta, Y_L = Y\sin\theta$, and θ is the angle between the wave propagation direction and the geomagnetic field. The plus (minus) sign in equation (18) corresponds to the ordinary (extraordinary) mode. In the limit $Y_T^4/4 << (1-X)^2 Y_L^2$ (quasi-longitudinal approximation), this equation further simplifies to

$$\mu^2 = 1 - X/(1 \pm Y_L) \quad . \tag{19}$$

Assuming that Y_L is small, the group velocity can be written

$$Vg = c[1 - X/(1 \pm Y_L)]^{1/2} \quad . \tag{20}$$

As is evident from equations (18)–(20), the net effect of dispersion is to introduce a time delay in the arrival of a particular frequency component of the incident pulse at the detector such that the high frequencies arrive first.

The transionospheric signal can be obtained for arbitrary electromagnetic pulses by convolving the pulse time history against the impulse response function of the ionosphere. Alternatively, one can multiply the Fourier transform of the pulse against the transfer function of the ionosphere. The latter can be derived from the equations given above for the index of refraction. In the limit that the quasi-longitudinal approximation applies and when $X << 1$, the transfer function of the deterministic ionosphere can be written

$$H_D(\omega) = \exp(i2\pi\alpha/\omega) \quad , \tag{21}$$

where $\alpha = 8.430 \times 10^4 TEC/(1 \pm Y_L^*)$, TEC is the total electron content along the propagation path in units of 10^{13} cm^{-2}, Y_L^* is taken to be $0.8\sin\theta/f$, and f is the frequency in MHz. Clearly, the choice of $0.8\sin\theta/f$ for Y_L^* assumes some weighted magnetic field strength and angle between the propagation vector and the magnetic field. The particular choice ($f_H = 0.8$ MHz) is typical of midlatitude propagation; however, because of the density weighting along the propagation path it is also possible to derive a more accurate value that depends on the latitude and longitude of the source and the azimuth and elevation angles of the receiver. The latter algorithm is not presently implemented in TIPC. The

angle θ can be determined from the magnetic dip angle at the altitude of peak ionospheric density and the propagation direction.

B. Reflection

Reflection occurs for the frequency components of the incident pulse that are below the peak plasma frequency. For a quiescent ionosphere, the plasma frequency generally falls below 9 MHz. As a result, frequency components at and below this value never arrive at a transionospheric receiver. Those frequencies just above the plasma frequency are strongly dispersed and refracted by the ionosphere and are therefore difficult to measure and interpret. Because the ionospheric model incorporated into TIPC assumes that the EMP frequencies of interest are much greater than the plasma frequency, and given the limited time window used by the code, frequencies below approximately 25 MHz are not well modeled for an ionosphere with a peak plasma frequency of 9 MHz. TIPC accounts for reflection and strong dispersion in effect by filtering out frequencies below that specified by the user. It is, therefore, important to choose a lower frequency limit that is approximately three times the peak ionospheric plasma frequency in order to obtain an accurate transionospheric signal.

C. Refraction

Refraction is a process whereby the direction of propagation of a wave front is altered as a result of a change in the index of refraction of the medium. Refraction has the effect of increasing the path length from source to receiver, thereby increasing the TEC and the corresponding time delay. This effect, however, is second order in the parameter X $(= \omega_p^2/\omega^2)$ and is negligible at VHF frequencies, provided the propagation vector is not at grazing incidence on the ionosphere. We note that the general concept of refraction is applicable only in cases where geometrical optics accurately describes the propagation of the electromagnetic wave. When scattering processes dominate, interference phenomena become important and the simple ideas of geometrical optics break down.

D. Absorption

Absorption of the incident pulse occurs when accelerated electrons transfer their energy to ions and neutrals by collision. In the ambient ionosphere the collision time is large compared to the duration of the EMP and absorption is negligible. In strategic environments, however, enhanced densities lead to higher collision rates and correspondingly higher absorption. Although this effect is included in the general Appleton-Hartree equation, it is not treated in TIPC.

E. Scattering

Scattering is the process by which the propagation path of different parts of an incident wave front are redirected in random directions by the presence of local inhomogeneities or electron-density perturbations. The net result is an interference phenomenon in which random variations in the amplitude and phase of the incident wave, called scintillations, occur. Two important and distinct effects associated with severe scintillations are spatial variations in signal strength, which appear as deep fades to moving communications receivers, and temporal distortions in received signals of broad bandwidth systems, resulting from a mixing of different frequency components of the undisturbed signal. The

former effect has been studied extensively both experimentally with various communications systems (Aarons, Whitney, and Allen, 1971; Umeki, Liu, and Yeh, 1977; Martin and Aarons, 1977; Fremouw *et al.*, 1978; Basu and Kelley, 1979; Basu *et al.*, 1980; Livingston, 1980; Fremouw, Livingston, and Miller, 1980; Rino and Owen, 1980; Whitney *et al.*, 1981; Rino *et al.*, 1981a and b; Johnson, 1981; Basu *et al.*, 1983; Kelley *et al.*, 1985) and theoretically (Booker, Ratcliffe, and Sherin, 1950; Tatarskii, 1971a or b; Budden, 1965; Salpeter, 1967; Lovelace *et al.*, 1970; Barabanenkov *et al.*, 1971; Rufenach, 1975a and b; Bahar and Agrawal, 1976) and is generally characterized in terms of the S_4 scintillation index, which is a measure of the root-mean-square deviation in signal intensity.

The extent of temporal distortions resulting from propagation through a structured atmosphere has been studied both theoretically and experimentally only in the last ten years (Rufenach, 1975a and b; Knepp and Bogusch, 1979; Wittwer, 1979; Wittwer, 1980; Bogusch *et al.*, 1981; Knepp, 1982) and is generally characterized in terms of F_{coh}, the frequency coherence bandwidth of the medium. The impact of scintillations on received signals is to introduce additional uncertainties in signal measurement and time-tagging because of reductions in SNR from fading and distortions in the temporal structure of the signal. The magnitude of these effects is random and can only be characterized statistically. As a result, one cannot compensate for them but must include them in assessing a system's performance.

The theoretical treatment of the scattering of radio waves in the ionosphere proceeds from the parabolic wave equation, which can be solved in its most general form by the use of multiple phase screen (MPS) techniques (see e.g., Knepp, 1982). In the limit of strong scattering conditions, the assumption of a quadratic phase structure function and the thin phase screen approximation combine to greatly simplify the analysis, yielding an analytic solution for the two-position, two-frequency mutual coherence function from which the impulse response function (IRF) can be derived. Physically the IRF represents the signal that would be measured following propagation of a delta function (in time) radio pulse through a thin phase screen. When the IRF is convolved against the mean dispersive effects of the ionosphere and against a given radio pulse, the result is a particular signal realization that is associated with the selected phase screen. Propagation through appropriately chosen phase screens then yields a statistical set of signal realizations from which the accuracies of TOA measurements, for example, can be determined. In the limit of a thin phase screen and assuming an isotropic distribution of irregularities, it is possible to represent the average statistical outcome of a series of such experiments by means of the power impulse response function (representing an average of many statistical realizations of the IRF squared), which can be written (see Knepp 1982, p. 175)

$$G_I(t) = \omega_{coh} \exp\left\{-(\alpha_s \omega_{coh} t)^2/2\right\} \exp\left\{(1-\alpha_s^2 \omega_{coh} t)^2/2\alpha_s^2\right\} \times \Phi_c\left\{(1-\alpha_s^2 \omega_{coh} t)/\sqrt{2}\alpha_s\right\} \quad , \tag{22}$$

where $\omega_{coh} = 2\pi\ F_{coh}$, $\alpha_s = \omega_0/\sigma_\phi \omega_{coh}$, ω_o is the central frequency of the pulse, σ_ϕ is the standard deviation of phase fluctuations as defined below, and Φ_c is the complementary error function. G_I is characterized by a mean time delay $< \tau_I > = 1/\omega_{coh}$ and a width corresponding to a time delay jitter $\sigma_I^2 = (1 + 1/\alpha_s^2)/\omega_{coh}^2$. In this approximation the statistical properties of the phase screens are contained in the parameters α_s and ω_{coh}. These quantities depend on the magnitude of electron-density perturbations and their associated scale sizes in a manner defined below. A single statistical realization of the scattered signal, on the other hand, is written

$$E_S(t) = \left\{G_I(t)/\Delta t\right\}^{1/2} r \quad , \tag{23}$$

where r ($= (r_1 + ir_2)/\sqrt{2}$) is a complex number whose real (r_1) and imaginary (r_2) parts are independent Gaussian random variables with zero mean and unity variance. The scattering transfer function $H_S(\omega)$ for use in equations (6) and (1) is obtained by taking the Fourier transform of equation (23).

The phase screens needed in the above analysis are generated from the known magnitude of electron-density fluctuations and their corresponding spectrum of scale sizes. Assuming the latter distribution to be isotropic and a power law in k-space (k^{-4} in three dimensions), one can derive three important propagation parameters in terms of the RMS electron-density fluctuations (σ_{n_e}), inner and outer scale lengths (L_i and L_0), and propagation distances (z_t – transmitter to phase screen; z_r – receiver to phase screen), namely

$$\sigma_\phi = (2L_i L_0)^{1/2} r_e \lambda_0 \sigma_{n_e} \quad , \tag{24}$$

$$l_0 = \frac{2^{1/2} L_0 (z_t + z_r)}{[\ell n (L_0/L_i) z_t \sigma_\phi]} \quad , \tag{25}$$

and

$$F_{coh} = \frac{4\pi^2 L_0^2 f_0 (z_t + z_r)}{[\ell n (L_0/L_i) \lambda_0 r_e \sigma_\phi^2]} \quad , \tag{26}$$

where r_e is the classical electron radius ($= 2.82 \times 10^{-13}$ cm); λ_0 and f_0 are the central wavelength and frequency, respectively, of the electromagnetic wave; L is the thickness of the phase screen; σ_ϕ represents the RMS phase fluctuation; F_{coh} is the frequency coherence bandwidth; and l_0 is the spatial correlation length associated with scintillations. These parameters together with the S_4 scintillation index describe the effects of strong scattering on a radio pulse propagated through a structured ionosphere.

It is important to note that the above formalism applies only to a narrowband radio pulse with a carrier frequency, f_0, that is well above the plasma frequency. In addition, it is evident from equations (24) (26) that the important propagation parameters depend rather strongly on the carrier frequency (e.g., $F_{coh} \propto f_0^4$), making it difficult to apply this method of analysis to a broadband pulse. The most general form of multiple-phase-screen techniques is necessary to treat the problem of broadband-pulse propagation through a structured plasma.

Nevertheless, to calculate the temporal distortions introduced by propagation through a structured ionosphere, TIPC initially made use of a code (CIRF) developed by Leon Wittwer of LNA (see Dana, 1988). This code computes statistical realizations of the IRF for a structured ionosphere characterized by isotropic irregularities with a 1-D power-law spectral density profile with spectral index of two. The only input to CIRF is F_{coh}; all other propagation parameters are defaulted to values that correspond to a strong scattering environment (i.e., $\alpha_s = 4$). CIRF was converted to a subroutine and modified slightly to return the Fourier transform of a single realization of the IRF. More recently, we have implemented a new algorithm that makes use of equation (23) to generate statistical realizations of the impulse response function. In our new algorithm, we ignore the spatial effects of scattering (assuming the receiver is stationary over the time duration of the signal) and replace the generalized power spectrum $S(k, \tau)$ in equation (3-52) on p. 103 of Knepp (1982) with G_I. The net result is given by equation (23). This method of analysis reduces the computing time significantly and produces results statistically identical to the output of CIRF.

Because the coherence bandwidth is assumed to be a constant across the spectrum of the EMP, the transionospheric signal computed in this way should be considered only as indicative of the effects of scattering on a broadband signal. It is possible, however, to model the performance of **individual**

narrowband receivers precisely by choosing the appropriate value for the coherence bandwidth corresponding to the receiver's central frequency. For studies that require multiple receivers (such as DTOA studies), however, the coherence bandwidth is treated as a constant.

To complete the analysis of pulse propagation through a structured environment, it is also necessary to consider the impact of signal fades associated with amplitude and phase scintillations. This can be accomplished with TIPC by simply varying the SNR in a way commensurate with the statistical amplitude variations to be expected for the desired S_4 scintillation index.

F. Deterministic Ionosphere

The vertical TEC of the ionosphere can vary by more than an order of magnitude depending on the latitude, longitude, solar activity, season, and time of day. It is important, therefore, to have as accurate an estimate of TEC as possible for the particular propagation channel of interest. The International Reference Ionosphere (IRI) (Rawer *et al.*, 1978a and b; Bilitza, 1986; Rawer and Bradley, 1987) is a computer program that generates vertical profiles of electron density, electron and ion temperatures, and positive ion composition for any given geographical location, time of day, season, and solar activity. The electron-density profile is computed by interpolating a series of tables developed from synoptic measurements obtained with ionosondes, topside sounders, and incoherent scatter radars. The data base in some cases constitutes more than 20 years of measurements taken on an hourly basis.

The input parameters to IRI include the geographical location (latitude and east longitude), the sunspot number (available from the Geomagnetic and Solar Data published monthly by the Journal of Geophysical Research), the month, time of day (hour), solar inclination, lower and upper limits of the height profile, and the profile height resolution. The output includes plots and a listing of the ionospheric constituents and electron and ion temperatures as a function of height. The electron-density profile can then be integrated to yield a TEC for input to TIPC.

G. Structured Ionosphere

The existence of electron-density irregularities in the ionosphere has been known for some time (see, e.g., Fejer and Kelley, 1980) and their variability and effect on radio communications has been studied extensively as noted above. Generally, the largest electron-density perturbations are found both at high latitude and at the equator and are associated with magnetic substorms and corresponding electron precipitation in the auroral oval and with equatorial spread-F. In contrast, midlatitude fluctuations tend to be of the order of a few percent of those found in equatorial regions. In order to give the TIPC user a feeling for the magnitude of S_4 and F_{coh}, we present a brief discussion of some of the available data. We note, however, that because midlatitude effects on radio communications are weak, the data available for this region are meager at best and our discussion is therefore limited to the worst-case conditions found in polar and equatorial regions.

The magnitude of ionospheric density fluctuations is either inferred from scintillations in radio signals or measured in situ. The largest RMS density perturbations range from $1-3 \times 10^5$ cm^{-3}, while the corresponding scale sizes are generally characterized by a power-law k-spectrum with a one-dimensional spectral index equal to 2 and outer scale lengths ranging from 1–100 km. In Table I we present typical values for RMS electron density perturbations measured in situ under severe ionospheric conditions, along with the corresponding propagation parameters. The latter parameters were calculated assuming the thickness of the phase screen (L) to be 100 km, a central frequency (f_0) of 150 MHz, a receiver

distance (z_r) equal to 20,000 km, a source distance (z_t) of 300 km, and an inner scale length equal to 200 m. Note that the inner scale length was chosen somewhat arbitrarily, representing the resolution of in-situ measurements in some cases, but that the propagation parameters do not depend strongly on L_i. The ion density measurements of Phelps and Sagalyn (1976) were obtained with the spherical electrostatic analyzer probe on ISIS 1 in the topside ionosphere at high latitudes (60–84 degrees invariant latitude) and altitudes ranging from 574 to 3523 km. Irregularity scale sizes ranging from 200 m to 100 km and irregularity amplitudes as high as 20% of the mean were measured. Assuming that these characteristics persist throughout the ionosphere, one would derive a relevant electron-density perturbation of 10^5 cm^{-3} and the corresponding propagation parameters shown in Table I. Dyson, McClure, and Hanson (1974) report on similar results obtained with the retarding potential analyzer on OGO-6. These measurements included equatorial regions, which were found to possess somewhat higher irregularities than the polar regions. The rocket measurements of Costa and Kelley (1976) were taken in the bottomside ionosphere (altitudes of 270–340 km) over Natal, Brazil, with a Langmuir probe. The results presented in Table I for these measurements are representative of a density hole or plumelike structure near the F-region peak. The OGO-6 measurements reported by Basu, Basu, and Kahn (1976) were obtained with the retarding potential analyzer over various equatorial regions at an altitude of 400 km. The irregularity amplitude of 3×10^5 cm^{-3} is associated with severe ionospheric conditions which are common over the African and Atlantic sectors.

TABLE I
In Situ Measurements of Ionospheric Irregularities

Ionospheric Conditions	L_o (km)	σ_{ne} (cm^{-3})	σ_ϕ (Radians)	ℓ_o (km)	F_{coh} (MHz)
Topside, high-latitude[a] Topside, equatorial[b]	10	10^5	25	20	63
Bottomside, equatorial spread F[c]	10	2×10^5	50	75	16
Near density peak, equatorial[d]	2	3×10^5	34	3.7	2.4

[a]Phelps and Sagalyn (1976).
[b]Dyson *et al.* (1974).
[c]Costa and Kelley (1976).
[d]Basu *et al.* (1976).

Under strong scattering conditions, the existence of a scintillation index approximately equal to one and corresponding signal fades of 20 to 30 dB at VHF frequencies is quite common. RMS phase fluctuations in the tens of radians at VHF are also typical. In Table II, the results of two representative scintillation studies are summarized. Fremouw *et al.*, 1978 report on scintillation measurements taken at the University of Alaska's Poker Flat Rocket Range, the Instituto Geofisico del Peru's receiving site at Ancon, and a receiving station at Kwajalein. The signals originated from a multifrequency (spanning VHF to S band) coherent radio beacon on the DNA Wideband satellite. At VHF (138 MHz) the measurements are characterized by signal fades as deep as 40 dB and phase excursions of several π. The propagation parameters presented in Table II were derived for an RMS phase fluctuation of

TABLE II
VHF Scintillation Measurements

Ionospheric Conditions	L_o (km)	σ_{ne} (cm^{-3})	σ_ϕ (Radians)	ℓ_o (km)	F_{coh} (MHz)
Polar (Poker Flat) and equatorial (Ancon)[a]	1.6	10^5	10	10	19
Equatorial (Ascension Island)[b]	1.0	10^5	8.0	9.3	11

[a]Fremouw *et al.* (1978).
[b]Rino *et al.* (1981).

10 radians and assuming an RMS electron density variation of 10^5 cm^{-3}. The scintillation study of Rino *et al.*, (1981a) is based on measurements taken on Ascension Island, which is located in the South Atlantic ionospheric equatorial anomaly region where strong scintillation effects are normally observed. VHF (250 MHz) signals originating from the Marisat satellite showed deep fades from 20 30 dB. Assuming an RMS electron-density fluctuation of 10^5 cm^{-3} and a magnitude of 10^{21} for their parameter C_s, one obtains an outer scale length of 1 km from their Figure IV-11. These values can be used to derive the propagation parameters presented in Table II.

These results indicate that under worst-case ionospheric conditions, VHF signals will experience deep fades ranging from 20 to 40 dB and strong distortions in their temporal structure for systems with bandwidths greater than approximately 2 MHz.

IV. TIPC SUBROUTINES

Many of the subroutines used in TIPC are available from the standard Fortran math library. On the other hand, some of the routines (e.g., FFT routines) were extracted from CLAMS and others were developed specifically for TIPC. A brief description of the individual routines and their associated input/output is given below. A chart showing the calling sequence and detailed nesting of these calls is given in Appendix A. The general flow of the program is illustrated in Figure 2.

AMAXMIN (ARRAY, *N*, AMAX, IMAX, AMIN, IMIN)

This subroutine searches the first *N* elements of the array ARRAY for the maximum and minimum value. It returns the maximum value in AMAX and its corresponding index in IMAX. It also returns the minimum value in AMIN and its corresponding index in IMIN.

AVGTIME (ARRAY, TAVG)

This subroutine uses the values in ARRAY as weights in constructing the weighted average of the times in TIME. The weighted average is returned in TAVG. NPTS are used in the average.

CFFTB (N, C, WSAVE)

This subroutine in FFT.FOR computes the backward complex (discrete) Fourier transform of the sequence of N points in the array C. WSAVE is a work array that must be first initialized with a call to CFFTI.

CFFTF (N, C, WSAVE)

This subroutine in FFT.FOR computes the forward complex (discrete) Fourier transform of the sequence of N points in the array C. WSAVE is a work array that must be first initialized with a call to CFFTI.

CFFTI (N, WSAVE)

This subroutine in FFT.FOR initializes values in the work array WSAVE for use with CFFTB and CFFTF.

CIRF (CDT, DD, NTIM, CR)

This subroutine computes NTIM values of a random function in the time domain, which is meant to simulate the stochastic features of the ionosphere. The Fourier transform of this function is returned in CR. CDT is the time increment to be used, while DD is the delay interval length to be used. The coherence frequency bandwidth, F_{coh}, is also used in constructing the function.

CLEAR (TTY)

This subroutine clears the user's screen and repositions the cursor. TTY indicates the type of terminal in use. Two types of terminals are supported by this routine: DEC-compatible and Tektronix-compatible terminals.

CROSCOR (INDEX)

This subroutine estimates the difference in times of arrivals for pairs of receivers by determining where the integral of $Ti(t)Tj(t-s)$ over t is maximized (Ti and Tj are the received signals for receivers i and j). This estimate is placed in the array DTOA using INDEX as a guide. This subroutine can also generate a plot of the above integral depending on the flag PLTCC.

CROSINFO (N, ARRAY, J1, J2, SHIFT, DTA, FWHM, TC)

This subroutine examines N elements in the array ARRAY for a maximum. The time corresponding to this point is used in estimating the delta time of arrival which is returned in DTA. J1 and J2 are the two receiver indices and SHIFT is the amount of time displacement found in the input array.

ETTYDEF

This subroutine is a dummy routine for the proper functioning of the OINK library.

ETTYNOP

This subroutine is a dummy routine for the proper functioning of the OINK library.

FFT (NT, NF, DT, TVAL, CFT, FVAL, WORK)

This subroutine computes the forward Fourier transform of a function. NT values of the function in the time domain are taken from TVAL (where DT is the time increment between successive values) and converted to NF values in the frequency domain. The resulting complex spectrum is output in CFT while the magnitude of these values are output in FVAL. WORK is used as a scratch array.

FILTERS

This subroutine establishes the necessary information to define the receivers. The time domain responses are stored in TVAL. The complex Fourier transforms are stored in FIL. The magnitudes of the spectral values are stored in FVAL. Time delays to be associated with each filter are stored in RCVRDLY.

FMTOA (VAL1, VAL2, NPTS, TIME, TOA, LEFT, RIGHT)

This subroutine estimates the time of arrival for a given receiver by the frequency-modulation method initially developed by R. Massey (group SST-7, LANL). It uses the envelope of the received signal in VAL1 to define a peak region. The boundaries of this region are returned in LEFT and RIGHT. The actual received signal in VAL2 is examined over this region. A function which is the square of the difference between successive zeros of the signal versus the time values in TIME is constructed in VAL2 (overwriting the previous values). A straight line is then fit to this function. The point where this line crosses the time axis is returned in TOA as the time of arrival for that receiver.

FRANGE (IESC)

This subroutine determines the frequency range to be used in generating time domain plots so that the plotted signal will fit within the time window allotted. IESC is the escape code entered by the user.

FREQPLOT (YVAR, IST, ND, NCURVES, NPLOTS, IOPT)

This subroutine generates plots in the frequency domain. YVAR is the array of values to be plotted. IST is the starting index for each column of the array while ND is the number of values to plot for each of the NPLOTS. NCURVES is the number of curves to overplot on each graph. IOPT indicates the type of plot desired.

FRFIL (FLO, FHI, COEF, FLTR)

This subroutine constructs a wide-band filter by smoothly patching together exponential functions in such a way that there is 50% transmission at the specified low and high frequencies of FLO and FHI. COEFF governs the steepness of the exponentials. The resulting complex function is returned in FLTR.

GETNOISE

This subroutine retrieves noise data from a file whose name is given in NSFILE and then sorts the values.

GETNP (FR, DM, DS)

This subroutine calculates the noise parameters DM and DS corresponding to a given frequency FR by interpolating in the random noise table.

GETRCVR (NR)

This subroutine retrieves tabular data from a file whose name appears in SRFILE and which defines the frequency response of a given receiver. NR is the index of the receiver.

GETTASK (ITASK, FLAG1, FLAG2, FLAG3)

This subroutine requests from the user the task that is to be performed (returned in ITASK) and calls the appropriate menu routines to establish the necessary parameter values. FLAG1, FLAG2, and FLAG3 indicate whether parameters were modified by the menus corresponding to Transionospheric Propagation, Satellite Detection, and Signal Processing respectively.

GETTEXT (CLUE, IERR)

This subroutine retrieves textual information from the file TIPC.TXT, using CLUE as the string in angle brackets that sets off the particular message and returning in IERR an error condition code.

INITMENU

This block data subprogram initializes the various menu options.

INITTF

This subroutine initializes the array of times in TIME and frequencies in FREQ. It uses the time increment DT and the number of samples points NPTS to do this. It also defines TMAX, TMID, and FMAX.

INITVAL

This subroutine performs the necessary initialization when TIPC is started up.

INITVEC (ARRAY, LOW, HIGH, NUM, LOGSCALE)

This subroutine establishes an array of NUM values in ARRAY by interpolating between the low and high values (LOW and HIGH respectively). LOGSCALE indicates whether interpolation is to be done on a log scale or not.

IOFIRS (CIN)

This function in the OINK library determines the position of the first significant character in the string CIN.

IOGETC (CPROMP, CESCAP, CINOUT, IESC)

This subroutine in the OINK library presents the prompt in CPROMP to the user and accepts the user's input of a string in CINOUT. CESCAP contains a string of "escape" characters. The status of the user's input is returned in IESC.

IOGETI (NEXPEC, CPROMP, CESCAP, IINOUT, NFOUND, IESC)

This subroutine in the OINK library presents the prompt in CPROMP to the user and accepts the user's input of an integer value or array in IINOUT. NEXPEC is the number of values to be expected. CESCAP contains a string of "escape" characters. The number of values entered is returned in NFOUND. The status of the user's input is returned in IESC.

IOGETR (NEXPEC, CPROMP, CESCAP, RINOUT, NFOUND, IESC)

This subroutine in the OINK library presents the prompt in CPROMP to the user and accepts the user's input of a real value or array in RINOUT. NEXPEC is the number of values to be expected. CESCAP contains a string of "escape" characters. The number of values entered is returned in NFOUND. The status of the user's input is returned in IESC.

IOGFIL (CPROMP, CESCAP, CSTAT, CFNAME, IESC)

This subroutine in the OINK library presents the prompt in CPROMP to the user and accepts the user's input of a file name in CFNAME. The string in CSTAT is then used to determine the file's status. CESCAP contains a string of "escape" characters. The status of the user's input is returned in IESC.

IOINIT (CSYS, CASE, NWIDTH)

This subroutine in the OINK library initializes the internal variables that are system dependent or global in nature. This must be called before any other OINK subroutine is invoked. CSYS contains the operating system keyword ("vms," "ctss," or "unix"). CASE indicates the case in which character information is returned to the program from some of the OINK routines ("u" or "l"). NWIDTH defines the width of the terminal screen or window in characters.

IOLAST (CIN)

This function in the OINK library determines the position of the last significant character in the string CIN.

IONMEN (NITEM, CTITLE, NEXPEC, CPROMP, IOPT, NFOUND)

This subroutine in the OINK library handles menus with integer-labeled options. NITEM is the number of items to appear on the menu. CTITLE contains the title to appear above the menu (may be a null string). NEXPEC indicated the number of options that the user is expected to enter. The array of prompts for the menu items is in CPROMP. The user's choices are returned in IOPT. The number of menu items chosen is returned in NFOUND.

IOOFIL (LUNIT, CPROMP, CESCAP, CSTAT, CFNAME, IESC)

This subroutine in the OINK library prompts the user with CPROMP and accepts the user's input of a file name in CFNAME. The string in CSTAT is used as the status in attempting to open the file using the logical unit number specified in LUNIT. CESCAP contains a string of "escape" characters. The status of the user's reply is returned in IESC.

IOUSER

This is a dummy routine for the proper functioning of the OINK library.

IOYORN (CPROMP, CESCAP, CYORN, IESC)

This subroutine in the OINK library presents the prompt in CPROMP to the user in the form of a yes or no question and accepts the user's response in CYORN. CESCAP is a string of "escape" characters. The status of the user's reply is returned in IESC.

IPCPB1

This subroutine generates a super-Gaussian input signal.

IPCPB2

This subroutine generates a modified Gaussian input signal.

IPDELTA

This subroutine generates a set of delta functions for an input signal.

IPEMP

This subroutine generates double exponential function for an input signal.

IPROP

This subroutine simulates the effect of the ionosphere on the input signal. It first adds the noise requested to the input signal. Then it applies to the Fourier transform of the input signal a transfer function that includes phase shifts as a function of frequency in order to simulate the deterministic aspects of the ionosphere. If scintillations are requested, a stochastic transfer function is generated that is also applied to the signal. This subroutine can generate a plot of the transionospheric signal if indicated by PLTTIS.

IPTABLE

This subroutine generates an input signal from tabular data. The name of the file which contains the data is found in IPFILE.

LETOA (ARRAY, NPTS, TIME, TOA, LEFT, RIGHT)

This subroutine estimates the time of arrival for a given receiver by the Leading Edge method. The envelope of the NPTS values of the received signal is in the array ARRAY. Two points on the leading edge where the signal drops to 2/3 and 1/3 of the peak value are found. A straight line is drawn between these two points, and the point where the line intersects the time axis is declared to be the time of arrival for that receiver. The time values in TIME are used for this purpose. The time of arrival is returned in TOA. Also returned are boundaries of the signal peak region, LEFT and RIGHT, for plotting purposes.

MENUDT (FLAG, IESC)

This subroutine requests from the user all the information needed to perform a delta-time-of-arrival study. FLAG indicates whether any of the parameter values were changed. IESC is the escape code corresponding to the user's last response.

MENUDU (FLAG, IESC)

This subroutine requests from the user all the information needed to perform a delta time of arrival uncertainty study. FLAG indicates whether any of the parameter values were changed. IESC is the escape code corresponding to the user's last response.

MENUPS (FLAG, IESC)

This subroutine requests from the user all the information needed to perform signal processing. FLAG indicates whether any of the parameter values were changed. IESC is the escape code corresponding to the user's last response.

MENUSD (FLAG, IESC)

This subroutine requests from the user all the information needed to perform signal detection. FLAG indicates whether any of the parameter values were changed. IESC is the escape code corresponding to the user's last response.

MENUTP (FLAG, IESC)

This subroutine requests from the user all the information needed to perform transionospheric propagation. FLAG indicates whether any of the parameter values were changed. IESC is the escape code corresponding to the user's last response.

NEWSEED (SEED)

This subroutine uses the current system time to establish an initial value for SEED which is used in the generation of random numbers.

NOISE

This subroutine defines the noise that is to be added to the input signal. It can also generate plots of the tabular noise, the input signal plus noise, or the spectrum of the input signal plus noise.

NOISE1

This subroutine generates white noise for the input signal.

NOISE2

This subroutine generates frequency-dependent noise for the input signal based on the tabulated data in a file. The name of the file is stored in NSFILE.

NOPRMT

This is a dummy routine for the proper functioning of the OINK library.

PARAMS (MODE)

This subroutine either saves the parameter values in a file or restores those values from a file depending on the string in MODE. The user is prompted for the name of the parameter file.

PARSEMSG (MSG, KEYWORD, VALUE, IESC)

This subroutine extracts the keyword and value from the first segment of an information message in MSG which is included with tabular data. These are returned in KEYWORD and VALUE respectively. The messages minus the first segment are returned in MSG. An error condition is returned in IESC.

PAUSE

This subroutine prints out an innocuous message and waits for any response from the user.

PLOTEXIT (TTY)

This subroutine in NCAR.FOR terminates any graphics output.

PLOTINIT (TTY)

This subroutine in NCAR.FOR prepares for graphics output.

PLOTXY (X, Y, ISTART, NDIM, NPTS, NC, IPLT, NPLT, IOPT, XLABEL, YLABEL, TITLE1, TITLE2, MESSAGE, TTY)

This subroutine in NCAR.FOR produces one or more X-Y plots. The columns of Y are plotted as a function of X starting at ISTART. NDIM is the column dimension of the two-dimensional array Y. NPTS is the number of points to plot. NC is the number of curves to plot per graph. IPLT is the index of the plot (from 1 to 4) that indicates where in the page it is to be plotted in case there is more than one plot. NPLT is the total number of plots. IOPT indicates the type of scales to use (1 for linear X, linear Y; 2 for linear X, logarithmic Y; 3 for logarithmic X, linear Y; and 4 for logarithmic X, logarithmic Y). XLABEL contains the label for the X axis while YLABEL contains the label for the Y axis. TITLE1 and TITLE2 contain titles and subtitles to be displayed at the top of the page. MESSAGE contains secondary information to be displayed immediately below the titles. TTY indicates the type of terminal being used.

POLY (N, COEF, X)

This function returns the value of a polynomial at X which is specified by N coefficients stored in COEF.

PULSE (IESC)

This subroutine requests from the user the information needed to determine the input signal. Returned in IESC is the escape code corresponding the user's last response.

QUAD

This subroutine performs the quadrature filtering algorithm. It sums the received signal itself and the received signal shifted out of phase by 90 degrees with a fictitious local oscillator. It can plot these quadrature results if indicated by PLTQUAD.

RANDG (MEAN, SIGMA, SEED)

This subroutine generates Gaussian random values with mean and standard deviation specified by MEAN and SIGMA respectively. SEED is automatically updated.

RANDU (SEED)

This subroutine generates uniform random values between zero and one. SEED is automatically updated.

RESTORE (NAME, VALUE, INDX)

This subroutine restores the value of a parameter specified in a line of the parameter file. The parameter's name is given in NAME. If it is an array, the index is given in INDX. The character-string containing the value is given in VALUE.

SATDET

This subroutine models the detection of a transionospheric signal by a specified receiver. It can also generate plots of the received signals.

SAVEC (LUN, NAME, VARIABLE)

This subroutine encodes the value of a character-string variable stored in VARIABLE along with its name in NAME and writes this out to the parameter file using the logical unit number specified in LUN.

SAVEI (LUN, NAME, VARIABLE)

This subroutine encodes the value of an integer variable stored in VARIABLE along with its name in NAME and writes this out to the parameter file using the logical unit number specified in LUN.

SAVEL (LUN, NAME, VARIABLE)

This subroutine encodes the value of a logical variable stored in VARIABLE along with its name in NAME and writes this out to the parameter file using the logical unit number specified in LUN.

SAVER (LUN, NAME, VARIABLE)

This subroutine encodes the value of a real variable stored in VARIABLE along with its name in NAME and writes this out to the parameter file using the logical unit number specified in LUN.

SETTEC (VALUE)

This subroutine establishes a new value for TEC and other global variables whose values depend on TEC.

SIGNAL

This subroutine constructs the input signal and its Fourier transform. It can also generate plots of the input signal.

SORT (N, X, Y, NDIM, NC)

This subroutine sorts the N elements of array X in ascending order and makes sure the elements in each column of Y are permuted in exactly the same way. NDIM is the column size of Y while NC is the number of columns to permute.

SQRLAW

This subroutine passes the signal power through a low-pass filter to determine its envelope.

TASKDT

This subroutine performs a Delta Time of Arrival study by simulating the reception of a transionospheric signal by specified receivers for various values of TEC and estimating the differences in times of arrival.

TASKDU

This subroutine generates various realizations of white noise or scintillations and then simulates the reception of the resulting transionospheric signal by specified receivers. The mean and sigma of the differences in arrival times are calculated.

TASKPS (INDEX)

This subroutine applies one of a number of signal-processing algorithms to the transionospheric signal received by the satellite receivers. INDEX is a pointer into the array DTOA.

TASKSD

This subroutine determines what signals are received by each of the receivers whose frequency response has been specified by the user.

TASKTP

This subroutine determines what happens to the input signal as it passes through the ionosphere.

TIMEPLOT (OFFSET, YVAR, IST, ND, NCURVES, NPLOTS, IOPT)

This subroutine generates one or more plots in the time domain of the values in the array YVAR such that the values in TIME are offset by an amount in OFFSET. IST is the starting index of the values to be used, while ND is the number of values to be used in the plots. NCURVES is the number of curves to plot on each graph, NPLOTS is the total number of plots, and IOPT is an indicator of the type of plot to make.

TSHIFT (CFT, TS, VAL, SAVE)

This subroutine multiplies the complex spectrum in CFT by an appropriate function so that the signal in the time domain is shifted by an amount TS. The resulting time domain representation is returned in VAL. SAVE indicates whether the spectrum is to be updated or not.

WINDOW (ARRAY, *N*, CUTOFF, IST, ND)

This subroutine searches the elements in ARRAY forward from 1 and backward from *N* for the first value that exceeds CUTOFF times the largest element in absolute value. The left index is returned in IST while the total number of points in between the limits is returned in ND.

YSCALE (*Y*, *N*, SCALE, LABEL)

This subroutine in NCAR.FOR scales *N* values in the array *Y* and returns the scaling factor in SCALE. It also returns a scaling label in LABEL that can be used in labeling the ordinates.

V. SAMPLE RUNS

To illustrate the capabilities of TIPC and familiarize the user with the format of its output, we present the results of representative runs for each of the five tasks discussed in Section II. These runs are limited in scope and are meant only to illustrate various aspects of the code. At the same time, however, the selected cases do touch on some interesting facets of propagation, signal detection, and signal processing.

A. Transionospheric Propagation

Two sample input pulses, the delta function and the super-Gaussian function, were run through the transionospheric propagation task for four values of the frequency coherence bandwidth ($f_{coh} = \infty$, 20, 10, and 1 MHz) and two signal-to-noise ratios (SNR = 100 and 10,000). The default pulse parameters were selected in each case and the results are summarized in Figures 3a j and 4a j, respectively. Figure 3a is a plot of the input delta function while Figure 3b is a plot of its spectrum. Note that the signal is zero except at t = 4 μs, where it has amplitude $1/\Delta t$ (the triangular shape is an artifact of the plotting routine and the finite sampling) and its spectrum is flat from zero to the Nyquist frequency with amplitude unity.

The transionospheric signal for a deterministic ionosphere with a TEC of 1×10^{13} cm^{-2} is shown in Figure 3c. Though not apparent in this plot, the initial impulse is separated into its individual frequency components with the high frequencies arriving first. The amplitude of the high-frequency components is larger because the delta function has a flat spectrum and the dispersion due to the ionosphere is higher at lower frequencies. The sudden rise in signal amplitude at 14 μs is an artifact introduced by aliasing.

The effect of scattering on the transionospheric signal is depicted in Figures 3d f. We see that the signal has been modulated in time over a scale defined by the temporal dispersion corresponding to the coherence bandwidth. Thus, the temporal structure in the signal increases as F_{coh} decreases and the modulation scale in time increases as a function of time. The latter result stems from our forcing the coherence bandwidth to be constant over the entire frequency range displayed. In general, a constant coherence bandwidth is not an accurate description of scattering because F_{coh} varies as f_0^4. A general theoretical treatment of this problem is not yet available.

The impact of Gaussian-distributed white noise on the signal is seen in Figures 3g j. The input signal with a SNR equal to 100 is shown in Figure 3g. The corresponding transionospheric signal is plotted in Figure 3h. Because of the tremendous dispersion introduced by the ionosphere, the amplitude of an initial impulse is greatly reduced relative to the noise, which represents a continuous signal that suffers no reduction in amplitude due to dispersion. This explains the obvious reduction in signal-to-noise seen in Figure 3h. The results for a SNR of 10,000 are shown in Figures 3i j. For this case the shape of the transionospheric signal is barely discernible in the noise.

A similar set of results, Figures 4a j, were obtained for the super-Gaussian default pulse. This pulse shape, shown in Figure 4a, has the corresponding spectrum shown in Figure 4b. The latter spectral shape introduces a modulation in the transionospheric signal, plotted in Figure 4c, relative to what one would obtain for a flat spectrum (see Figure 3c). Additional modulation is introduced in the case of finite-coherence bandwidths as seen in Figures 4d f. As in the case of the delta function, the transionospheric signal is lost in the noise for a SNR of 100 (Figure 4h) and barely discernible for a SNR of 10,000 (Figure 4j).

B. Signal Filtering

The transionospheric signals obtained for all of the cases above were run through task two (Signal Filtering) for a set of four Gaussian filters with central frequencies of 40, 70, 150, and 250 MHz and a single bandwidth of 2 MHz FWHM. The filter profiles are shown in Figures 5a d while the corresponding impulse responses are plotted in Figures 6a d. Each filter has a built in group delay equal to approximately 2.12 μs. The filtered transionospheric signals for the delta function input

pulse are shown in Figures 7a–d. Because the bandwidth of these filters and the ionospheric TEC are sufficiently small, the dispersion introduced by the ionosphere across the filter is not readily observable in these signals. We do, however, observe a frequency-dependent delay in the arrival time of the peak signal. For example, the 40 MHz signal (Figure 7a) shows a total delay equal to the sum of the signal delay of 4 μs measured relative to a chosen absolute zero time, the filter delay of 2.12 μs, and the ionospheric delay of 7.97 μs equal to that expected for the central frequency of that filter. A similar result is obtained for the other filters. We also find that the total delay decreases with increasing frequency in a manner consistent with the expected ionospheric delays. By time-tagging the arrival of the signal in different receivers, it is therefore possible to determine the ionospheric TEC. Algorithms developed for this purpose have already been discussed.

The effect of finite coherence bandwidths on the filtered transionospheric signal is shown in Figures 8a–d, 9a–d, and 10a–d. When the coherence bandwidth of the ionosphere is larger than the filter bandwidth, the signal appears virtually unaffected. Closer examination, however, reveals additional time delays and larger temporal widths that are consistent with the expected mean delays and the time delay jitter for the given coherence bandwidth. When F_{coh} falls below the bandwidth of the filter, significant distortion of the signal is observed (Figures 10a–d). Additional delays and a broader filtered signal are also seen. Each of these results represents a single realization of a statistical set of potential outcomes. It is apparent that the statistical delays obtained in the presence of strong scattering introduce random uncertainties in the computed ionospheric TEC's. The additional temporal structure introduced at low coherence bandwidths exacerbates the problem by making it difficult to determine an accurate time of arrival.

The filtered transionospheric signals for a delta function and a SNR of 100 are shown in Figures 11a–d. It is apparent that the signal is washed out by the combination of ionospheric dispersion and noise for all of the received frequency components. For a SNR of 10,000 the filtered transionospheric signal stands out prominently above the noise and has the appropriate delays for all receivers (see Figures 12a–d).

A complementary set of results for the super-Gaussian pulse is displayed in Figures 13a–d, 14a–d, 15a–d, 16a–d, 17a–d and 18a–d. The primary difference between the two pulses stems ultimately from their Fourier spectra. For example, the results show that for a SNR of 100, it is possible in the case of the super-Gaussian pulse to identify a signal in the 70 MHz channel (see Figure 17b) but not possible in the case of the delta function. As seen in Figure 4b, the super-Gaussian pulse has a peak in its spectrum at 70 MHz. The amplitude of this peak is greater than that at the other channel frequencies and is sufficiently large to be detected above the noise. For the super-Gaussian pulse and a SNR of 10,000 (Figures 18a–d) we find that only the 40 and 70 MHz channels have signals sufficiently above the noise to be measured, whereas signals in all channels are detectable for the delta function (see Figures 12a–d). These results indicate that for narrowband systems, signal detection relies strongly on a judicious choice of frequency channels.

The essential aspects of signal detection that are relevant to identification of important spectral features in the radiated pulse are summarized in a series of plots that show the received power in the various channels along with the input pulse spectrum. The results are shown in Figures 19a–f for the delta function and Figures 20a–f for the super-Gaussian function. In the case of infinite coherence bandwidth and SNR (Figures 19a and 20a), we see that the received power in all the channels lies on top of the input pulse spectrum. The effect of a finite coherence bandwidth (Figures 19b–d and 20b–d) is to introduce random fluctuations in the measured spectrum. Some channels experience enhanced power caused by "spectral focusing" while others show reduced power resulting from a defocusing of

the signal in that channel. The effect of noise (Figures 19e f and 20e f) is to obscure the signal in some channels relative to others.

These results illustrate the difficulties encountered in measuring signal characteristics in the presence of ionospheric dispersion, scattering, and Gaussian white noise. The magnitude of these effects depends strongly on the value of TEC, F_{coh}, and SNR, and it is important to have a good estimate or measurement (preferably in situ) for these parameters. It is apparent that narrowband systems suffer from the additional problem of not knowing the optimum channel frequencies *a priori* for arbitrary pulse shapes. Broadband systems that cover the entire VHF range from 30 to 300 MHz are far superior for the task of signal reconstruction.

C. Signal Processing

The default delta-function along with a wideband filter characterized by a central frequency of 130 MHz and a FWHM of 40 MHz were used as input to the quadrature and square-law detection subtasks. The absolute magnitude of the filter transfer function is shown in Figure 21 while the results for quadrature detection and square-law detection are shown in Figures 22a d and 23, respectively. The I and Q components and the phase of the received signal show the effect of the local oscillator (chosen to be 130 MHz). As the signal chirps through the 130 MHz filter frequency, the relative beating of the signal and the local oscillator first decreases and then increases with time, as is evident in Figures 22a, b, and d. The minor differences between the transionospheric signal obtained with square-law detection and the amplitude (Figure 22c) obtained with quadrature detection are associated with the fact that the default 10 MHz low-pass filter was used for square-law detection while the receiver filter has a 40 MHz FWHM.

Results for the TOA algorithms were generated for the default delta function and two filters, namely, a wideband filter from 100 400 MHz and a narrowband filter with a central frequency of 70 MHz and a bandwidth of 6 MHz. The filter transfer functions are shown in Figures 24a and b. Three cases, corresponding to (1) infinite coherence bandwidth and 70 dB SNR, (2) infinite coherence bandwidth and 30 dB SNR, and (3) 10 MHz coherence bandwidth and 70 dB SNR, were run for each TOA algorithm and receiver. The transionospheric signal for each case is shown in Figures 25a c. The results using the FMTOA technique are shown in Figures 26a c and 27a c for the wideband and narrowband filters, respectively. The TOA obtained for any given filter is measured relative to the zero time in our initial time window and therefore corresponds to a signal delay plus some delay caused by the filter characteristics. In the case of the narrowband filter, the FMTOA technique arrives at a TOA of 4.469 μs, which equals the signal delay plus a nominal filter delay of 0.469 μs (see Figure 26a). This result would in general be independent of TEC provided the FMTOA technique compensates accurately for the ionospheric delays. For the broadband filter, the FMTOA technique yields a TOA equal to the signal delay to an accuracy of one nanosecond (see Figure 27a). The effect of a SNR of 1,000 is shown in Figures 26b and 27b. In both cases, the FMTOA technique fails completely. This result illustrates the strong effect of dispersion on the measured SNR at the receiver and our ability to measure the TOA with this technique. For a coherence bandwidth of 10 MHz we find that the narrowband channel makes a 102-ns error in the TOA while the wideband channel makes a 14.0-ns error. On the basis of F_{coh} we expect the rms jitter in the arrival time to be approximately 16 ns. For this particular realization, we find that the narrowband channel shows a larger error than expected while the wideband channel performs as expected. As a result of the phase mixing that occurs within the wideband channel and finite sampling, the FMTOA technique has difficulty measuring the oscillation period at any instant in

time (as evidenced by the large relative variation in period compared to the narrowband channel) but nevertheless measures the chirp rate more accurately because of the broader bandwidth.

Results for the leading-edge analysis are shown in Figures 28a–c and 29a–c. Two basic conclusions can be drawn from this exercise. First, we find that the broadband measurements yield a greater SNR and faster rise times on the leading edge than the narrowband measurements. As a result, the wideband TOA determinations are more consistent and accurate for all cases. Second, because Gaussian white noise introduces uniform phase mixing, techniques that rely on phase information (e.g., zero-crossing measurements in the FMTOA algorithm) are at a disadvantage relative to wideband amplitude measurements, provided a reasonable SNR is achieved. Indeed, the leading-edge technique performs better with the wideband receiver than the FMTOA algorithm in these examples.

The output of the cross-correlation analysis includes both a DTOA and a TEC estimate that is derived from equation (16). The results are shown in Figures 30a–c. This technique yields DTOAs that are accurate to within 15 ns in the presence of both noise and scattering and thus performs better than either the FMTOA or leading-edge algorithms for the cases considered. The TEC estimate is approximately 30% higher than the actual input value; however, the accuracy can be improved significantly by constructing a look-up table for TEC as a function of DTOA.

D. DTOA Study

The default delta function and two Gaussian filters with central frequencies of 150 MHz and 50 MHz and bandwidths of 6 MHz were used as input to the DTOA study. The leading-edge, time-tagging algorithm was invoked to obtain TOAs for each channel over a range of TECs from 0.1 to 10×10^{13} cm^{-2}. The results, shown in Figure 31, illustrate some of the limitations of TIPC. We see that the plot is characterized by two straight lines of different slope with the break occurring at a TEC of approximately 4.5×10^{13} cm^{-2}. Because the DTOA is proportional to TEC, ideally one expects to obtain a single straight line. The discrepancy results from the fact that the dispersion across the 50 MHz filter becomes comparable to the 10-μs window and significant aliasing occurs for TECs $\gtrsim 5 \times 10^{13}$ cm^{-2}. This result leads to erroneous values for the TOA of the 50 MHz channel. The user can bypass this problem by selecting a larger sampling time, thereby increasing the size of the window. The tradeoff is that the maximum frequency (Nyquist frequency) that can be treated is reduced. The user should take care to select an appropriate range of TECs.

This task will generate, for any specified set of receivers, a look-up table that allows the user to determine a TEC for any given measurement of DTOA. This analysis represents the first step in removing ionospheric effects from the received signal.

E. DTOA Uncertainty Study

The default delta function and the two Gaussian filters specified for the DTOA study were used as input to the DTOA uncertainty study. The TEC was fixed at 1×10^{13} cm^{-2} and Gaussian white noise was added to the input signal. The resulting filtered transionospheric signal was time-tagged with the leading-edge algorithm for signal-to-noise ratios ranging from 10^2 to 10^5 (see Eqs. (7), (8), and (9) for implementation of noise). A mean DTOA and the standard deviation from the mean were calculated using 25 realizations of each SNR. The results are shown in Figures 32a and b. Note the rapid reduction in performance below a SNR of 10^4 (see Figure 32b). A similar study was undertaken for coherence bandwidths ranging from 1 to 20 MHz. The results are shown in Figures 33a and b. The magnitude

of the DTOA uncertainties over the range considered are smaller by an order of magnitude than those obtained in the case of noise, suggesting that scattering of the EMP by ionospheric irregularities is less disruptive. Note the rapid deterioration in performance below a coherence bandwidth of 10 MHz. As F_{coh} approaches the filter bandwidth, amplitude variations are introduced to the leading-edge of the received signal, resulting in additional uncertainties in time-tagging.

VI. FUTURE ENHANCEMENTS

TIPC represents a powerful tool for exploring important aspects of transionospheric propagation, the detection of RF signals following propagation and signal analysis of pulse waveforms. The code was designed to be versatile enough both to address general questions regarding the effects of ionospheric propagation on received signals and to simulate actual receiver systems and their measurements of transionospheric signals. As such, TIPC is both an educational tool and an applications code. Future enhancements to TIPC will be in the area of input/output (the output is primarily in the form of graphics at present) and in the development of a reconstruction algorithm.

The output of future versions of TIPC will include text files with relevant results from each of the tasks and subtasks selected by the user. A log file containing a listing of the input parameters and a history of the execution of various tasks and associated results will be created. In addition, the user will be given the option of writing out the transionospheric signal-filtered transionospheric signals, results from quadrature detection and square-law detection, and tables of DTOA vs. TEC (obtained from the DTOA Study task) for specified filter pairs. The user will also be given the option of specifying a transionospheric signal or a set of filtered transionospheric signals in tabular form. In this way the signal-processing algorithms developed for TIPC can be applied to measurements and, for example, a TEC can be estimated with the DTOA algorithms for the specified data. Finally, a Signal Reconstruction task will be created. This task will have two options: Inverse Filter and Correlation Analysis. The first option will create an inverse filter for the ionosphere based on a TEC estimate obtained from the Signal Processing task, create the inverse filters for the specified set of receivers, and apply these filters to the specified filtered transionospheric signal to obtain an estimate of the initial signal. The second option will perform a cross-correlation between a set of calculated or specified filtered transionospheric signals and a corresponding set (read in from a specified text file) obtained for a known input signal. Depending on the amplitude of the cross-correlation, the user can determine whether or not the known input signal is a viable candidate for the specified measurements.

REFERENCES

Aarons, J. H., E. Whitney, and R. S. Allen, 1971, *Proc. IEEE*, **59**, 159.

Bahar, E., and B. S. Argrawal, 1976, *Radio Sci.*, **11**, No. 11, 885.

Barabanenkov, Yu. N., Yu. A. Kravtsov, S. M. Rytov, and V. I. Tamarskii, 1971, *Sov. Phys. Usjrekhi*, **13**, No. 5, 5510.

Basu, S., S. Basu, and K. Kahn, 1976, *Radio Sci.*, **11**, No. 10, 821.

Basu, S., and M. C. Kelley, 1979, *Radio Sci.*, **14**, 471.

Basu, S., S. Basu, J. P. Mullen, and A. Birshby, 1980, *Geophys. Res. Lett.*, **1**, No. 4, 259.

Basu, S., E. MacKenzie, E., S. Basu, H. O. Carlson, D. A. Hardy, R. J. Rich, and R. C. Livingston, 1983, *Radio Sci.*, **18**, No. 6, 1151.

Bilitza, D., 1986, "IRI: Recent Developments," *Radio Sci.*, **21**, 343.

Bogusch, R. L., F. W. Crugliano, D. L. Knepp, and A. H. Michelet, 1981, *Proc. IEEE*, **69**, No. 7, 787.

Booker, H. G., J. A. Ratcliffe, and D. H. Sherin, 1950, *Proc. Phys. Soc.*, London, **242**, A856, 75.

Budden, K. G., 1965, *J. Atmos. Terr. Phys.*, **27**, 155.

Costa, E., and M. C. Kelley, 1976, *Geophys. Res. Lett.*, **3**, No. 11, 677.

Dana, R. A., 1988, "ACIRF User's Guide," **Volume 1**: "Theory and Examples," DNA001-87-C-0169, MRC-R-1198, Mission Research Corporation.

Davies, K., 1965, *Ionspheric Radio Propagation*, National Bureau Standards Monograph 80 (U.S. Government Printing Office, Washington, DC), 470 pp.

Dyson, P. R., J. P. McClure, and W. B. Hanson, 1974, *J. Geophys. Res.*, **79**, 1497.

Fejer, B. G. and M. C. Kelley, "Ionospheric Irregularities," 1980, *Rev. Geophys.*, **18**, 401.

Fremouw, E. J., R. L. Leadabrand, R. C. Livingston, M. D. Cousins, C. L. Rino, B. C. Fir, and R. A. Long, 1978, *Radio Sci.*, **13**, No. 9. 1, 167.

Fremouw, E. J., R. C. Livingston, and D. A. Miller, 1980, *J. Atmos. Terr. Phys.*, **42**, 717.

Johnson, A., 1981, Air Force Geophysics Laboratory Technical Memo No. 56, **V-1**.

Kelley, M. C., J. LaBelle, E. Kudeki, B. G. Fejer, S. Basu, S. Basu, K. D. Baker, C. Hanuise, P. Argo, R. F. Woodman, W. E. Swartz, and D. T. Farley, 1985, preprint.

Knepp, D. L., 1982, Defense Nuclear Agency report DNA-TR-81-78.

Knepp, D. L., and R. L. Bogusch, 1979, Mission Research Corporation report DNA 5038T, MRC-R-514.

Livingston, R. C., 1980, *Radio Sci.*, **15**, No. 4, 801.

Lovelace, R. V. E., E. E. Salpeter, L. E. Sharp, and D. E. Harris, 1970, *Ap. J.*, **159**, 1047.

Martin, E., and J. Aarons, 1977, *J. Geophys. Res.*, **82**, No. 19, 2713.

Phelps, A. R., and R. C. Sagalyn, 1976, *J. Geophys. Res.*, **81**, No. 4, 515.

Rawer, K., D. Bilitza and S. Ramakrishnan, 1978a, *Rev. of Geophys. and Spa. Phys.*, **16**, No. 2, 177.

Rawer, K. and P. A. Bradley, 1987, editors of "International Reference Ionosphere Status 1986/87," *Advances in Spa. Res.*, **7**, No. 6, 129 pages.

Rawer, K., S. Ramakrishnan, and D. Bilitza, 1978b, "IRI 1978," International Union of Radio Science (URSI), Brussels, Belgium.

Rino, C. L., and J. Owen, 1980, *Radio Sci.*, **15**, 479.

Rino, C. L., M. D. Cousins, and N. B. Walker, 1981a, Air Force Geophysics Laboratory Technical Memo No. 56, IV-1.

Rino, C. L., A. T. Tsunoda, J. Petriceks, R. C. Livingston, M. C. Kelley, and K. D. Baker, 1981b, *J. Geophys. Res.*, **86**, 2411.

Rishbeth, H., and O. K. Garriott, 1969, *Introduction to Ionospheric Physics*, in *International Geophysics Series*, **14**, J. Van Mieghern and Austin L. Hales, Eds. (Academic Press, New York, 1969).

Rufenach, C. L., 1975a, *Radio Sci.*, **10**, No. 2, 155.

Rufenach, C. L., 1975b, *Radio Sci.*, **10**, No. 11, 973.

Salpeter, E. E., 1967, *Ap. J.*, **147**, No. 2, 433.

Tatarskii, V. I., 1971, "The Effects of the Turbulent Atmosphere on Wave Propagation," U.S. Department of Commerce, National Technical Information Service, TT68-50464, 472.

Umeki, R., C. H. Liu, and K. C. Yeh, 1977, *J. Geophys. Res.*, **82**, No. 19, 2752.

Wittwer, L. A., 1979, Defense Nuclear Agency report DNA-5304D.

Wittwer, L. A., 1980, Defense Nuclear Agency report DNA-5662D.

Whitney, H. E., J. A. Klobuchar, J. Lloyd, and J. Austin, 1981, Air Force Geophysics Laboratory Technical Memo No. 56, III-I.

APPENDIX A. Flow Chart Tree

```
TIPC
.   CLEAR
.   FREIL
.   GETTASK
.   .   CLEAR
.   .   IOGETI  (OINK library)
.   .   IONMEN  (OINK library)
.   .   MENUDT
.   .   .   CLEAR
.   .   .   GETTEXT
.   .   .   INITVEC
.   .   .   IOGETI  (OINK library)
.   .   .   IOGETR  (OINK library)
.   .   .   IOYORN  (OINK library)
.   .   .   PAUSE
.   .   .   .   IOGETC  (OINK library)
.   .   MENUDU
.   .   .   CLEAR
.   .   .   GETTEXT
.   .   .   INITVEC
.   .   .   IOGETI  (OINK library)
.   .   .   IOGETR  (OINK library)
.   .   .   IOYORN  (OINK library)
.   .   .   PAUSE
.   .   .   .   IOGETC  (OINK library)
.   .   MENUPS
.   .   .   CLEAR
.   .   .   GETTEXT
.   .   .   IOGETR  (OINK library)
.   .   .   IONMEN  (OINK library)
.   .   .   IOYORN  (OINK library)
.   .   .   PAUSE
.   .   .   .   IOGETC  (OINK library)
.   .   MENUSD
.   .   .   CLEAR
.   .   .   GETTEXT
.   .   .   IOGETI  (OINK library)
.   .   .   IOGETR  (OINK library)
.   .   .   IOGFIL  (OINK library)
.   .   .   IOYORN  (OINK library)
.   .   .   PAUSE
```

```
. . . . IOGETC  (OINK library)
. . MENUTP
. . . CLEAR
. . . FRANGE
. . . . CLEAR
. . . . IOGETR  (OINK library)
. . . . IOYORN  (OINK library)
. . . GETTEXT
. . . INITTF
. . . IOGETR  (OINK library)
. . . IOGFIL  (OINK library)
. . . IONMEN  (OINK library)
. . . IOYORN  (OINK library)
. . . PAUSE
. . . . IOGETC  (OINK library)
. . . PULSE
. . . . CLEAR
. . . . IOGETI  (OINK library)
. . . . IOGETR  (OINK library)
. . . . IOGFIL  (OINK library)
. . . . IONMEN  (OINK library)
. . . . IOYORN  (OINK library)
. . . SETtec
. . PARAMS
. . . IOOFIL  (OINK library)
. . . RESTORE
. . . . SETtec
. . . SAVEC
. . . SAVEI
. . . SAVEL
. . . SAVER
. . PAUSE
. . . IOGETC  (OINK library)
. GETTEXT
. INITVAL
. . CFFTI  (FFT.FOR)
. . INITTF
. . INITVEC
. . NEWSEED
. . . SYSGETTIM
. . SETtec
. IOINIT  (OINK library)
. IONMEN  (OINK library)
. PLOTEXIT  (NCAR.FOR)
. PLOTINIT  (NCAR.FOR)
```

```
TASKDT
    CLEAR
    FILTERS
        AVGTIME
        FFT
            CFFTF   (FFT.FOR)
        FREQPLOT
            PLOTXY   (NCAR.FOR)
        FRFIL
        GETRCVR
            CFFTB   (FFT.FOR)
            PARSEMSG
                IOFIRS   (OINK library)
                IOLAST   (OINK library)
            SORT
        TIMEPLOT
            PLOTXY   (NCAR.FOR)
        TSHIFT
            CFFTB   (FFT.FOR)
            CFFTF   (FFT.FOR)
        WINDOW
            AMAXMIN
    FRFIL
    IPROP
        CIRF
            CFFTF   (FFT.FOR)
            POLY
            RANDG
        FRFIL
        TIMEPLOT
            PLOTXY   (NCAR.FOR)
        TSHIFT
            CFFTB   (FFT.FOR)
            CFFTF   (FFT.FOR)
        WINDOW
            AMAXMIN
    NOISE
        FREQPLOT
            PLOTXY   (NCAR.FOR)
        NOISE1
            AMAXMIN
            RANDG
        NOISE2
            GETNOISE
                PARSEMSG
```

```
. . . . . SORT
. . . . GETNP
. . . . RANDG
. . . . RANDU
. . . TIMEPLOT
. . . . . PLOTXY   (NCAR.FOR)
. . . TSHIFT
. . . . . CFFTB   (FFT.FOR)
. . . . . CFFTF   (FFT.FOR)
. . PLOTXY
. . SATDET
. . . AMAXMIN
. . . FREQPLOT
. . . . . PLOTXY   (NCAR.FOR)
. . . TIMEPLOT
. . . . . PLOTXY   (NCAR.FOR)
. . . TSHIFT
. . . . . CFFTB   (FFT.FOR)
. . . . . CFFTF   (FFT.FOR)
. . . WINDOW
. . . . . AMAXMIN
. . SETtec
. . SIGNAL
. . . AVGTIME
. . . FFT
. . . . . CFFTF   (FFT.FOR)
. . . FREQPLOT
. . . . . PLOTXY   (NCAR.FOR)
. . . IPCPB1
. . . IPCPB2
. . . IPDELTA
. . . . . AMAXMIN
. . . IPEMP
. . . IPTABLE
. . . . . CFFTI   (FFT.FOR)
. . . . . PARSEMSG
. . . . . . IOFIRS   (OINK library)
. . . . . . IOLAST   (OINK library)
. . . . . SORT
. . . TIMEPLOT
. . . . . PLOTXY   (NCAR.FOR)
. . . TSHIFT
. . . . . CFFTB   (FFT.FOR)
. . . . . CFFTF   (FFT.FOR)
. . . WINDOW
```

```
. . . . AMAXMIN
. . TASKPS
. . . CROSCOR
. . . . CROSINFO
. . . . . AMAXMIN
. . . . TIMEPLOT
. . . . . PLOTXY (NCAR.FOR)
. . . . TSHIFT
. . . . . CFFTB (FFT.FOR)
. . . . . CFFTF (FFT.FOR)
. . . . WINDOW
. . . . . AMAXMIN
. . . FMTOA
. . . . AMAXMIN
. . . LETOA
. . . . AMAXMIN
. . . QUAD
. . . . TIMEPLOT
. . . . . PLOTXY (NCAR.FOR)
. . . . TSHIFT
. . . . . CFFTB (FFT.FOR)
. . . . . CFFTF (FFT.FOR)
. . . . WINDOW
. . . . . AMAXMIN
. . . SQRLAW
. . . . CFFTB (FFT.FOR)
. . . . CFFTF (FFT.FOR)
. . . . FREIL
. . . . TSHIFT
. . . . . CFFTB (FFT.FOR)
. . . . . CFFTF (FFT.FOR)
. . . TIMEPLOT
. . . . PLOTXY (NCAR.FOR)
. . . WINDOW
. . . . AMAXMIN
. TASKDU
. . FILTERS
. . . AVGTIME
. . . FFT
. . . . CFFTF (FFT.FOR)
. . . FREQPLOT
. . . . PLOTXY (NCAR.FOR)
. . . FREIL
. . . GETRCVR
. . . . CFFTB (FFT.FOR)
```

```
. . . . PARSEMSG
. . . . . IOFIRS  (OINK library)
. . . . . IOLAST  (OINK library)
. . . . SORT
. . . TIMEPLOT
. . . . PLOTXY  (NCAR.FOR)
. . . TSHIFT
. . . . CFFTB  (FFT.FOR)
. . . . CFFTF  (FFT.FOR)
. . . WINDOW
. . . . AMAXMIN
. . IPROP
. . . CIRF
. . . . CFFTF  (FFT.FOR)
. . . . POLY
. . . . RANDG
. . . FRFIL
. . . TIMEPLOT
. . . . PLOTXY  (NCAR.FOR)
. . . TSHIFT
. . . . CFFTB  (FFT.FOR)
. . . . CFFTF  (FFT.FOR)
. . . WINDOW
. . . . AMAXMIN
. . NOISE
. . . FREQPLOT
. . . . PLOTXY  (NCAR.FOR)
. . . NOISE1
. . . . AMAXMIN
. . . . RANDG
. . . NOISE2
. . . . GETNOISE
. . . . . PARSEMSG
. . . . . IOFIRS  (OINK library)
. . . . . IOLAST  (OINK library)
. . . . . SORT
. . . . GETNP
. . . . RANDG
. . . . RANDU
. . . TIMEPLOT
. . . . PLOTXY  (NCAR.FOR)
. . . TSHIFT
. . . . CFFTB  (FFT.FOR)
. . . . CFFTF  (FFT.FOR)
. . PLOTXY
```

```
. . SATDET
. . . AMAXMIN
. . . FREQPLOT
. . . . PLOTXY (NCAR.FOR)
. . . TIMEPLOT
. . . . PLOTXY (NCAR.FOR)
. . . TSHIFT
. . . . CFFTB (FFT.FOR)
. . . . CFFTF (FFT.FOR)
. . . WINDOW
. . . . AMAXMIN
. . SIGNAL
. . . AVGTIME
. . . FFT
. . . . CFFTF (FFT.FOR)
. . . FREQPLOT
. . . . PLOTXY (NCAR.FOR)
. . . IPCPB1
. . . IPCPB2
. . . IPDELTA
. . . . AMAXMIN
. . . IPEMP
. . . IPTABLE
. . . . CFFTI
. . . . PARSEMSG
. . . . . IOFIRS (OINK library)
. . . . . IOLAST (OINK library)
. . . . SORT
. . . TIMEPLOT
. . . . PLOTXY (NCAR.FOR)
. . . TSHIFT
. . . . CFFTB (FFT.FOR)
. . . . CFFTF (FFT.FOR)
. . . WINDOW
. . . . AMAXMIN
. . TASKPS
. . . CROSCOR
. . . . CROSINFO
. . . . . AMAXMIN
. . . . TIMEPLOT
. . . . . PLOTXY (NCAR.FOR)
. . . . TSHIFT
. . . . . CFFTB
. . . . . CFFTF
. . . . WINDOW
```

```
. . . . . AMAXMIN
. . . FMTOA
. . . . AMAXMIN
. . . LETOA
. . . . AMAXMIN
. . . QUAD
. . . . TIMEPLOT
. . . . . PLOTXY (NCAR.FOR)
. . . . TSHIFT
. . . . . CFFTB (FFT.FOR)
. . . . . CFFTF (FFT.FOR)
. . . . WINDOW
. . . . . AMAXMIN
. . . SQRLAW
. . . . CFFTB (FFT.FOR)
. . . . CFFTF (FFT.FOR)
. . . . FRFIL
. . . . TSHIFT
. . . . . CFFTB (FFT.FOR)
. . . . . CFFTF (FFT.FOR)
. . . TIMEPLOT
. . . . PLOTXY (NCAR.FOR)
. . . WINDOW
. . . . AMAXMIN
. TASKPS
. . CROSCOR
. . . CROSINFO
. . . . AMAXMIN
. . . TIMEPLOT
. . . . PLOTXY (NCAR.FOR)
. . . TSHIFT
. . . . CFFTB (FFT.FOR)
. . . . CFFTF (FFT.FOR)
. . . WINDOW
. . . . AMAXMIN
. . FMTOA
. . . AMAXMIN
. . LETOA
. . . AMAXMIN
. . QUAD
. . . TIMEPLOT
. . . . PLOTXY (NCAR.FOR)
. . . TSHIFT
. . . . CFFTB (FFT.FOR)
. . . . CFFTF (FFT.FOR)
```

```
. . . WINDOW
. . . . AMAXMIN
. . SQRLAW
. . . CFFTB (FFT.FOR)
. . . CFFTF (FFT.FOR)
. . . FRFIL
. . . SHIFT
. . . . CFFTB (FFT.FOR)
. . . . CFFTF (FFT.FOR)
. . . TIMEPLOT
. . . . PLOTXY (NCAR.FOR)
. . . WINDOW
. . . . AMAXMIN
. . TASKSD
. . . AMAXMIN
. . . FILTERS
. . . . AVGTIME
. . . . FFT
. . . . CFFTF (FFT.FOR)
. . . FREQPLOT
. . . . PLOTXY (NCAR.FOR)
. . . FRFIL
. . . GETRCVR
. . . . CFFTB (FFT.FOR)
. . . . PARSEMSG
. . . . IOFIRS (OINK library)
. . . . IOLAST (OINK library)
. . . SORT
. . TIMEPLOT
. . . PLOTXY (NCAR.FOR)
. . TSHIFT
. . . CFFTB (FFT.FOR)
. . . CFFTF (FFT.FOR)
. . WINDOW
. . . AMAXMIN
. FREQPLOT
. . PLOTXY (NCAR.FOR)
. SATDET
. . AMAXMIN
. . FREQPLOT
. . . PLOTXY (NCAR.FOR)
. . TIMEPLOT
. . . PLOTXY (NCAR.FOR)
. . TSHIFT
. . . CFFTB (FFT.FOR)
```

```
. . . CFFTF (FFT.FOR)
. . WINDOW
. . . AMAXMIN
TASKTP
. IPROP
. . CIRF
. . . CFFTF (FFT.FOR)
. . . POLY
. . . RANDG
. . FRFIL
. . TIMEPLOT
. . . PLOTXY (NCAR.FOR)
. . TSHIFT
. . . CFFTB (FFT.FOR)
. . . CFFTF (FFT.FOR)
. . WINDOW
. . . AMAXMIN
. NOISE
. . FREQPLOT
. . . PLOTXY (NCAR.FOR)
. . NOISE1
. . . AMAXMIN
. . . RANDG
. . NOISE2
. . . GETNOISE
. . . . PARSEMSG
. . . . SORT
. . . GETNP
. . . RANDG
. . . RANDU
. . TIMEPLOT
. . . PLOTXY (NCAR.FOR)
. . TSHIFT
. . . CFFTB (FFT.FOR)
. . . CFFTF (FFT.FOR)
. SIGNAL
. . AVGTIME
. . FFT
. . . CFFTF (FFT.FOR)
. . FREQPLOT
. . . PLOTXY (NCAR.FOR)
. . IPCPB1
. . IPCPB2
. . IPDELTA
. . . AMAXMIN
```

```
. . IPEMP
. . IPTABLE
. . . CFFTI
. . . PARSEMSG
. . . . IOFIRS (OINK library)
. . . . IOLAST (OINK library)
. . . SORT
. . TIMEPLOT
. . . LOTXY (NCAR.FOR)
. . TSHIFT
. . . CFFTB (FFT.FOR)
. . . CFFTF (FFT.FOR)
. . WINDOW
. . . AMAXMIN
```

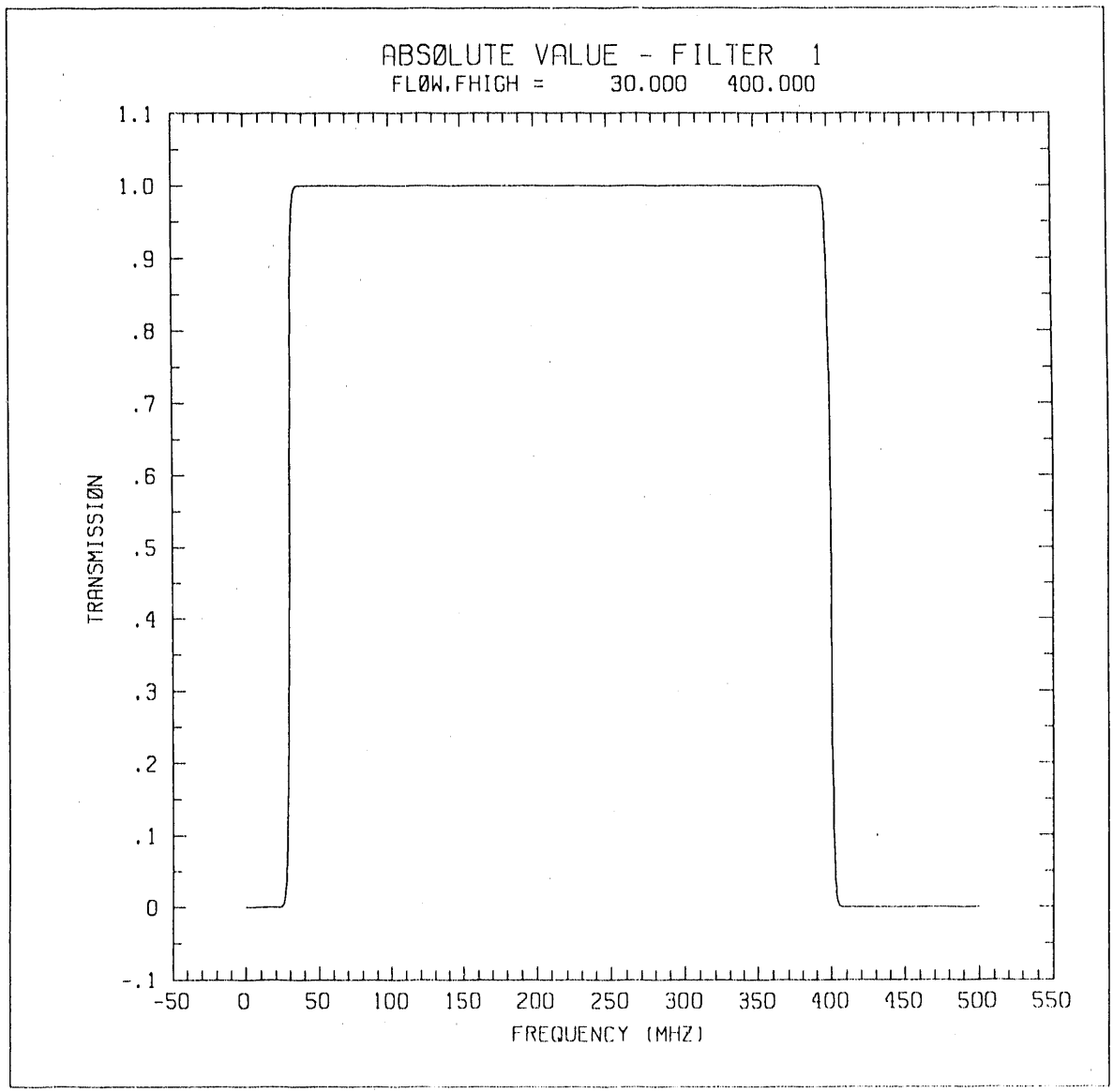

Figure 1 The filter shape used to limit the frequency range in the time domain to fit in the 10-μs window. This functional form preserves the signal amplitude through most of the time domain and minimizes ringing near the edges of the window.

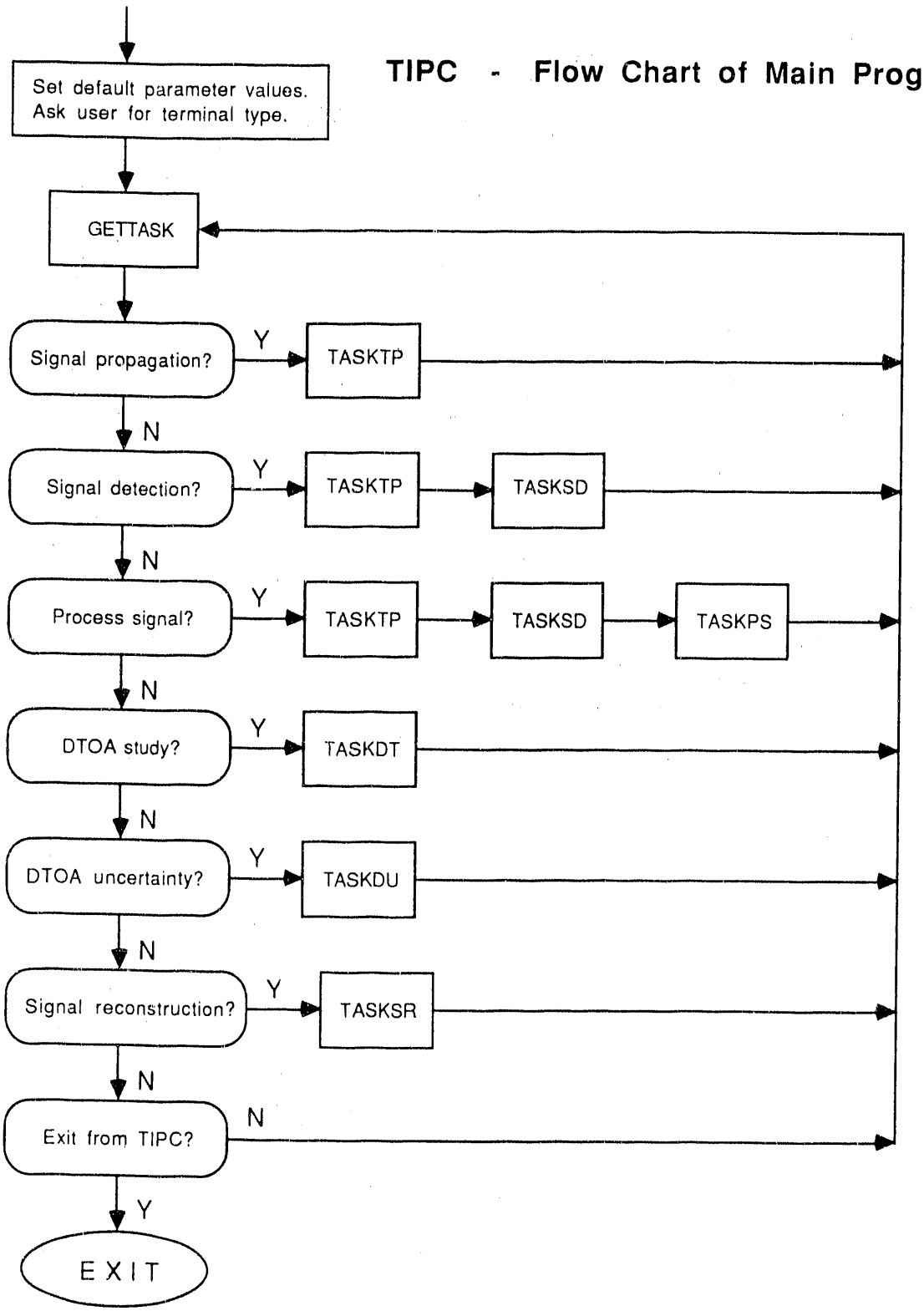

Figure 2 Flow diagram for TIPC.

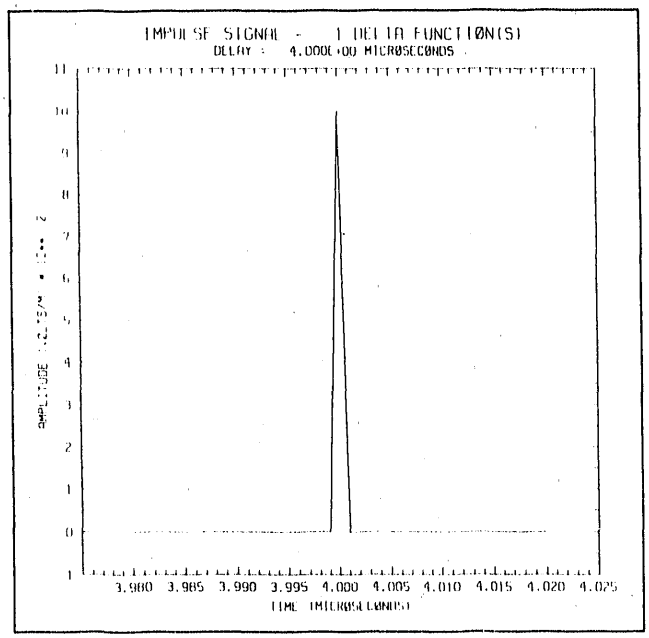

Figure 3a

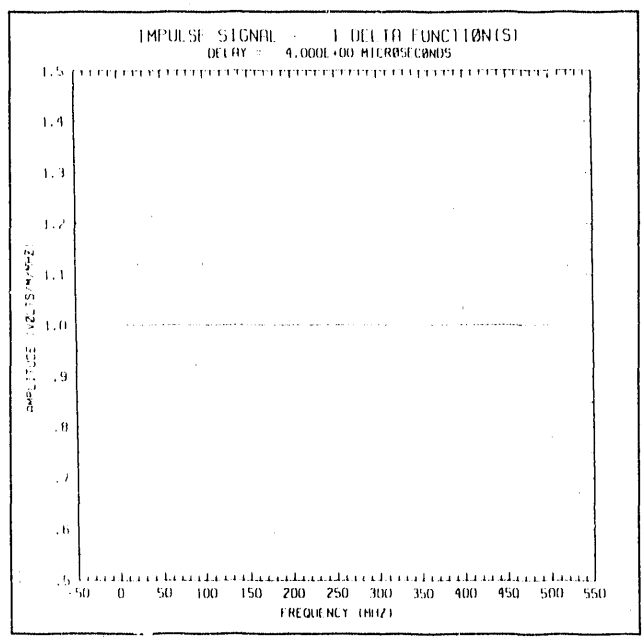

Figure 3b

Figures 3a–j (a) Temporal plot of the default delta function. (b) FFT of the default delta function. (c)–(f) Corresponding transionospheric signals for a TEC of 1×10^{13} cm^{-2}, an SNR limited by numerical noise (70 dB), and $F_{coh} = \infty$, 20, 10, and 1 MHz, respectively. (g)–(h) Temporal plot of the noisy default delta function with an SNR = 100 and the corresponding transionospheric signal for a TEC of 1×10^{13} cm^{-2}, $F_{coh} = \infty$, and a SNR = 100, respectively. (i)–(j) Temporal plot of the noisy default delta function with an SNR = 10,000 and the corresponding transionospheric signal for a TEC of 1×10^{13} cm^{-2}, $F_{coh} = \infty$, and an SNR = 10,000, respectively.

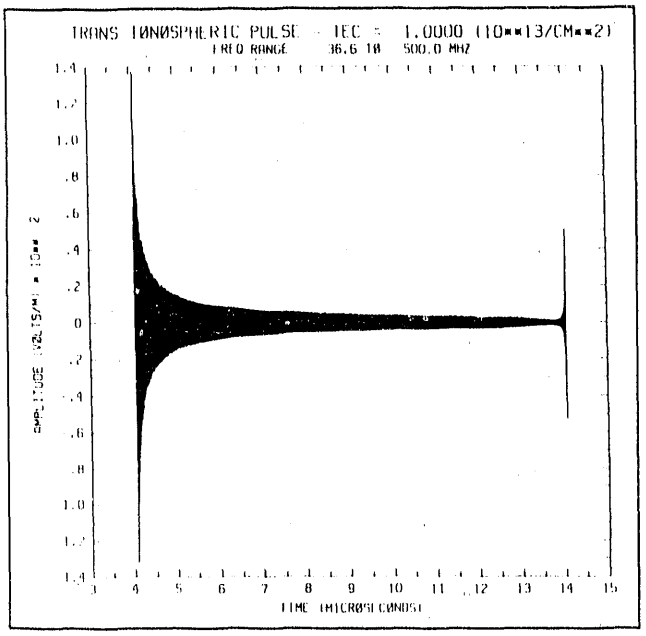

Figure 3c

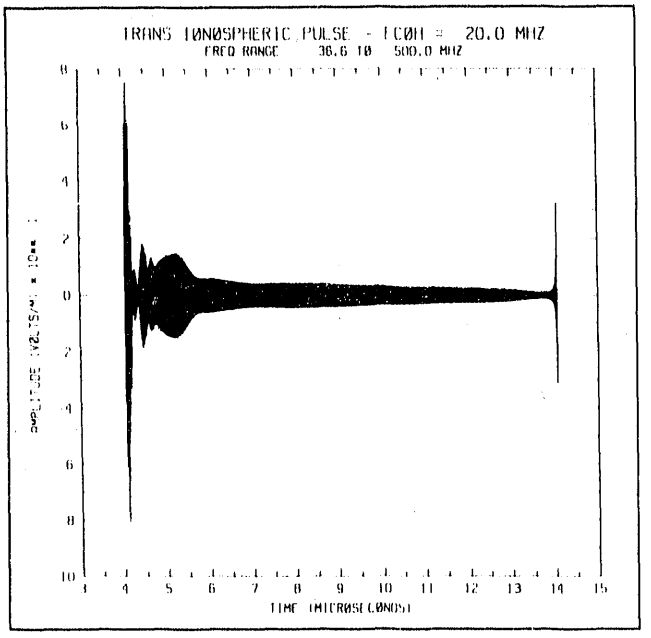

Figure 3d

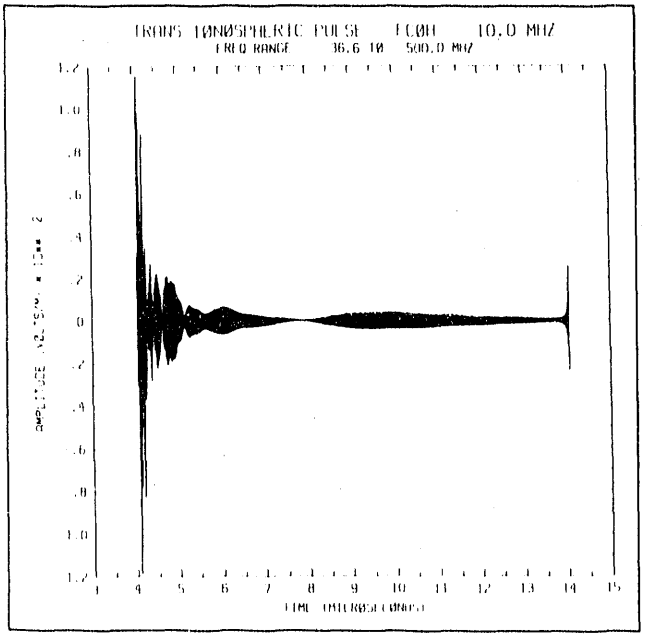

Figure 3e

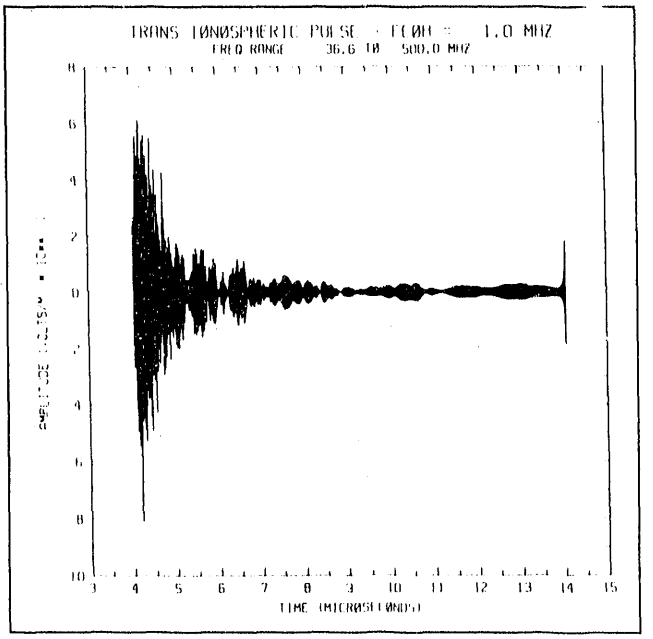

Figure 3f

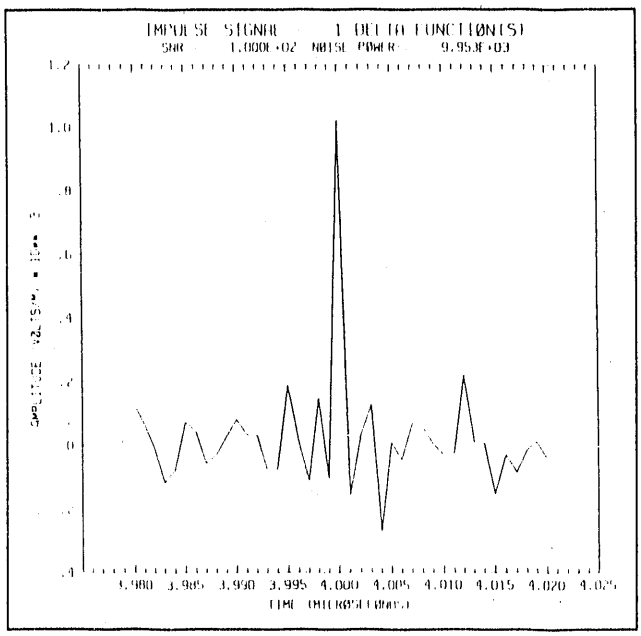

Figure 3g

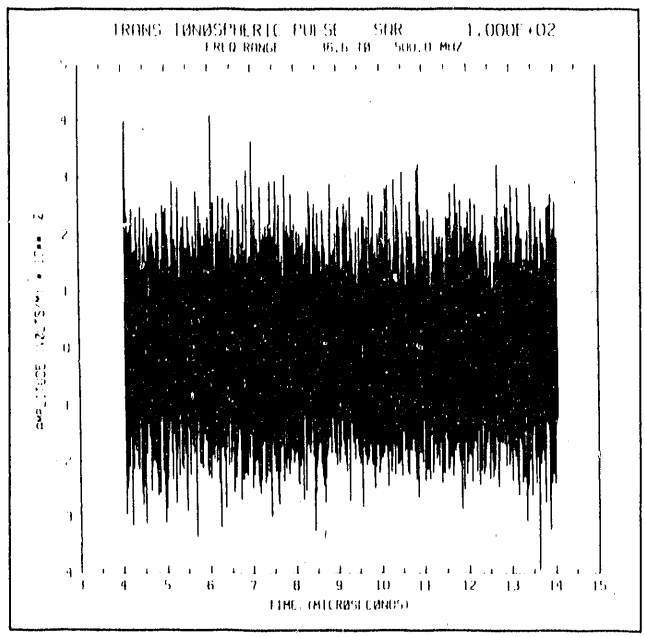

Figure 3h

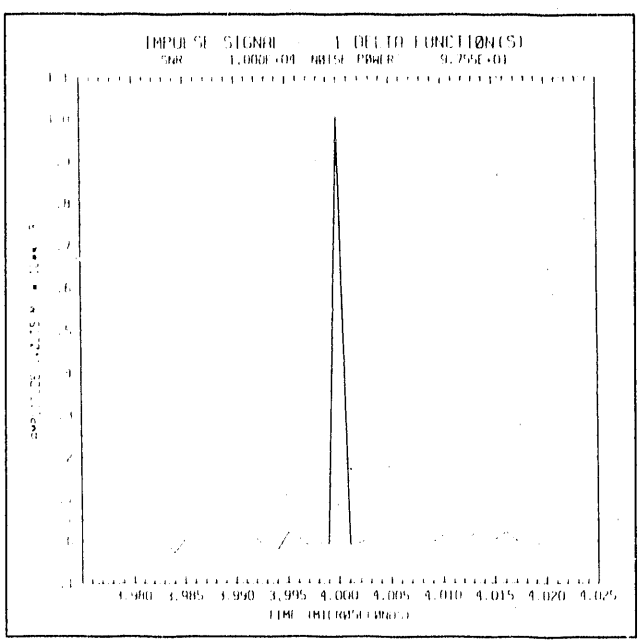

Figure 3i

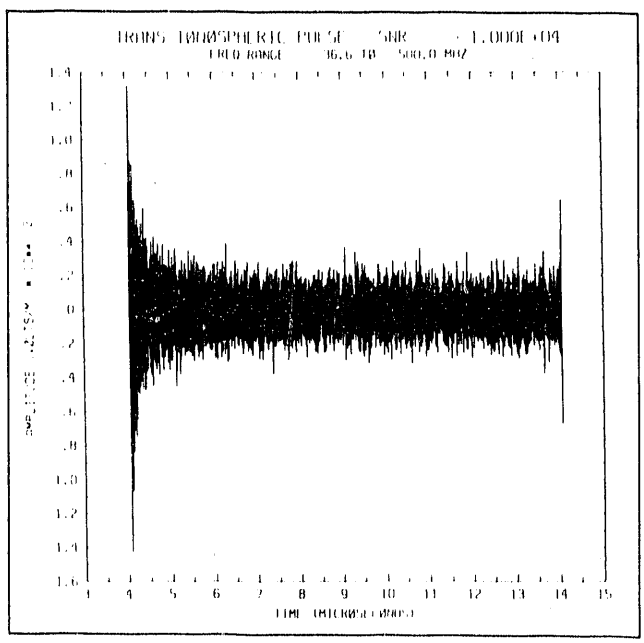

Figure 3j

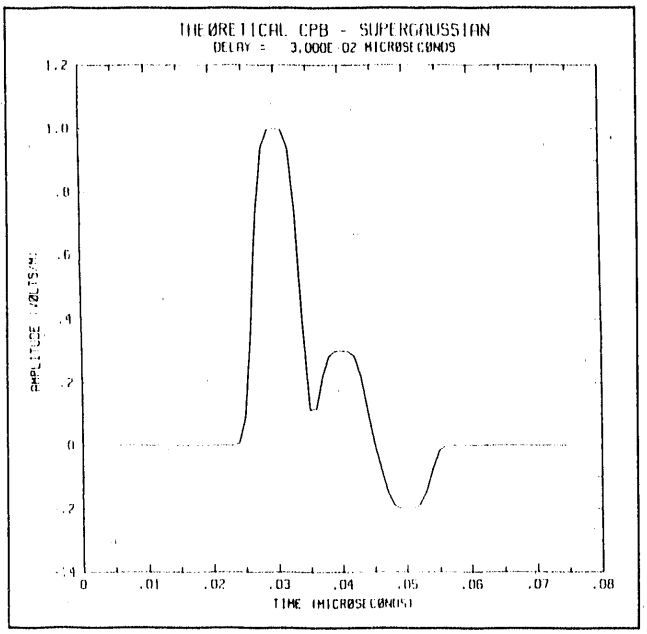

Figure 4a

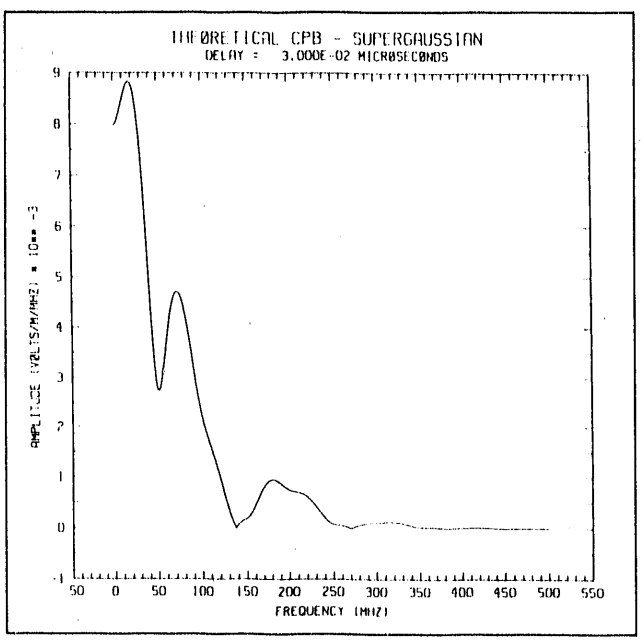

Figure 4b

Figures 4a–j (a) Temporal plot of the default super-Gaussian pulse. (b) FFT of the default super-Gaussian pulse. (c)–(f) Corresponding transionospheric signals for an TEC of 1×10^{13} cm^{-2}, an SNR limited by numerical noise (70 dB), and $F_{coh} = \infty$, 20, 10, and 1 MHz, respectively. (g)–(h) Temporal plot of the noisy default delta function with an SNR = 100 and the corresponding transionospheric signal for a TEC of 1×10^{13} cm^{-2}, $F_{coh} = \infty$, and an SNR = 100, respectively. (i)–(j) Temporal plot of the noisy default delta function with an SNR = 10,000 and the corresponding transionospheric signal for a TEC of 1×10^{13} cm^{-2}, $F_{coh} = \infty$, and an SNR = 10,000, respectively.

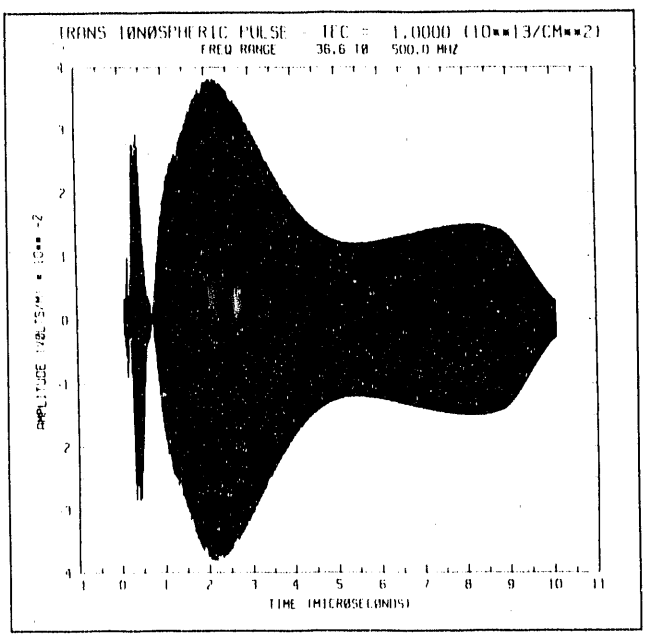

Figure 4c

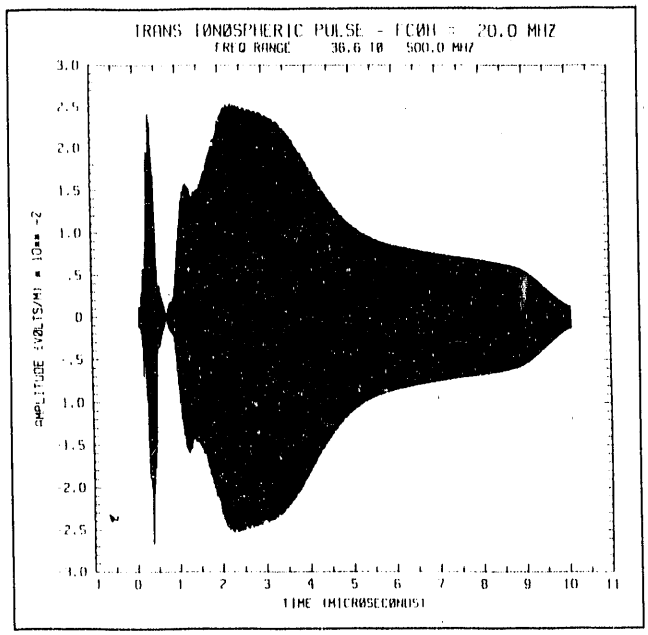

Figure 4d

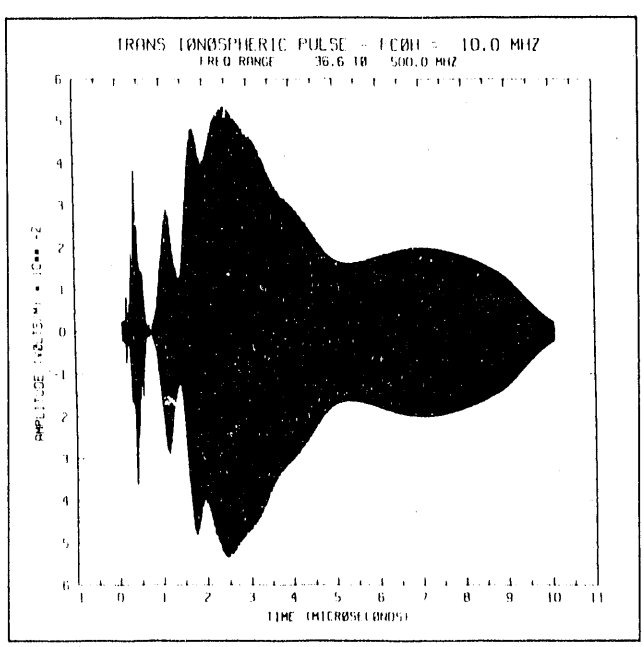

Figure 4e

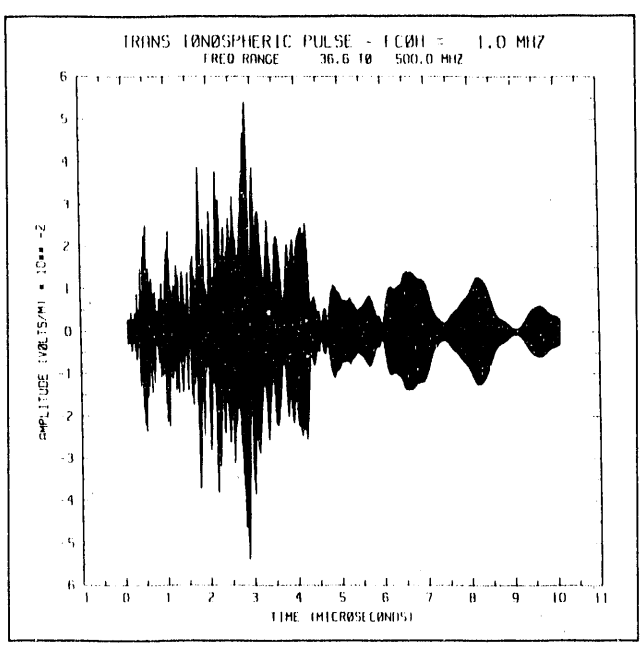

Figure 4f

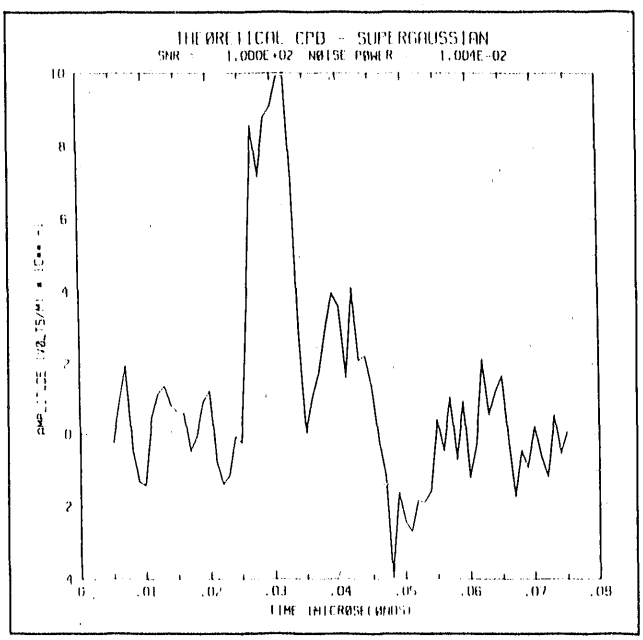

Figure 4g

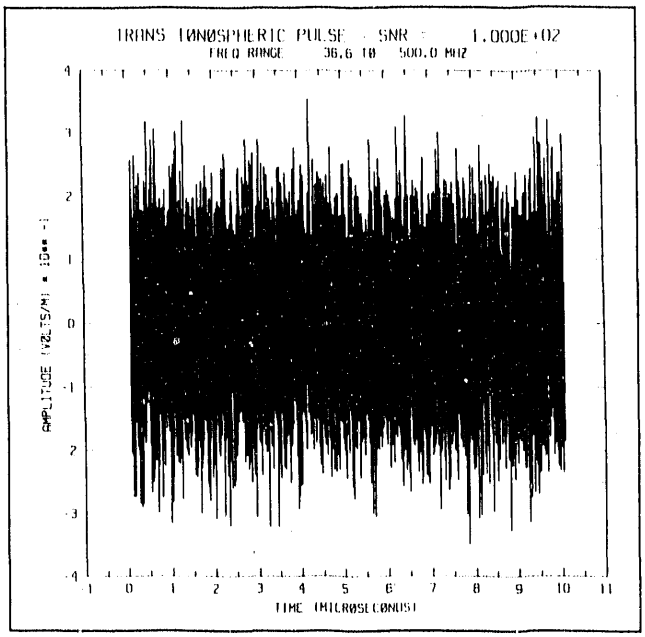

Figure 4h

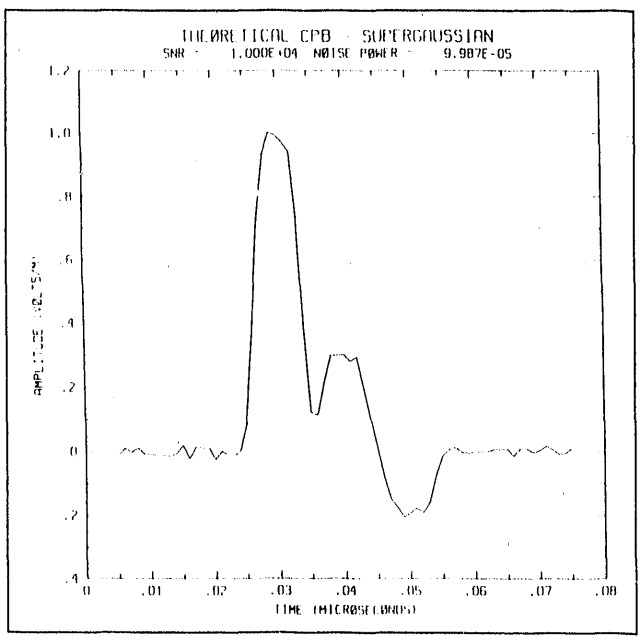

Figure 4i

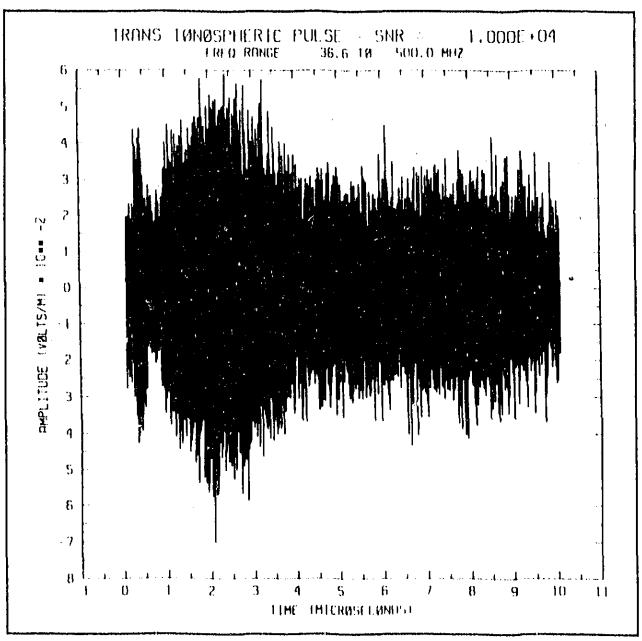

Figure 4j

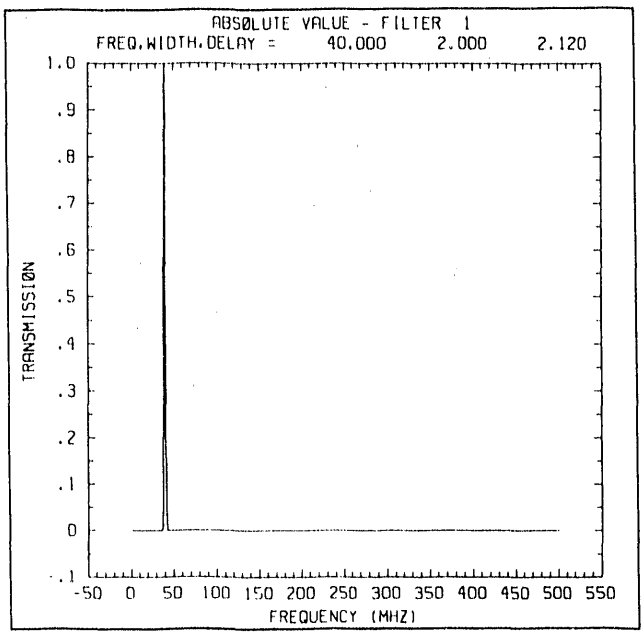

Figure 5a

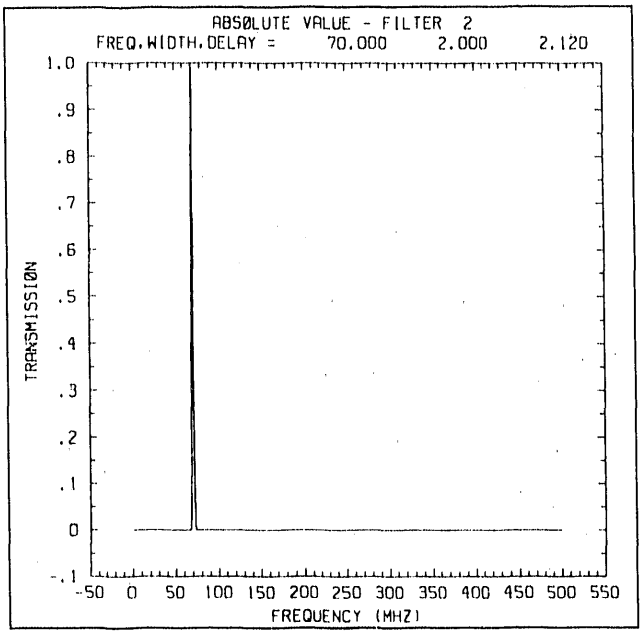

Figure 5b

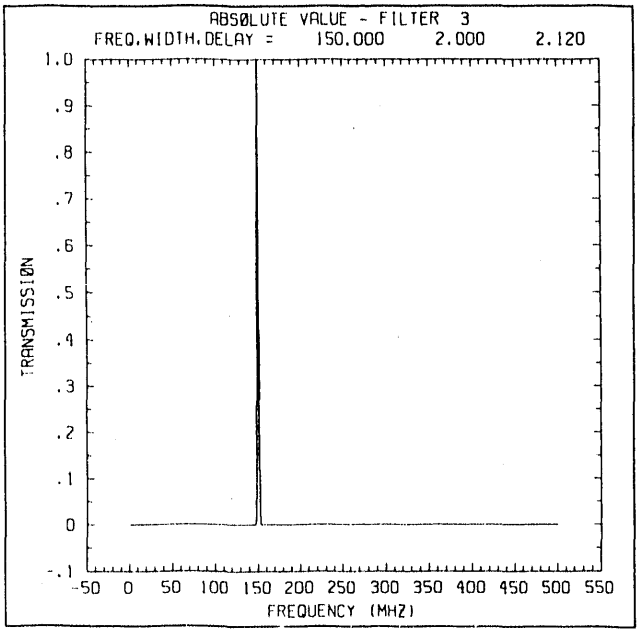

Figure 5c

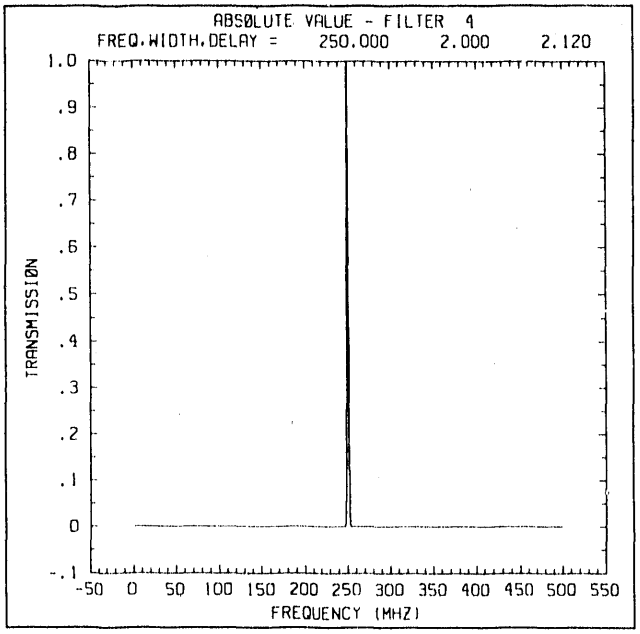

Figure 5d

Figures 5a–d (a)–(d) Transmission profiles for four Gaussian filters with 2-MHz bandwidths (FWHM) and central frequencies of 40, 70, 150, and 250 MHz, respectively.

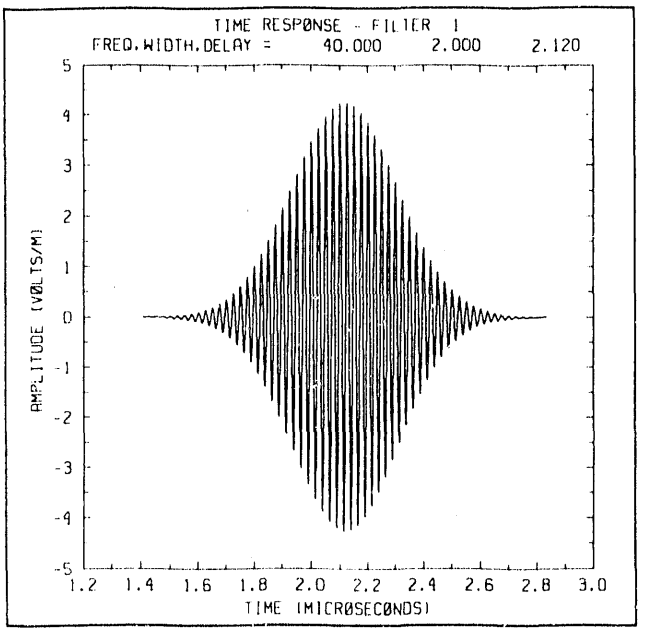

Figure 6a

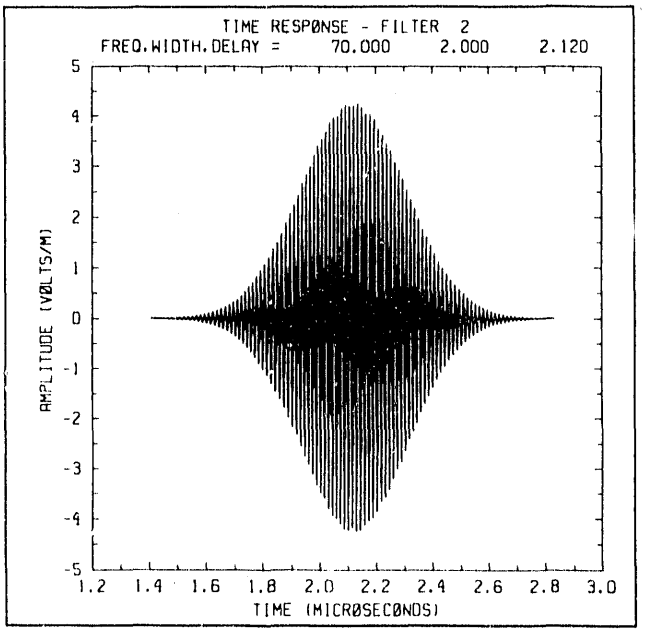

Figure 6b

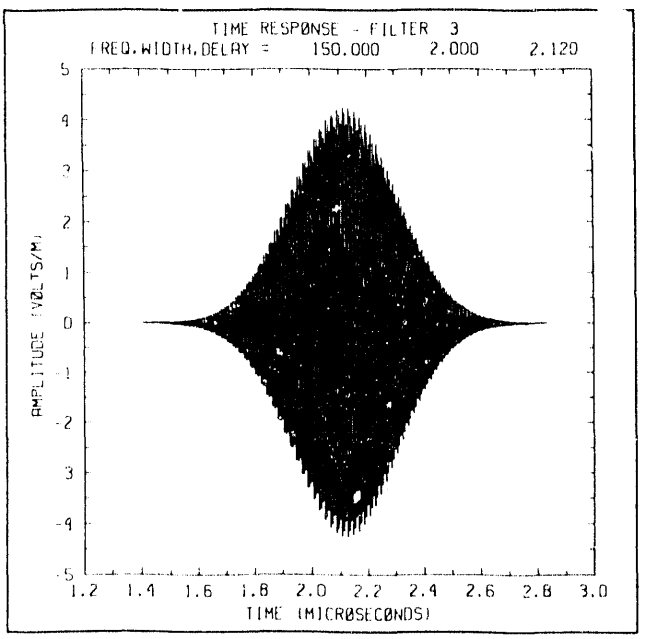

Figure 6c

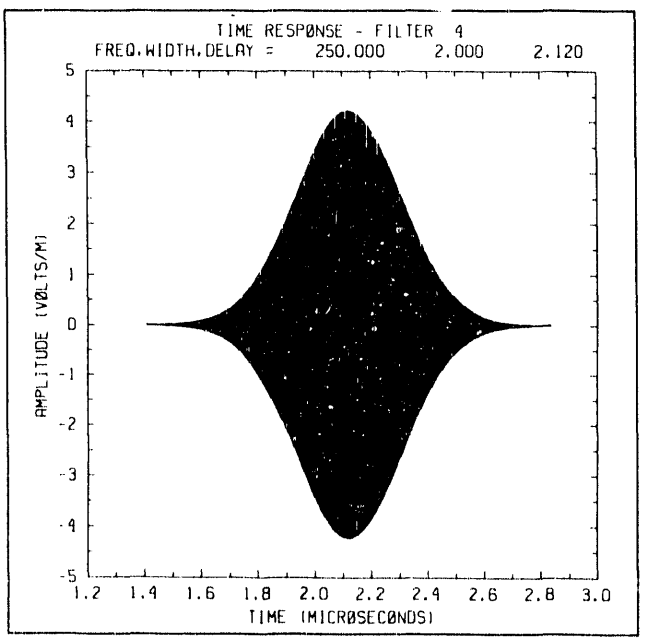

Figure 6d

Figures 6a–d (a)–(d) Impulse response functions for four Gaussian filters with 2-MHz bandwidths (FWHM) and central frequencies of 40, 70, 150, and 250 MHz, respectively. Each filter has a built in time delay equal to approximately 2.12 μs.

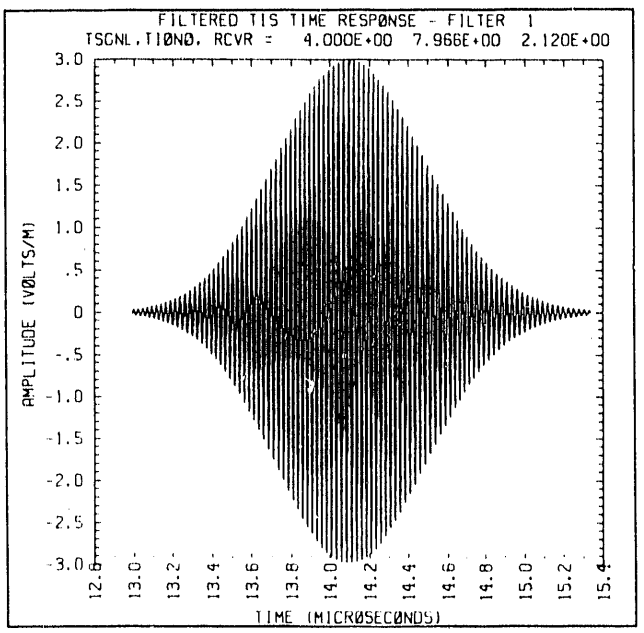

Figure 7a

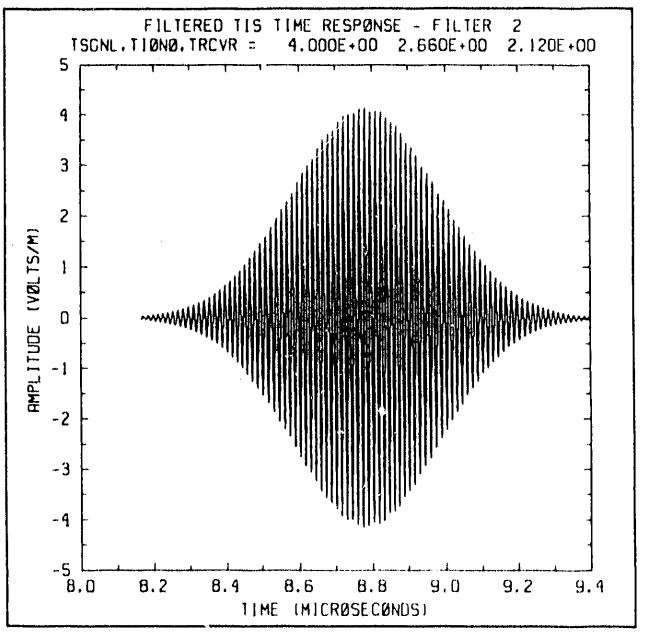

Figure 7b

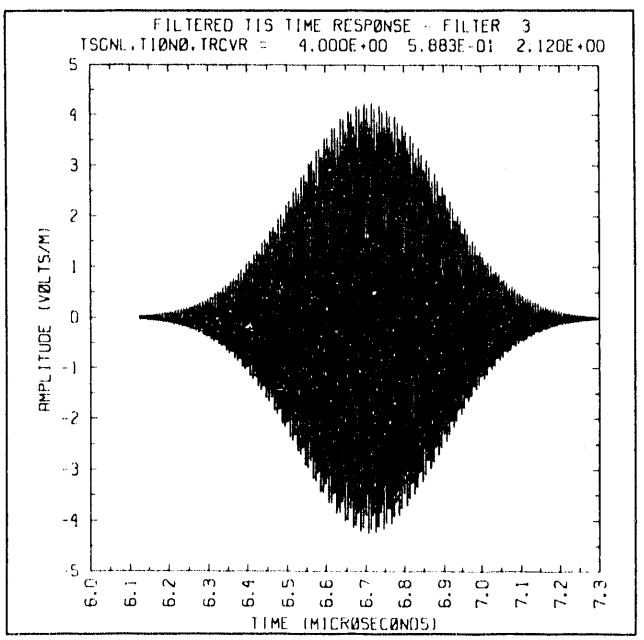

Figure 7c

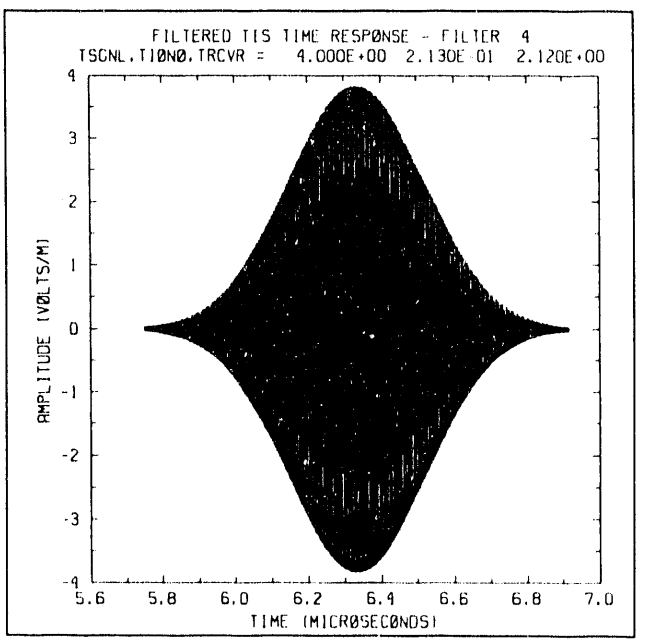

Figure 7d

Figures 7a–d (a)–(d) Filtered transionospheric signals obtained for the default delta function with the four Gaussian filters shown in Figures 5a–d, respectively. A TEC of 1×10^{13} cm^{-2}, $F_{coh} = \infty$, and an SNR of 70 dB were assumed.

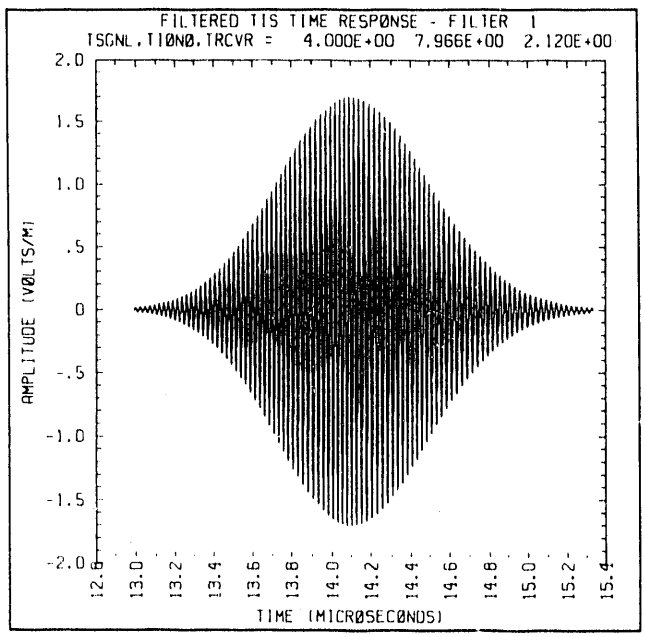

Figure 8a

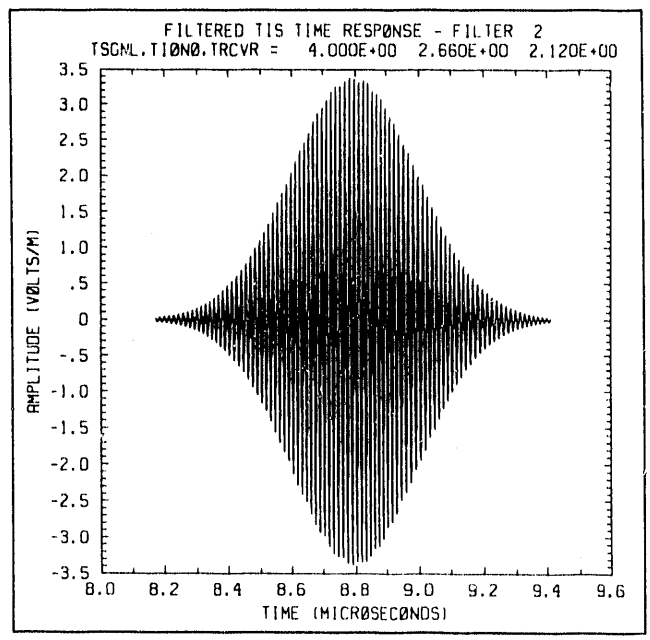

Figure 8b

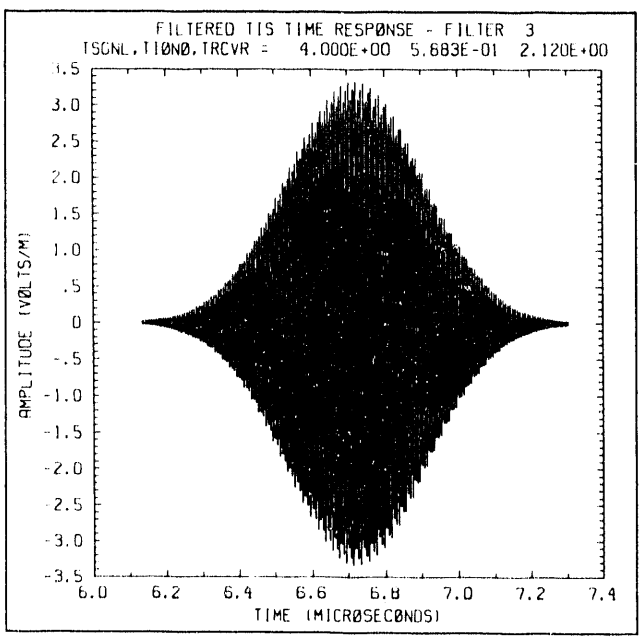

Figure 8c

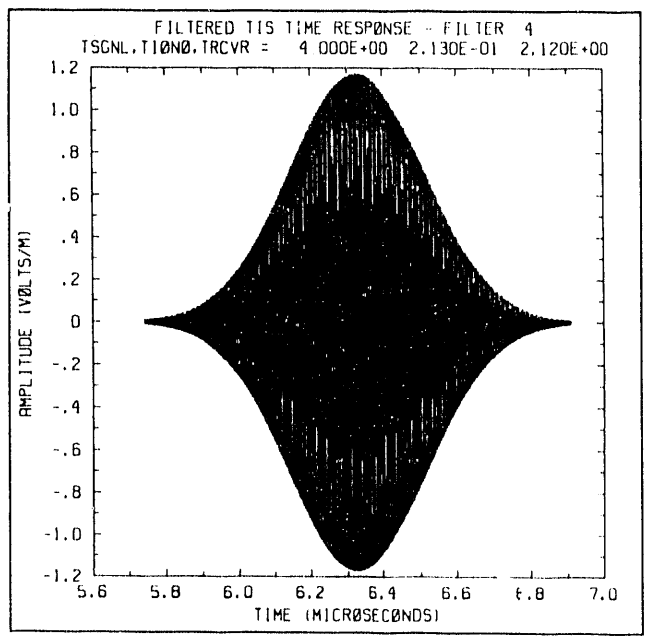

Figure 8d

Figures 8a–d (a)–(d) Filtered transionospheric signals obtained for the default delta function with the four Gaussian filters shown in Figures 5a–d, respectively. A TEC of 1×10^{13} cm^{-2}, F_{coh} = 20 MHz, and an SNR of 70 dB were assumed.

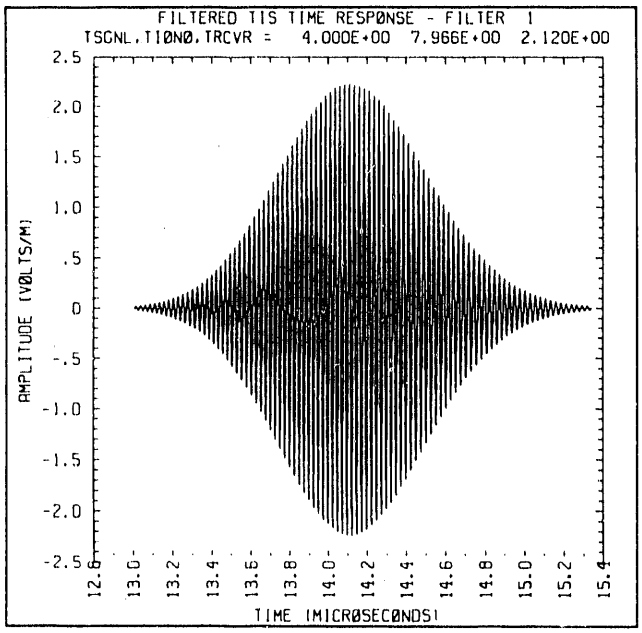

Figure 9a

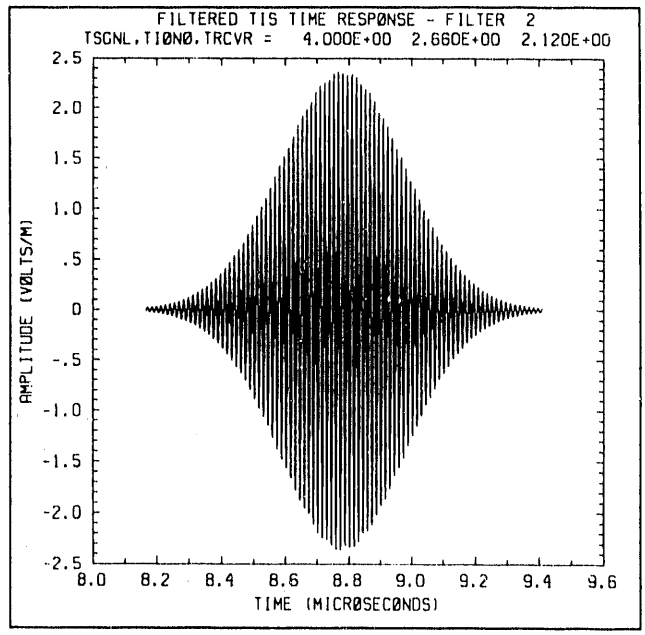

Figure 9b

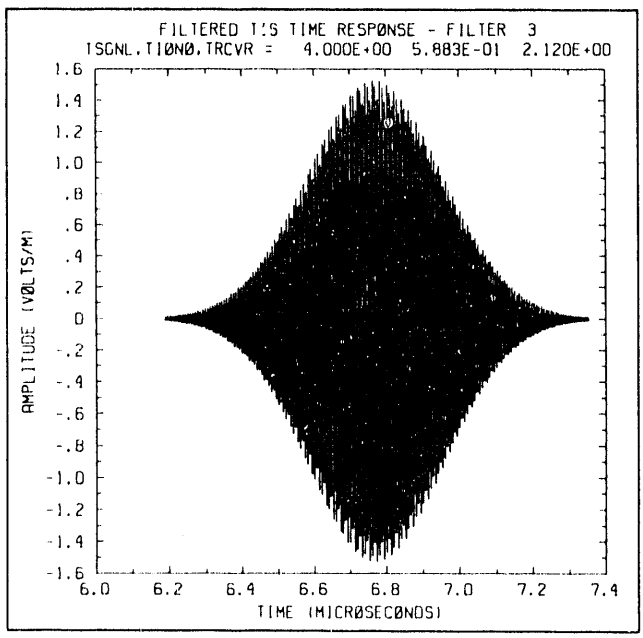

Figure 9c

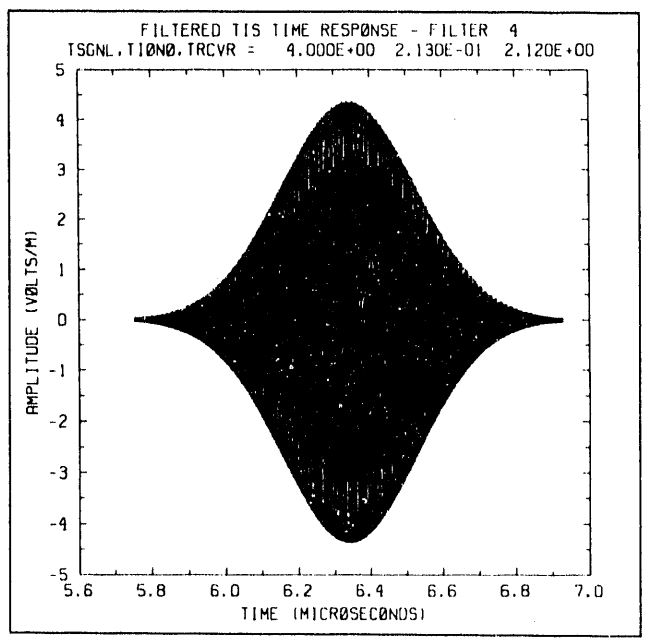

Figure 9d

Figures 9a–d (a)–(d) Filtered transionospheric signals obtained for the default delta function with the four Gaussian filters shown in Figures 5a–d, respectively. A TEC of 1×10^{13} cm^{-2}, F_{coh} = 10 MHz, and an SNR of 70 dB were assumed.

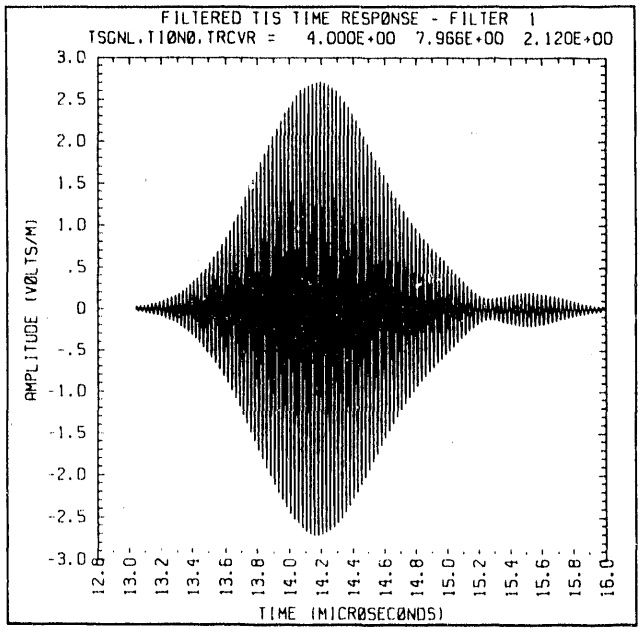

Figure 10a

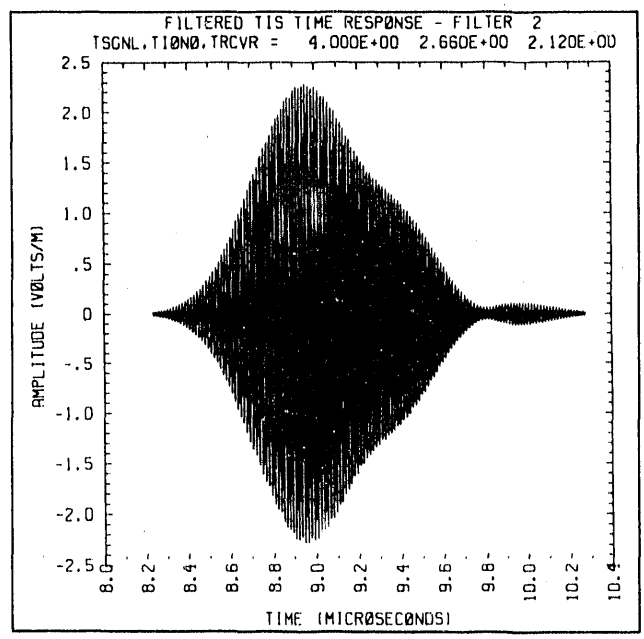

Figure 10b

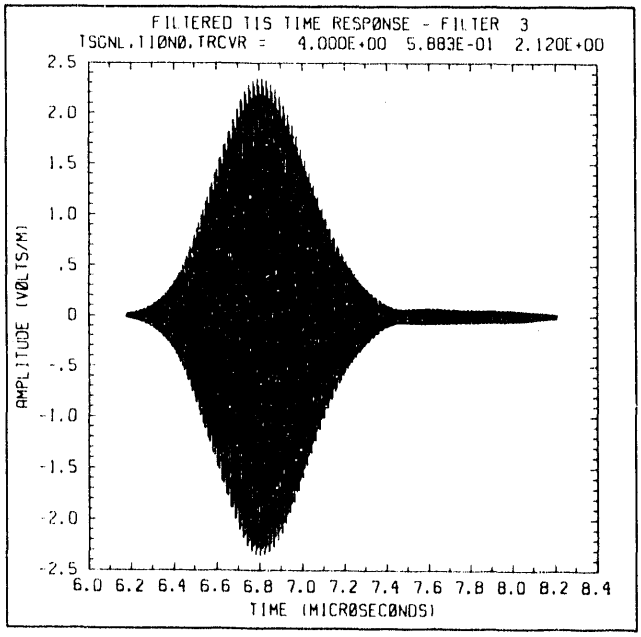

Figure 10c

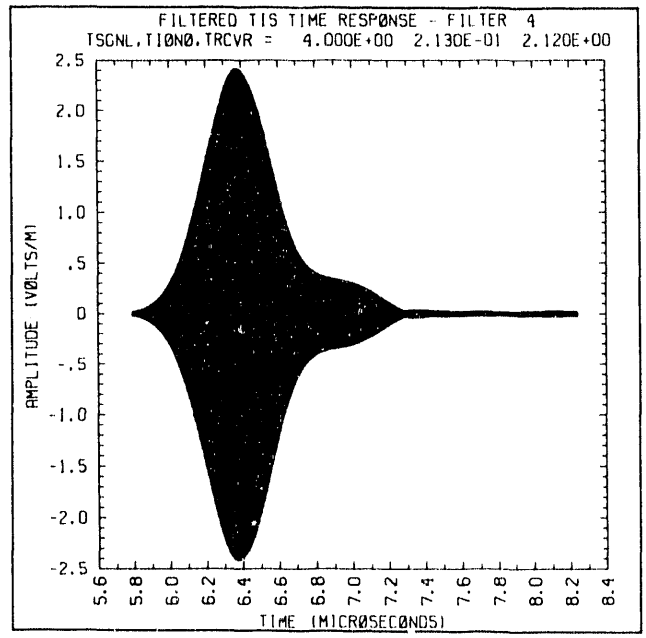

Figure 10d

Figures 10a–d (a)–(d) Filtered transionospheric signals obtained for the default delta function with the four Gaussian filters shown in Figures 5a–d, respectively. A TEC of 1×10^{13} cm $^{-2}$, F_{coh} = 1 MHz, and an SNR of 70 dB were assumed.

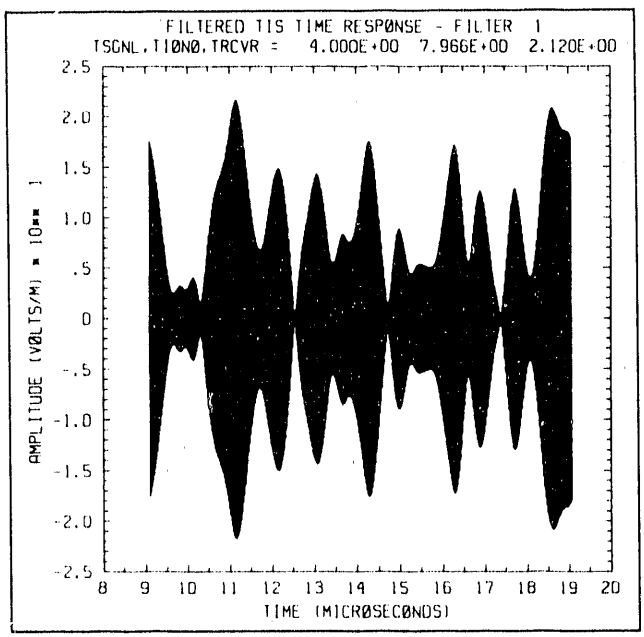

Figure 11a

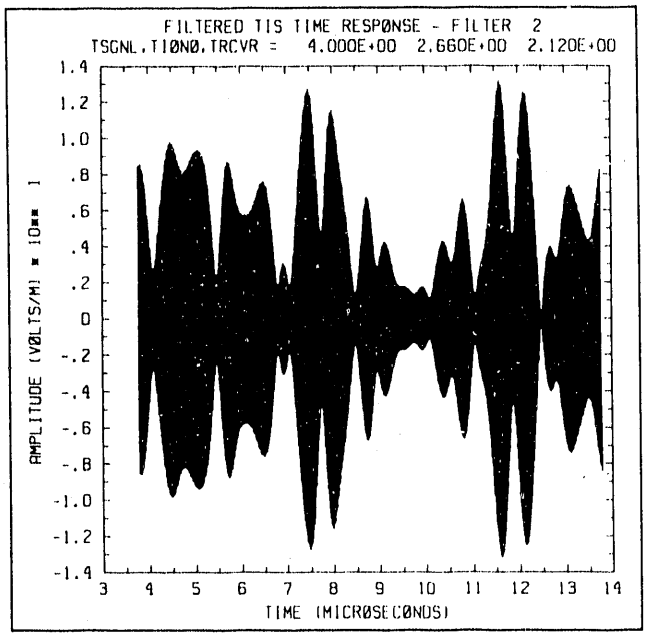

Figure 11b

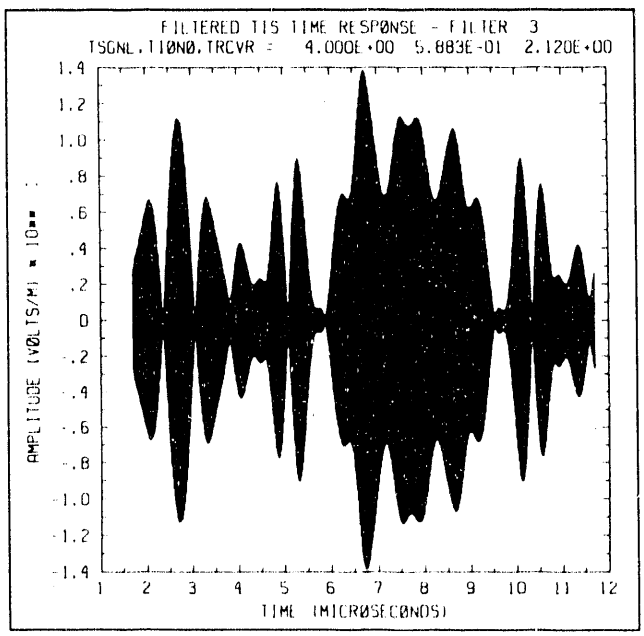

Figure 11c

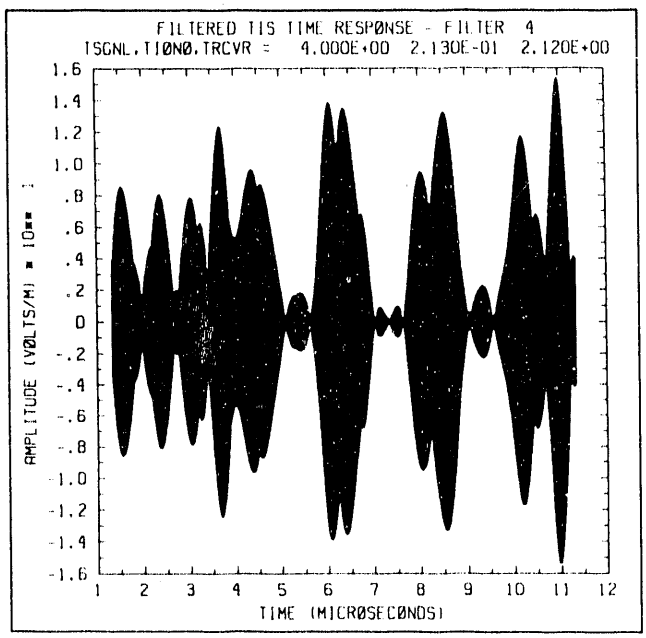

Figure 11d

Figures 11a–d (a)–(d) Filtered transionospheric signals obtained for the default delta function with the four Gaussian filters shown in Figures 5a–d, respectively. A TEC of 1×10^{13} cm $^{-2}$, $F_{coh} = \infty$, and an SNR of 100 were assumed.

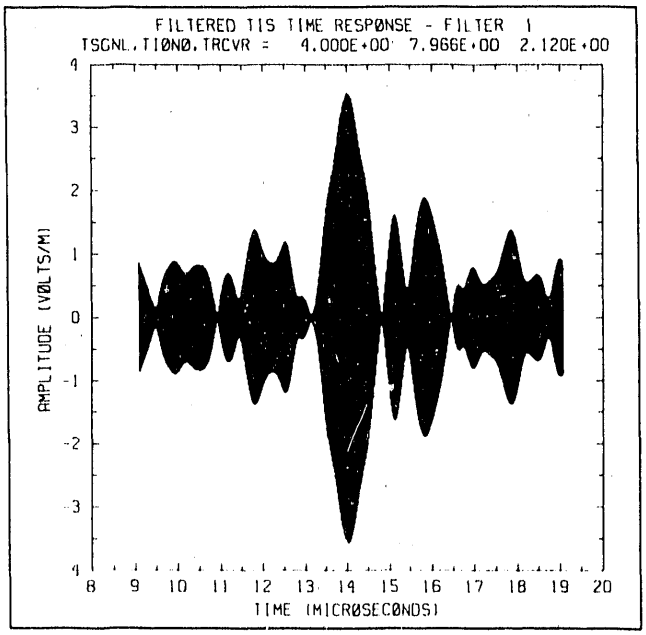

Figure 12a

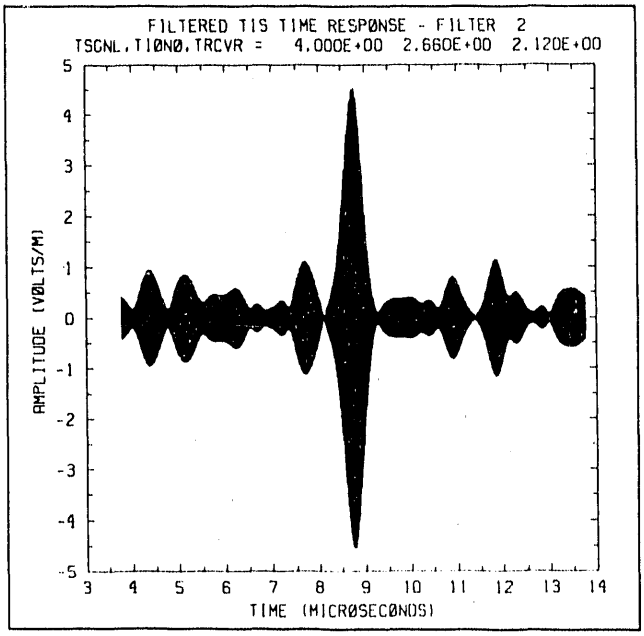

Figure 12b

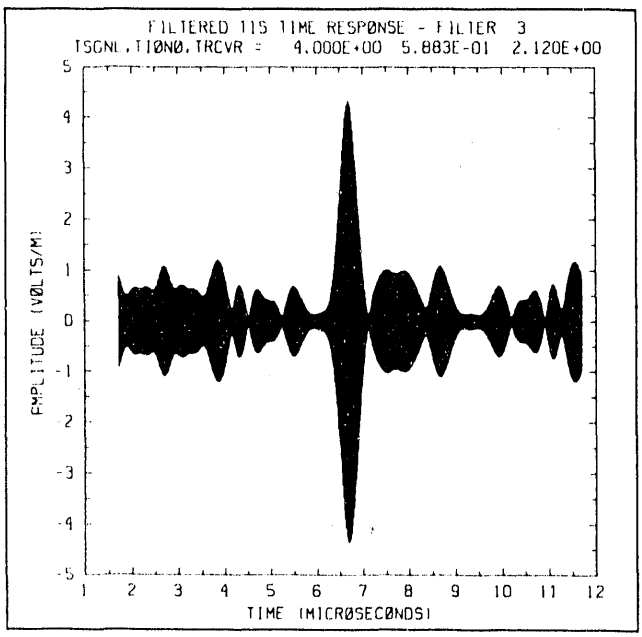

Figure 12c

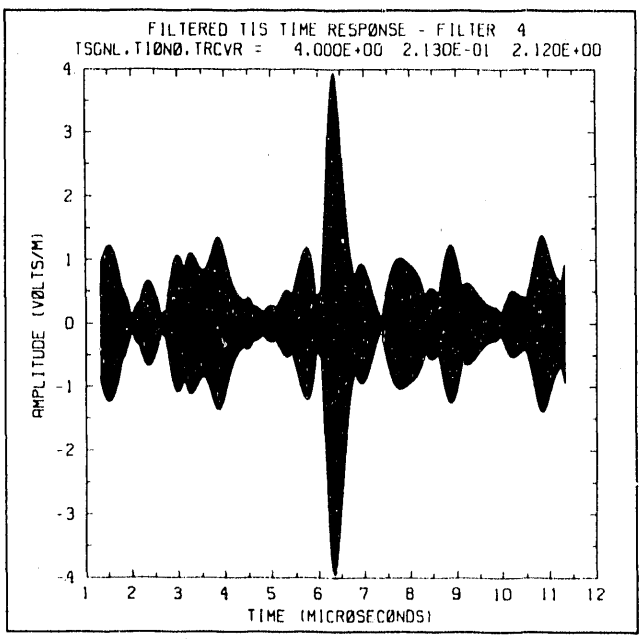

Figure 12d

Figures 12a–d (a)–(d) Filtered transionospheric signals obtained for the default delta function with the four Gaussian filters shown in Figures 5a–d, respectively. A TEC of 1×10^{13} cm^{-2}, $F_{coh} = \infty$, and an SNR of 10,000 were assumed.

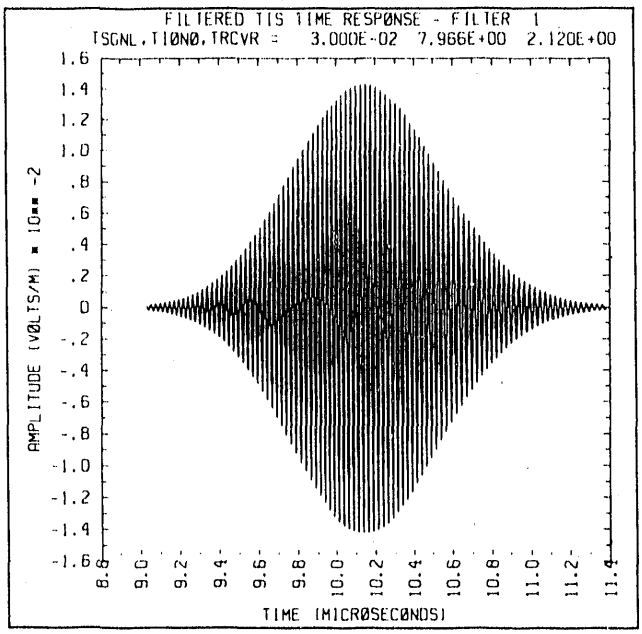

Figure 13a

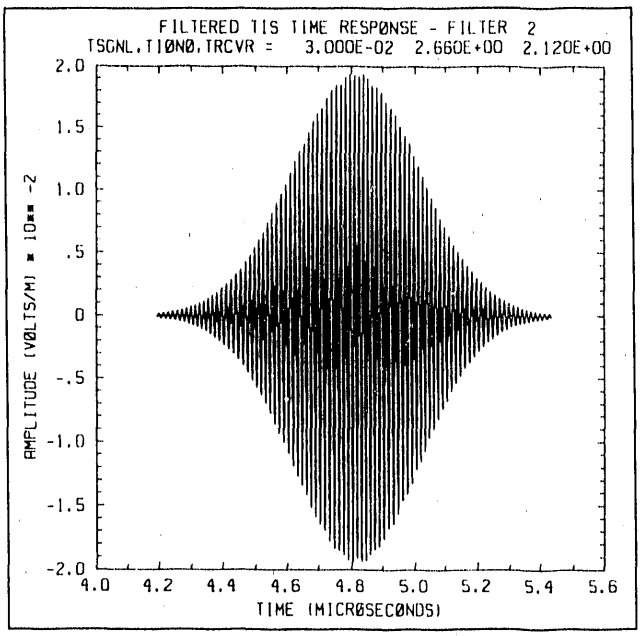

Figure 13b

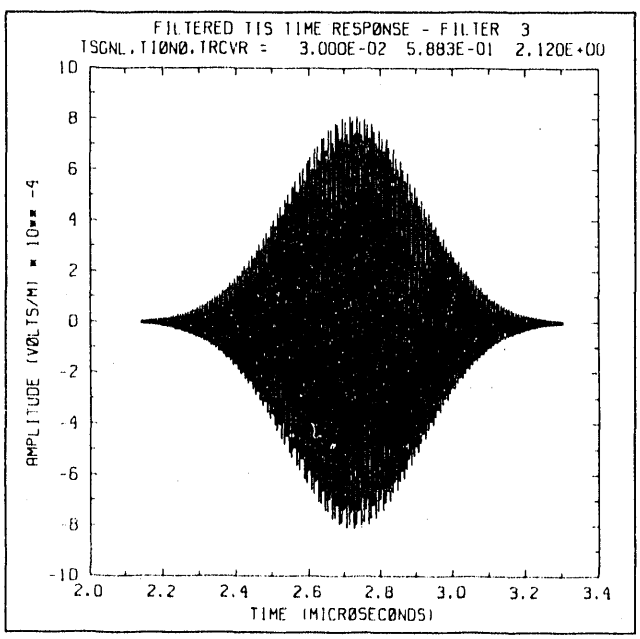

Figure 13c

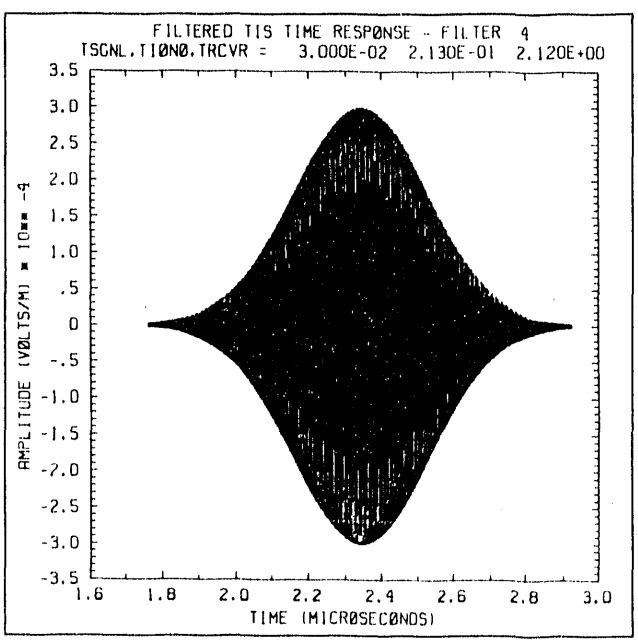

Figure 13d

Figures 13a–d (a)–(d) Filtered transionospheric signals obtained for the default super-Gaussian pulse with the four Gaussian filters shown in Figures 5a–d, respectively. A TEC of 1×10^{13} cm^{-2}, $F_{coh} = \infty$, and an SNR of 70 dB were assumed.

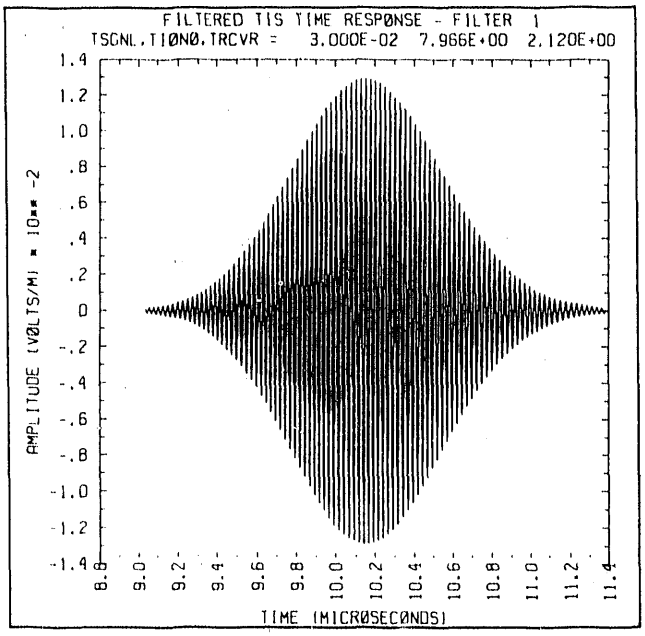

Figure 14a

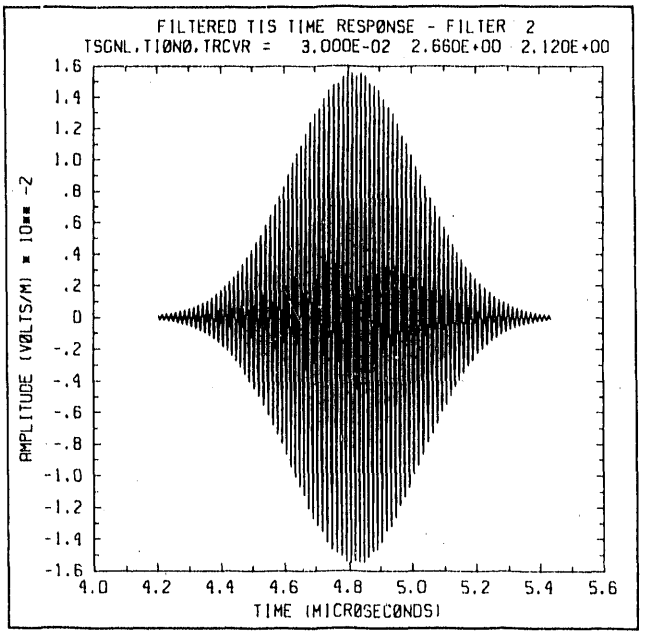

Figure 14b

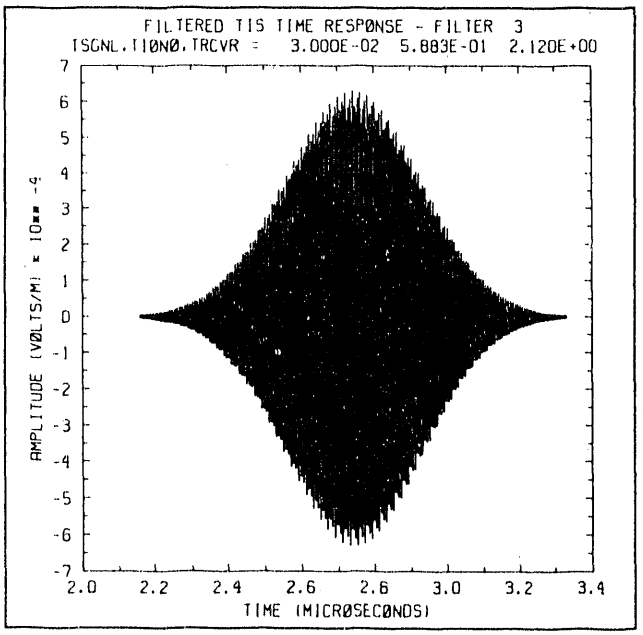

Figure 14c

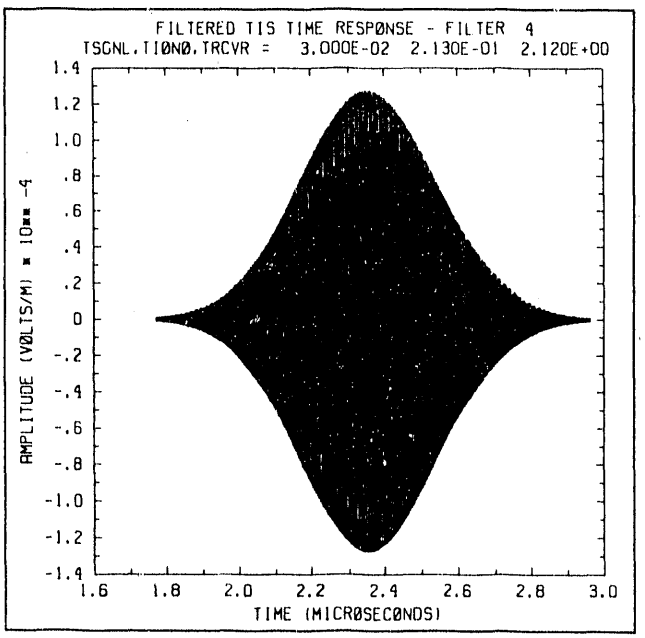

Figure 14d

Figures 14a-d (a)–(d) Filtered transionospheric signals obtained for the default super-Gaussian pulse with the four Gaussian filters shown in Figures 5a–d, respectively. A TEC of 1×10^{13} cm^{-2}, F_{coh} = 20 MHz, and an SNR of 70 dB were assumed.

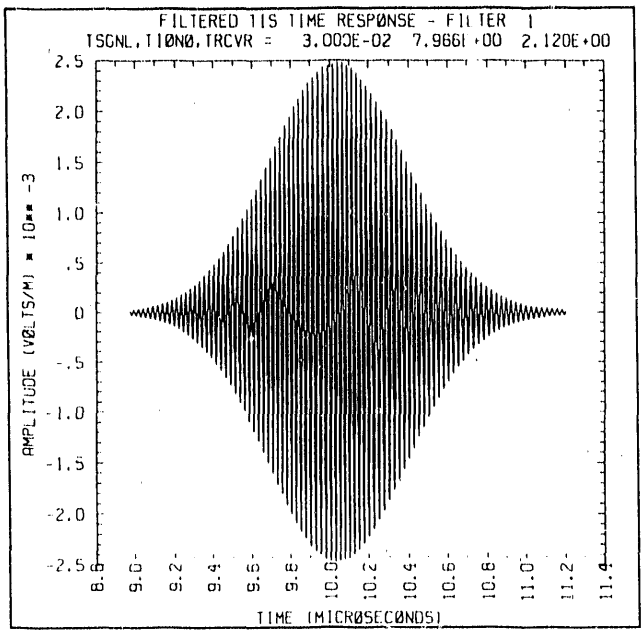

Figure 15a

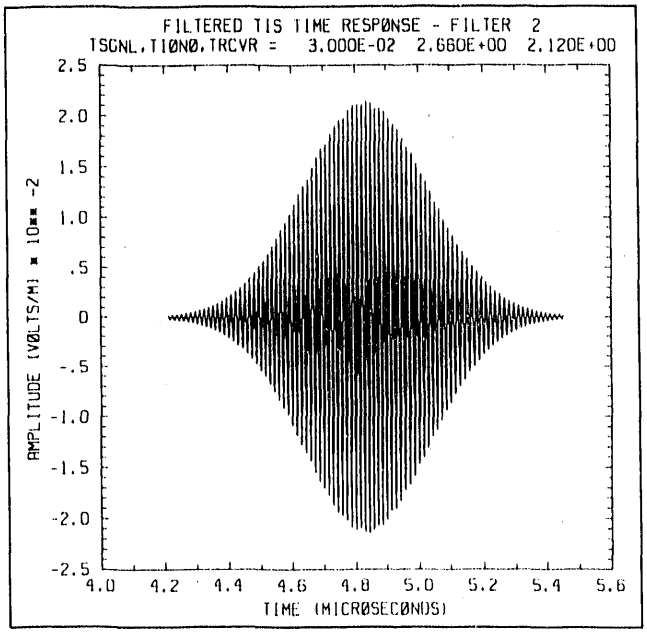

Figure 15b

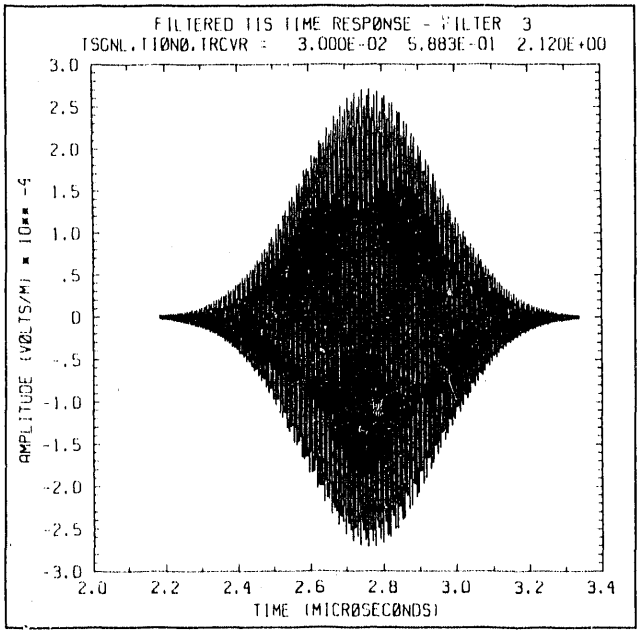

Figure 15c

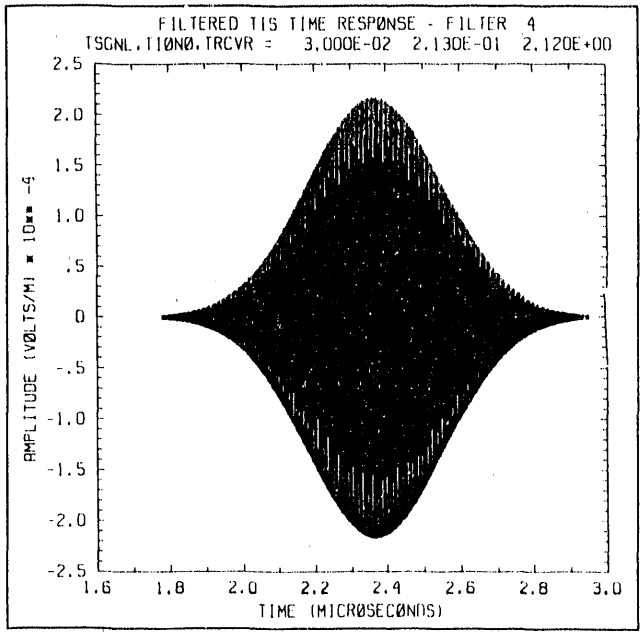

Figure 15d

Figures 15a–d (a)–(d) Filtered transionospheric signals obtained for the default super-Gaussian pulse with the four Gaussian filters shown in Figures 5a–d, respectively. A TEC of 1×10^{13} cm^{-2}, F_{coh} = 10 MHz, and an SNR of 70 dB were assumed.

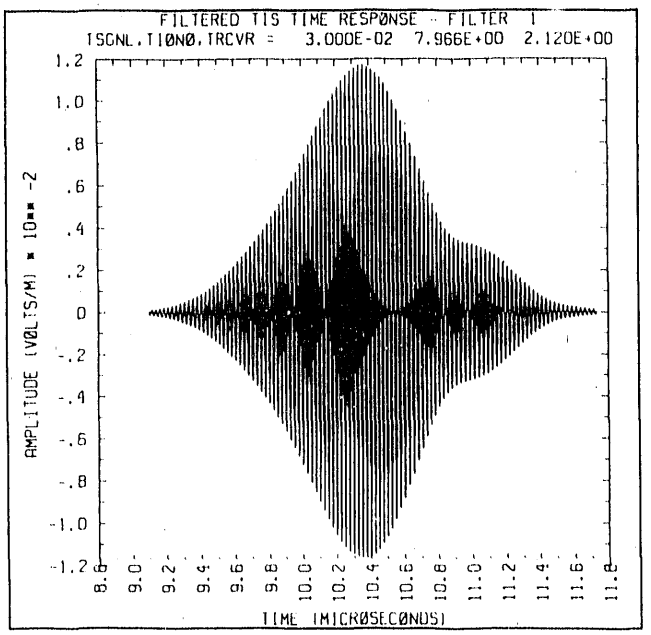

Figure 16a

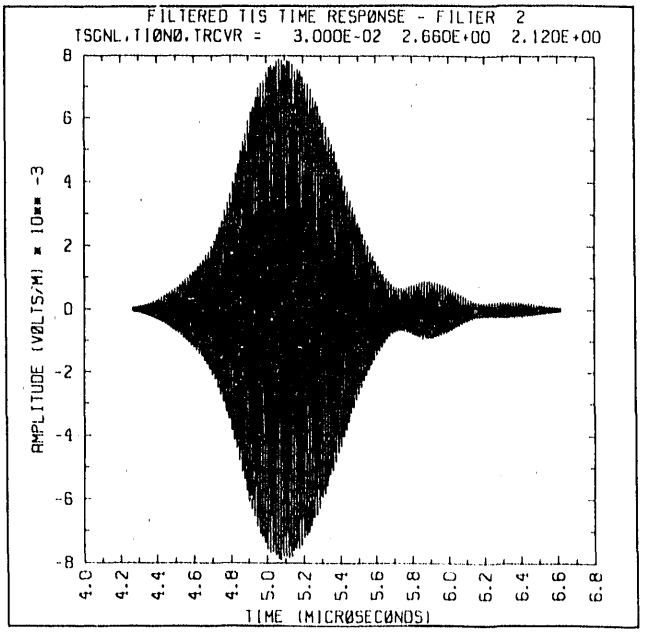

Figure 16b

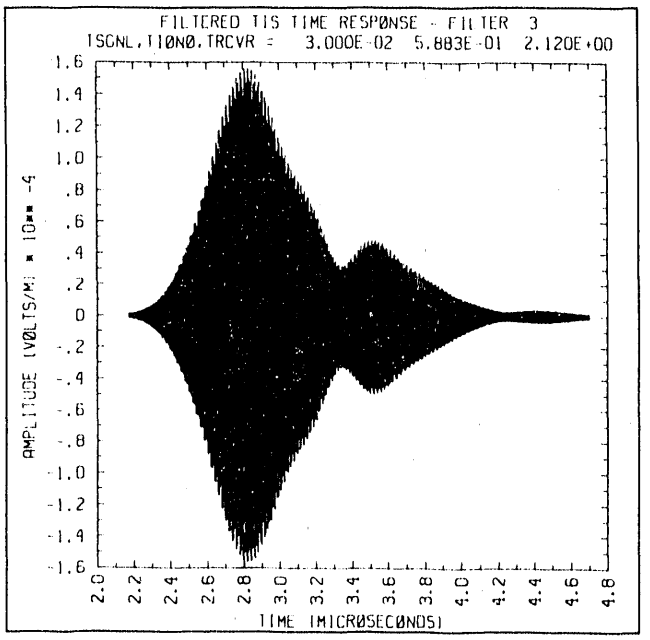

Figure 16c

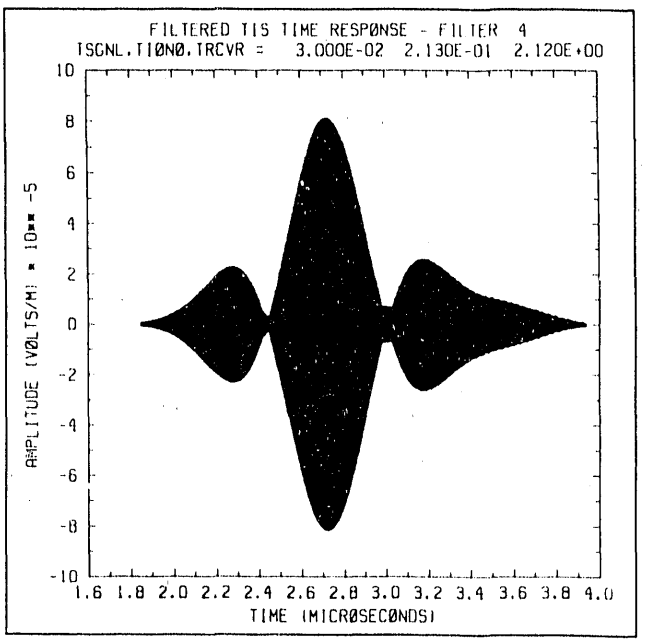

Figure 16d

Figures 16a–d (a)–(d) Filtered transionospheric signals obtained for the default super-Gaussian pulse with the four Gaussian filters shown in Figures 5a–d, respectively. A TEC of 1×10^{13} cm^{-2}, F_{coh} = 1 MHz, and an SNR of 70 dB were assumed.

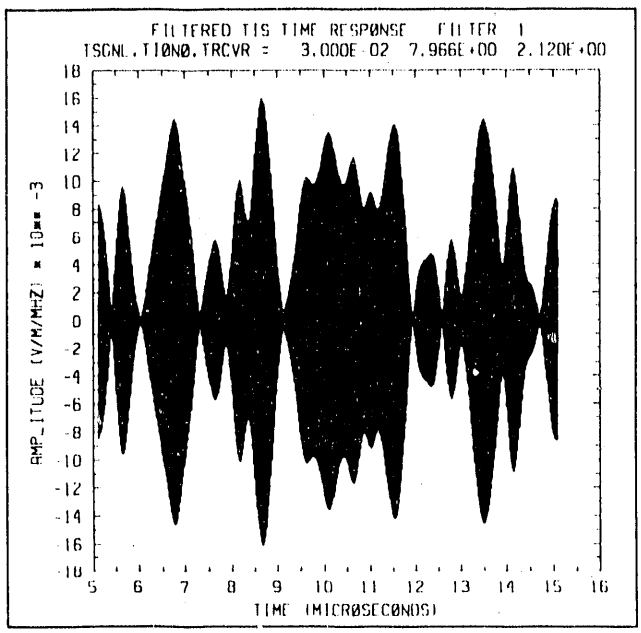

Figure 17a

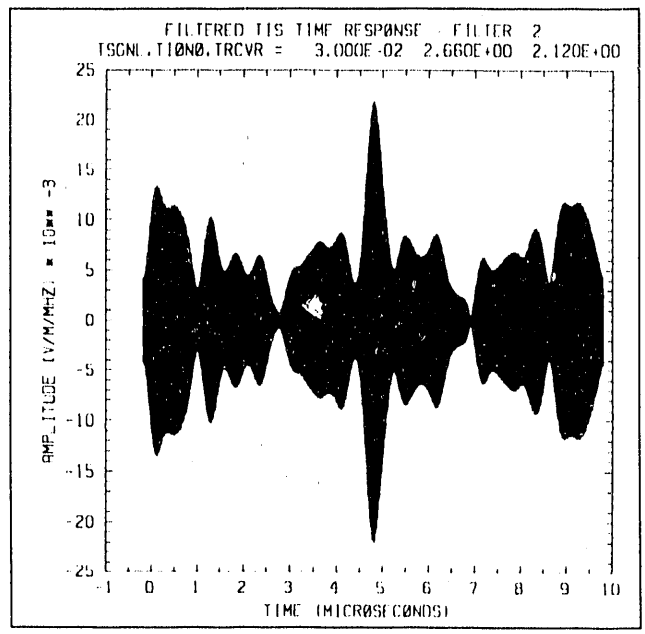

Figure 17b

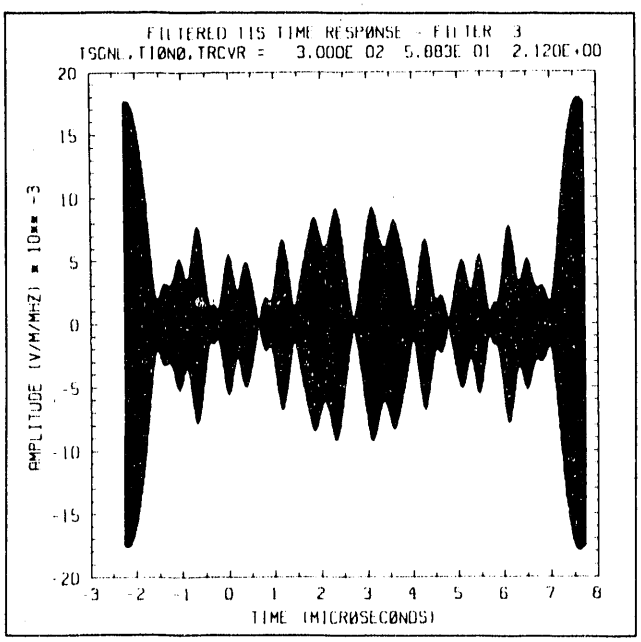

Figure 17c

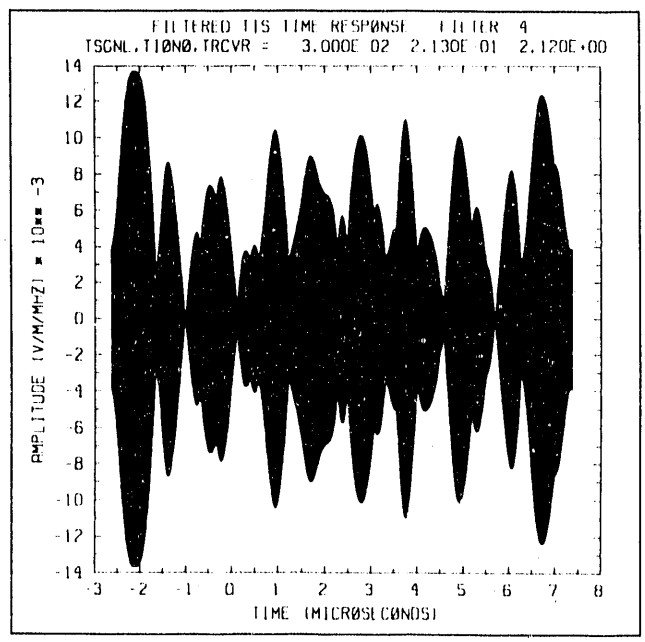

Figure 17d

Figures 17a–d (a)–(d) Filtered transionospheric signals obtained for the default super-Gaussian pulse with the four Gaussian filters shown in Figures 5a–d, respectively. A TEC of 1×10^{13} cm $^{-2}$, $F_{coh} = \infty$, and an SNR of 100 were assumed.

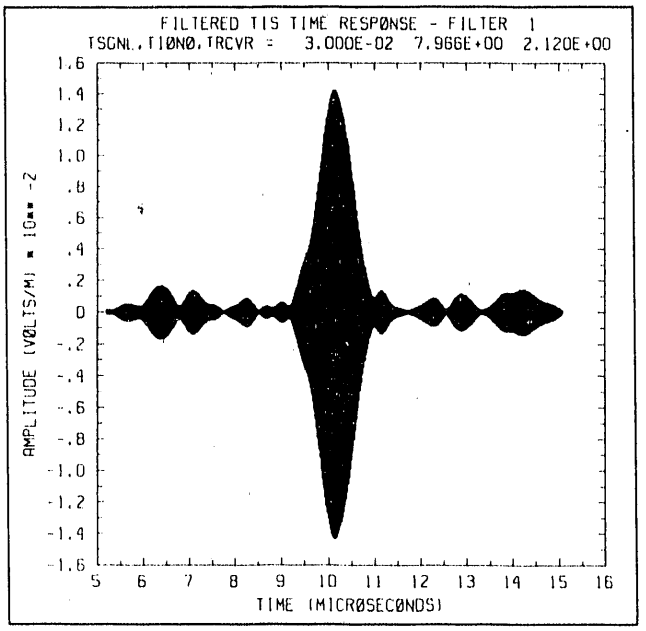

Figure 18a

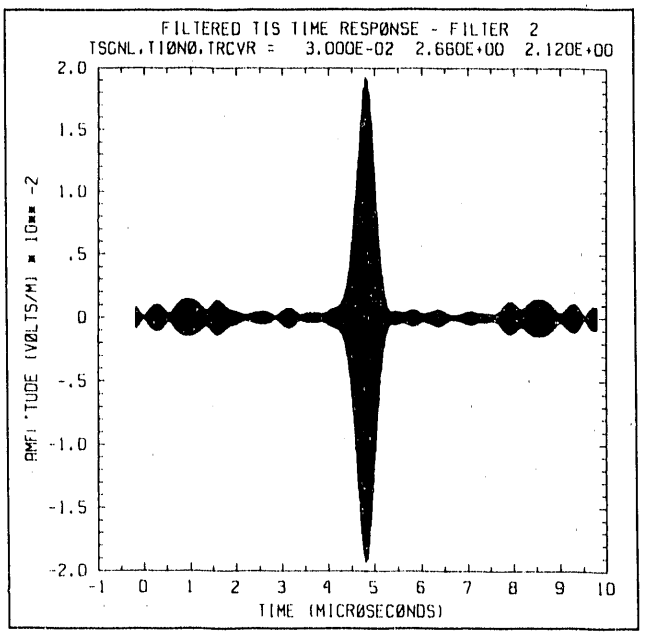

Figure 18b

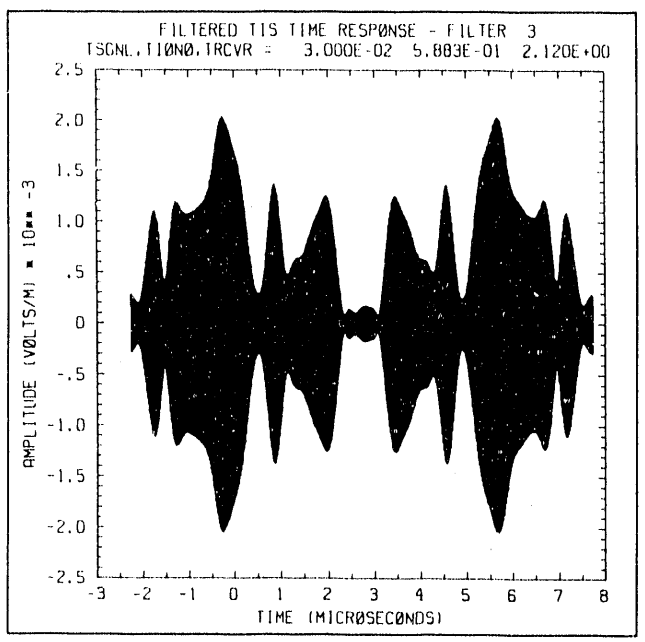

Figure 18c

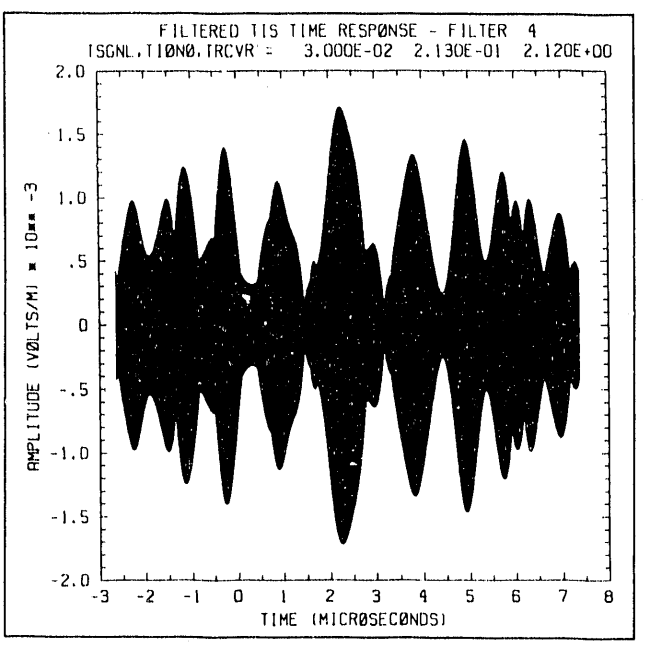

Figure 18d

Figures 18a-d (a)–(d) Filtered transionospheric signals obtained for the default super-Gaussian pulse with the four Gaussian filters shown in Figures 5a–d, respectively. A TEC of 1×10^{13} cm $^{-2}$, $F_{coh} = \infty$, and an SNR of 10,000 were assumed.

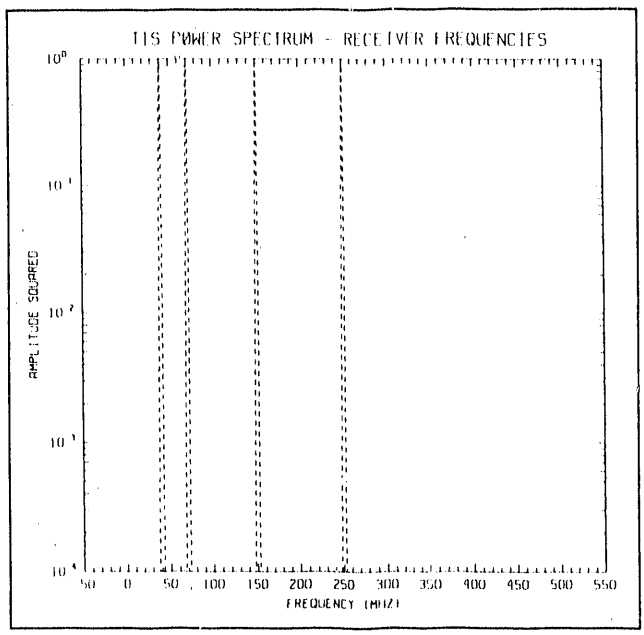

Figure 19a

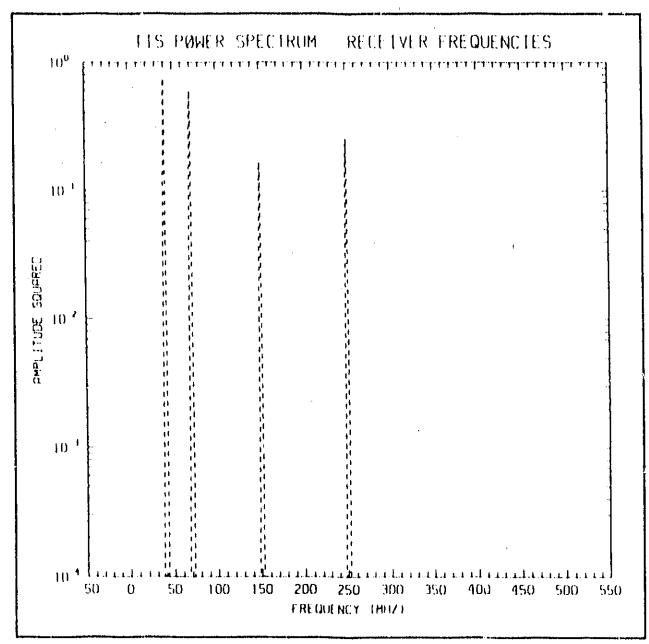

Figure 19b

Figures 19a–f Received power in the four Gaussian filters shown in Figures 5a–d for the default delta function. The solid curve is a plot of the FFT of the input pulse while the peak values of the dashed curves represent the magnitude of the received power in the respective channels. (a) Results for a TEC of 1×10^{13} cm^{-2}, $F_{coh} = \infty$, and SNR of 70 dB. (b)–(d) Results for a TEC of 1×10^{13} cm^{-2}, an SNR of 70 dB, and F_{coh} of 20, 10 and 1 MHz, respectively. (e)–(f) Results for a TEC of 1×10^{13} cm^{-2}, $F_{coh} = \infty$, and SNR of 100 and 10,000, respectively.

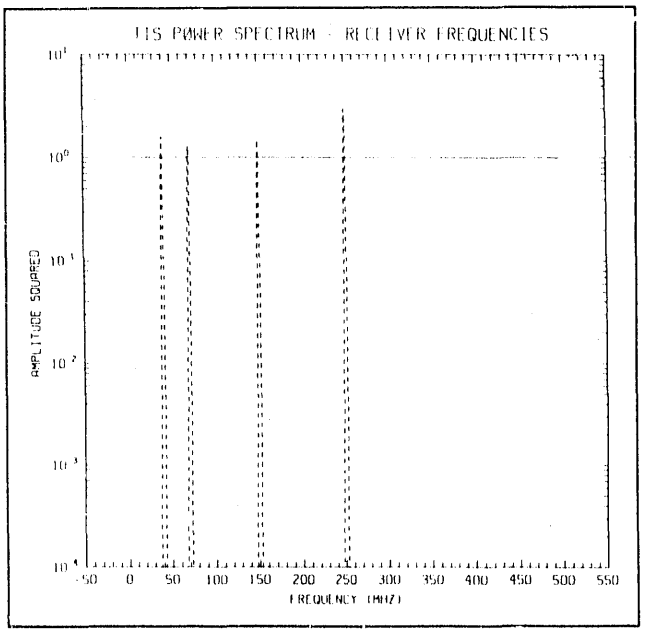

Figure 19c

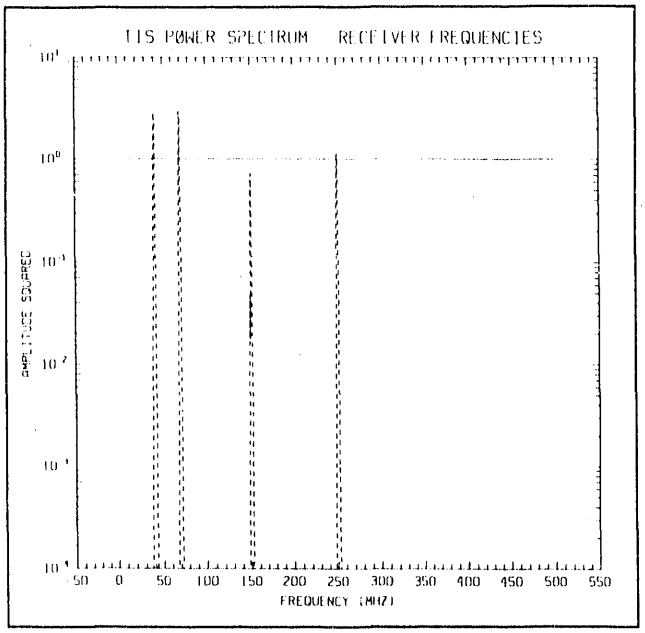

Figure 19d

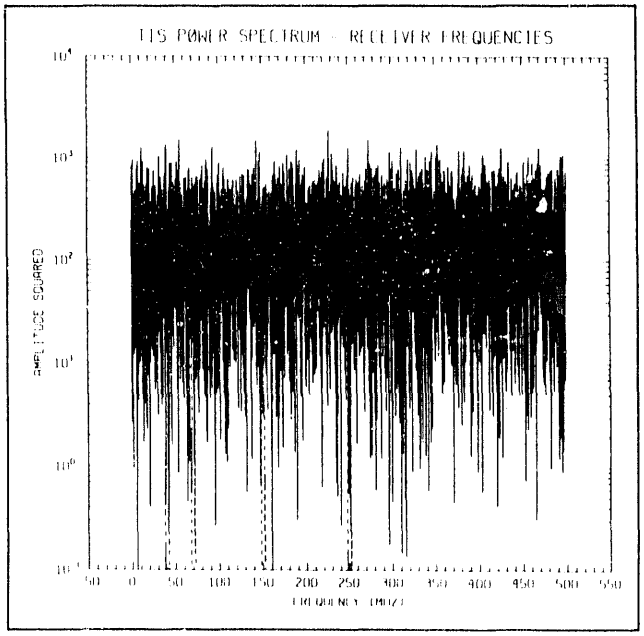

Figure 19e

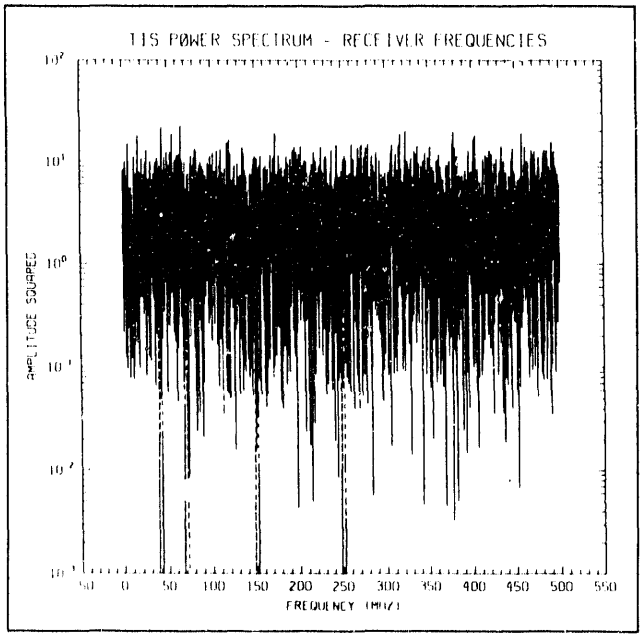

Figure 19f

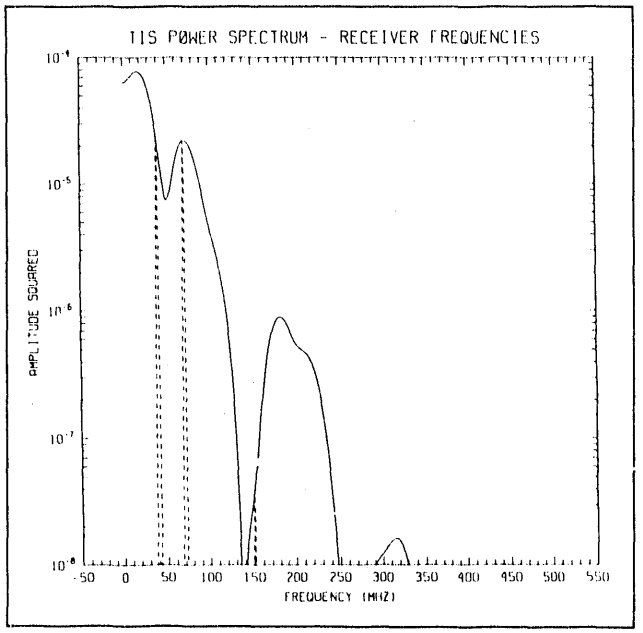

Figure 20a

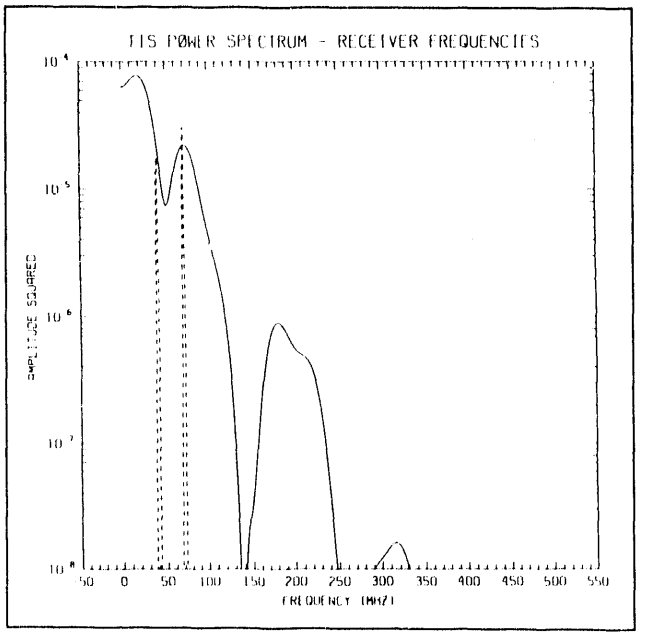

Figure 20b

Figures 20a–f Received power in the four Gaussian filters shown in Figures 5a–d for the default super-Gaussian pulse. The solid curve is a plot of the FFT of the input pulse while the peak values of the dashed curves represent the magnitude of the received power in the respective channels. (a) Results for a TEC of 1×10^{13} cm^{-2}, $F_{coh} = \infty$, and SNR of 70 dB. (b)–(d) Results for a TEC of 1×10^{13} cm^{-2}, an SNR of 70 dB, and F_{coh} of 20, 10 and 1 MHz, respectively. (e)–(f) Results for a TEC of 1×10^{13} cm^{-2}, $F_{coh} = \infty$, and SNR of 100 and 10,000, respectively.

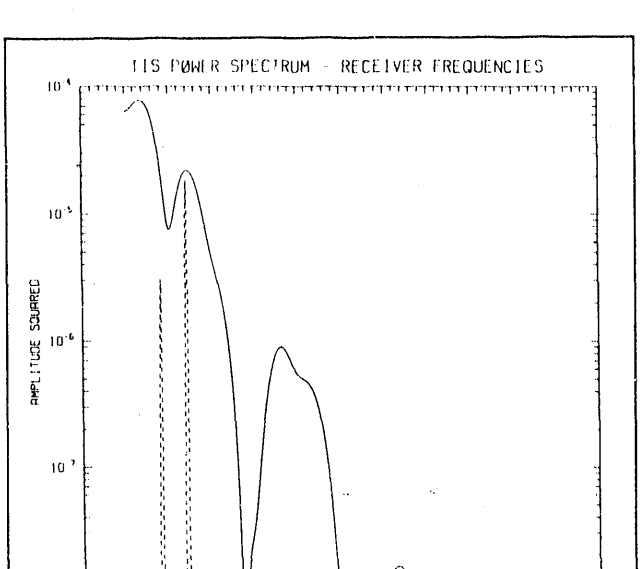

Figure 20c

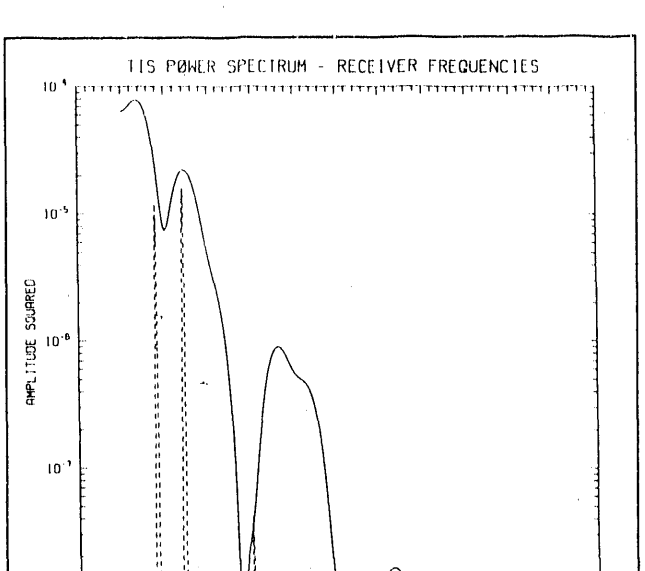

Figure 20d

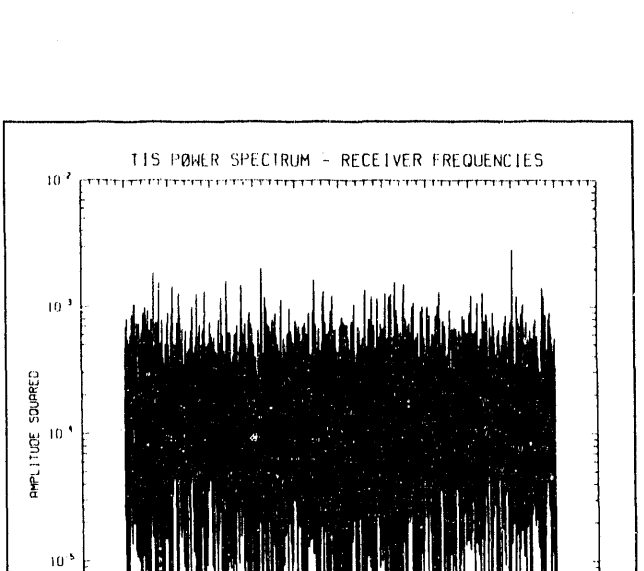

Figure 20e

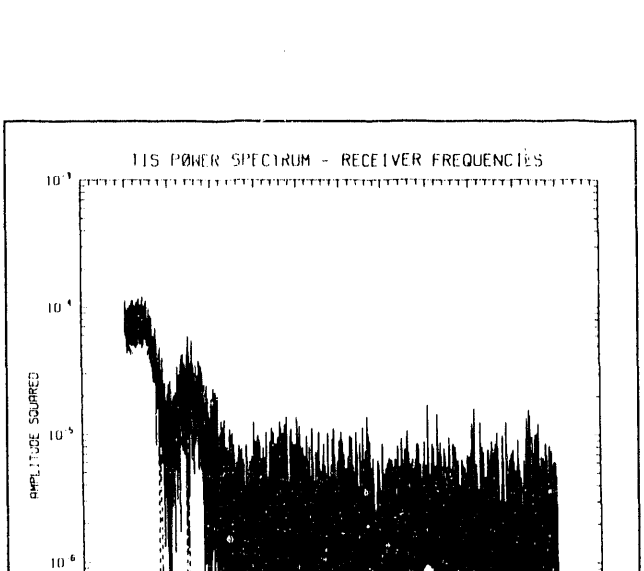

Figure 20f

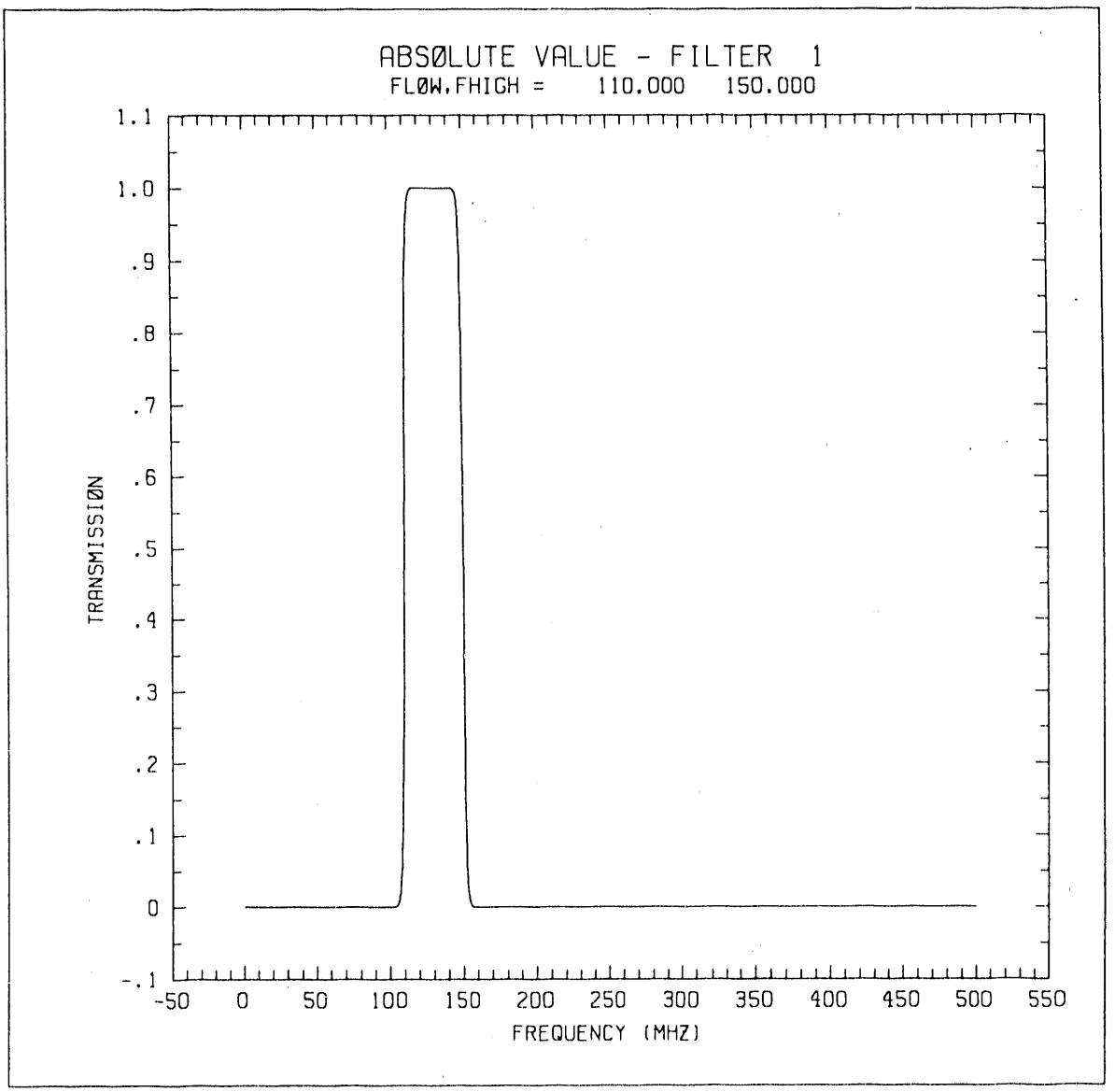

Figure 21 Transmission profile for a wideband filter with a central frequency of 130 MHz and FWHM of 40 MHz.

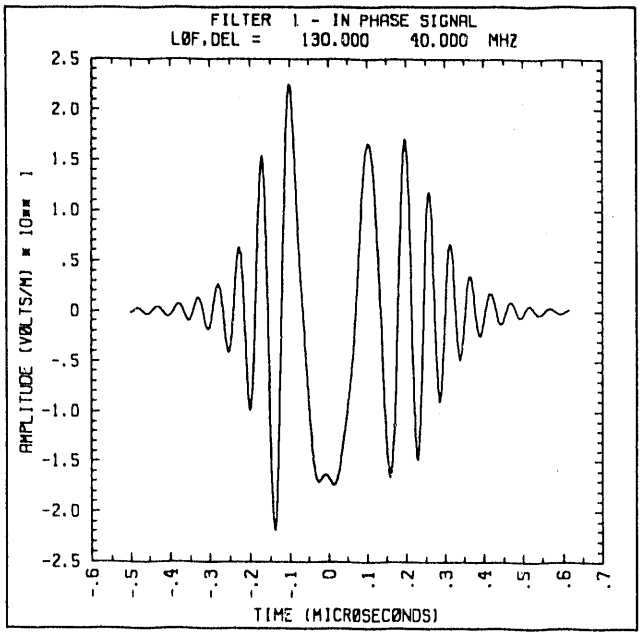

Figure 22a

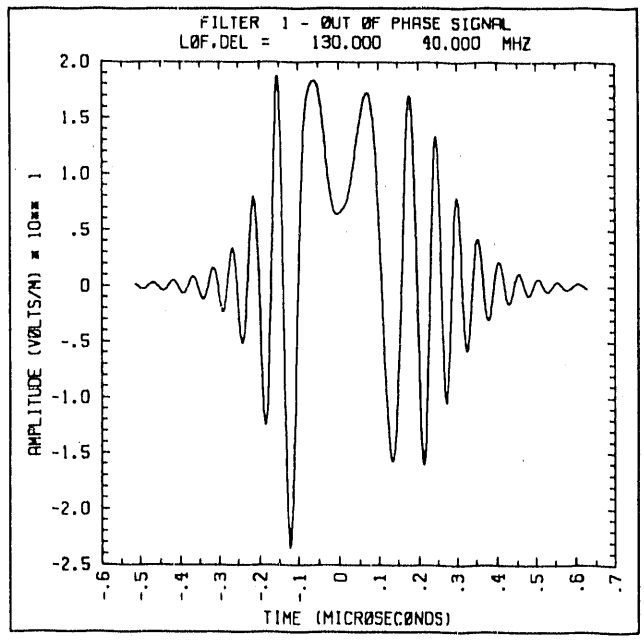

Figure 22b

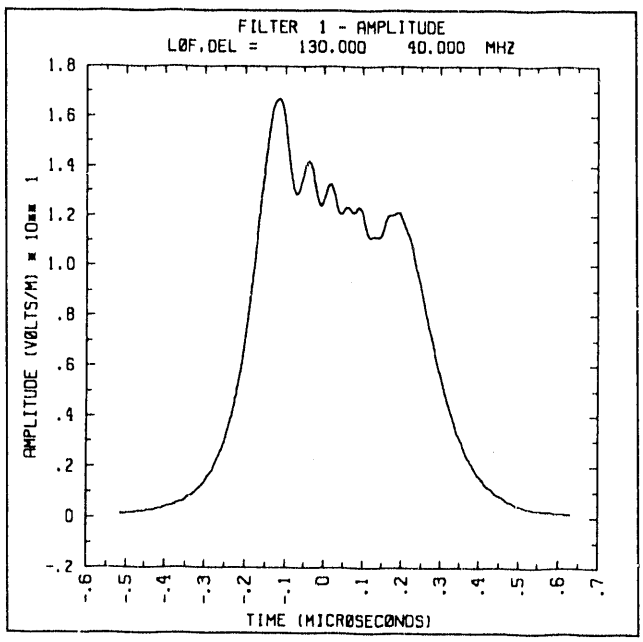

Figure 22c

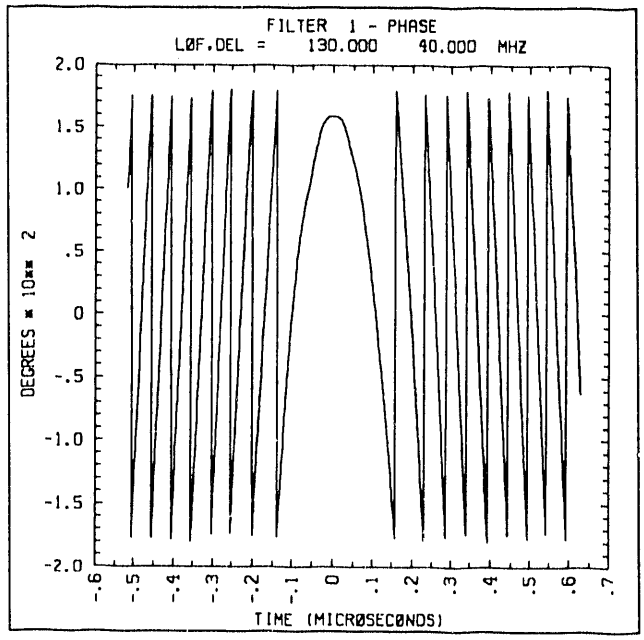

Figure 22d

Figures 22a–d Results of quadrature detection with the filter shown in Figure 21 and for the default delta function and an ionosphere with TEC = 1×10^{13} cm^{-2}, $F_{coh} = \infty$, and SNR = 70 dB. (a) In-phase component (I), (b) Out-of-phase component (Q), (c) Amplitude and, (d) Phase of filtered transionospheric signal.

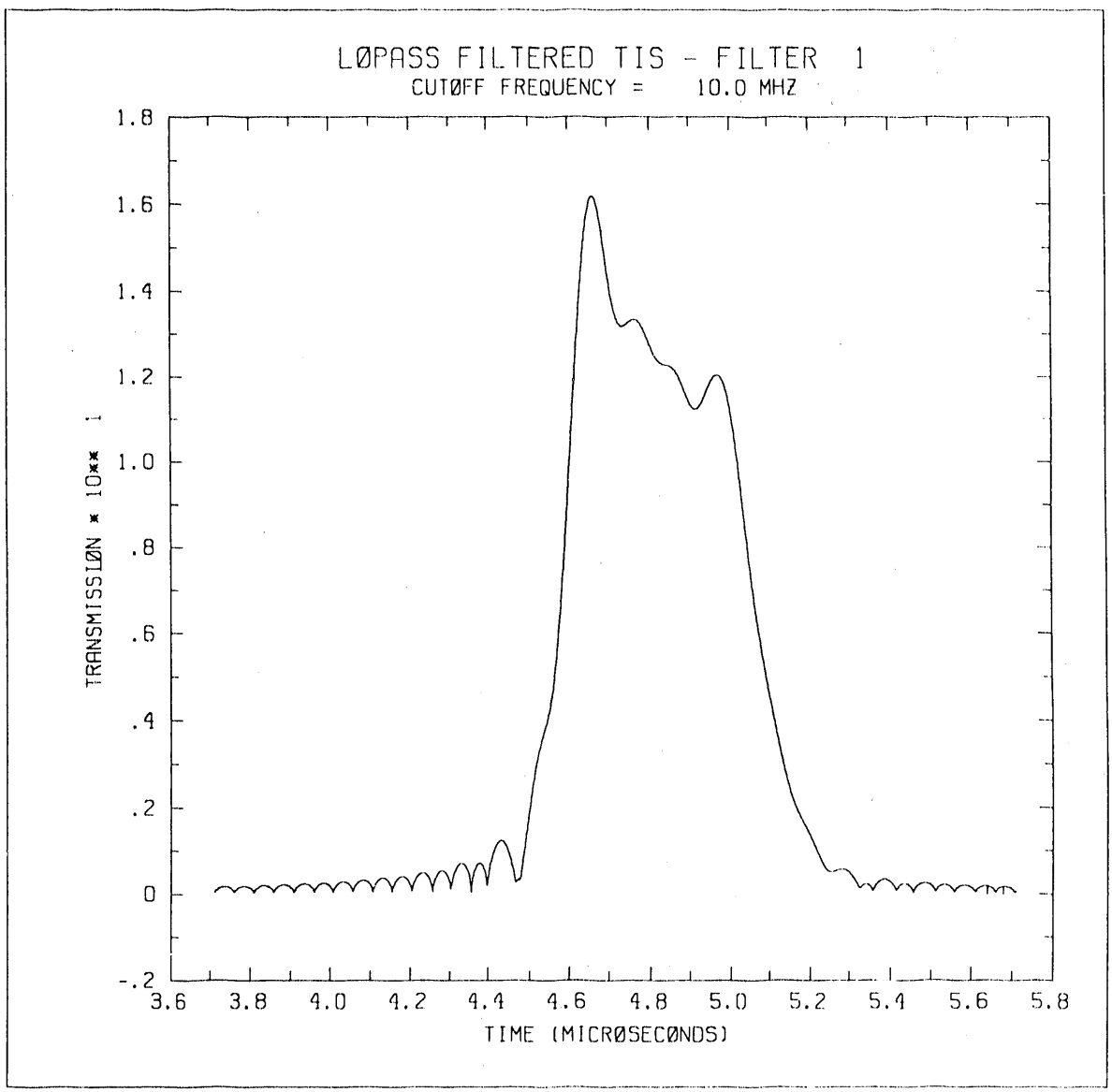

Figure 23 Results of square-law detection with the filter shown in Figure 21 and for the default delta function and an ionosphere with TEC = 1×10^{13} cm^{-2}, $F_{coh} = \infty$, and SNR = 70 dB. The low-pass filter used in this exercise has a FWHM equal to 10 MHz.

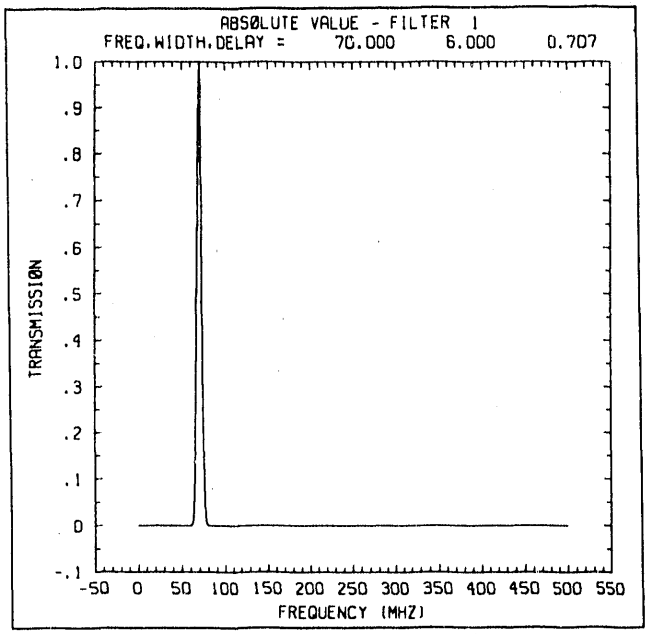

Figure 24a

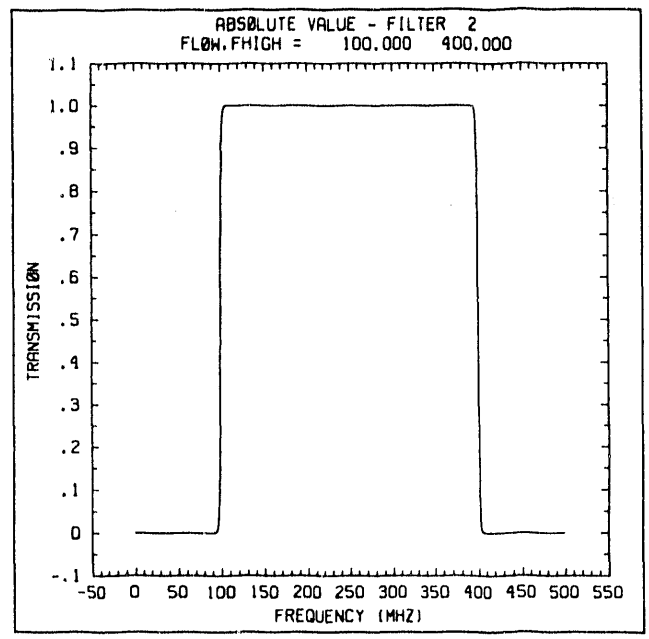

Figure 24b

Figures 24a–b Filter transmission functions used in TOA analysis. (a) Wideband filter with a FWHM of 300 MHz and 50% transmission points at 100 and 400 MHz. (b) Narrowband Gaussian filter with a central frequency of 70 MHz and FWHM of 6 MHz.

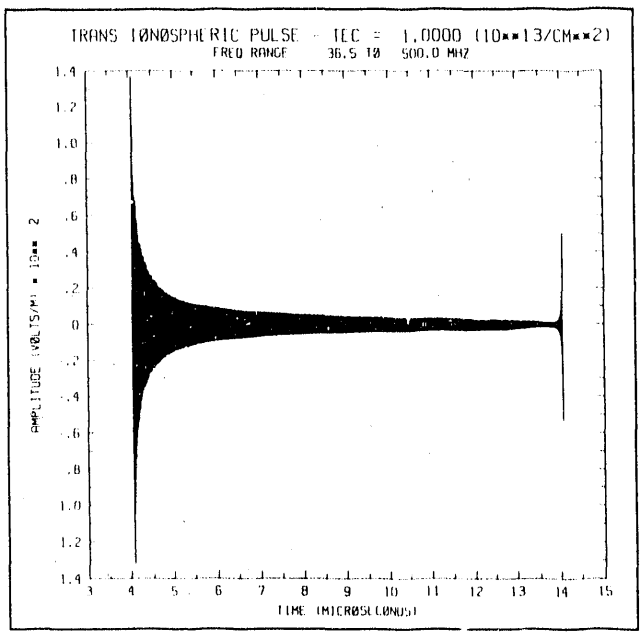

Figure 25a

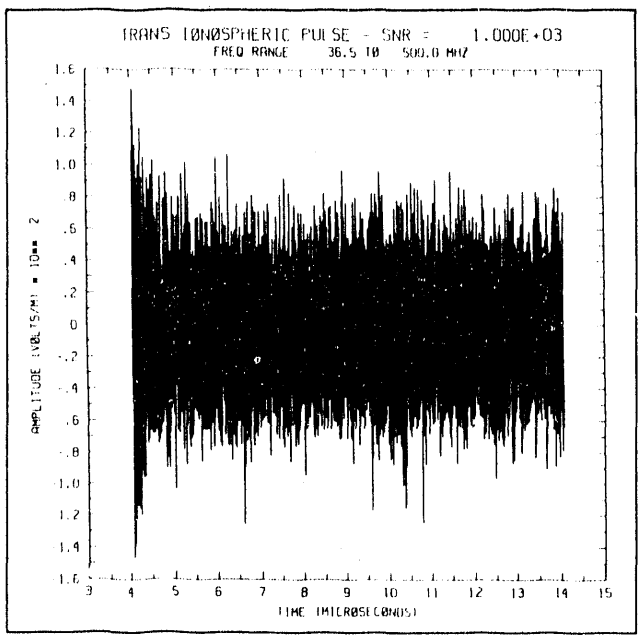

Figure 25b

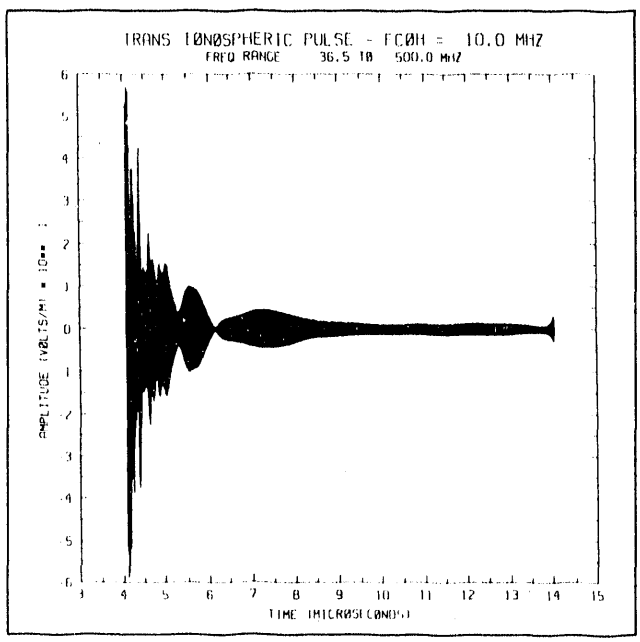

Figure 25c

Figures 25a–c Plots of transionospheric signals obtained for the default delta function, a TEC of 1×10^{120} cm^{-2} and three sets of parameters (a) infinite coherence bandwidth and 70 dB SNR, (b) infinite coherence bandwidth and 30 dB SNR, and (c) 10 MHz coherence bandwidth and 70 dB SNR.

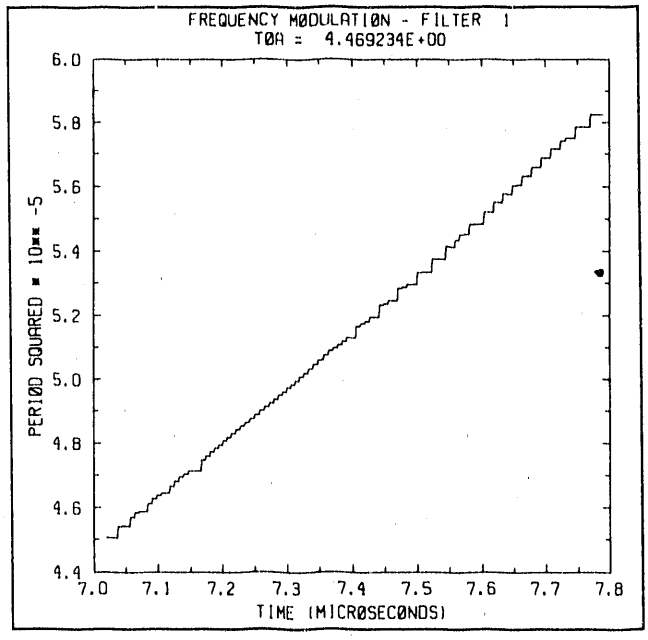

Figure 26a

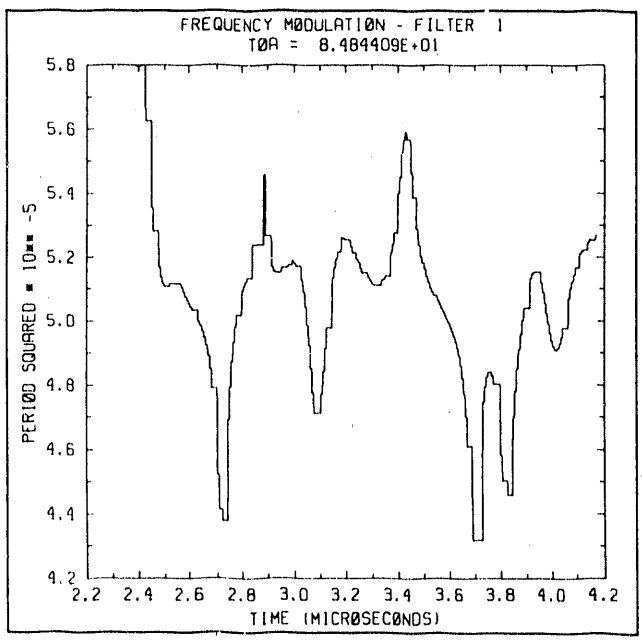

Figure 26b

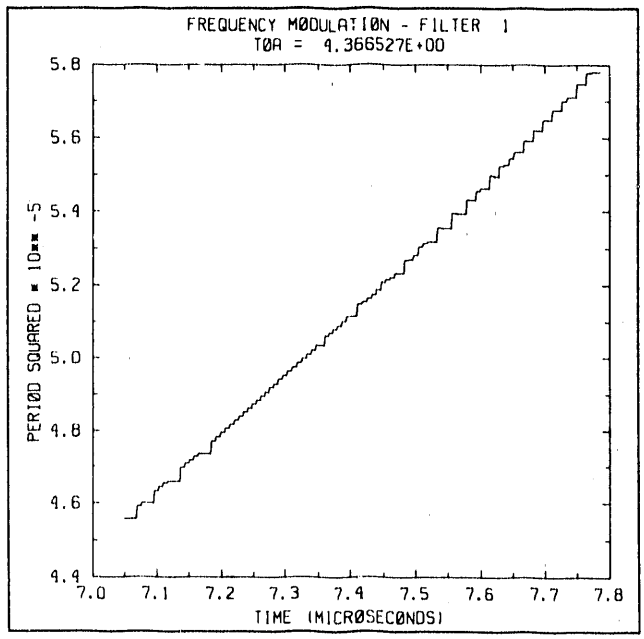

Figure 26c

Figures 26a–c The period squared computed by the FMTOA algorithm plotted vs. time for the wideband filter shown in Figure 24a. The default delta function, a TEC of 1×10^{13} cm^{-2} and the three sets of parameters (a) infinite coherence bandwidth and 70 dB SNR, (b) infinite coherence bandwidth and 30 dB SNR, and (c) 10 MHz coherence bandwidth and 70 dB SNR, are assumed for the three figures, respectively.

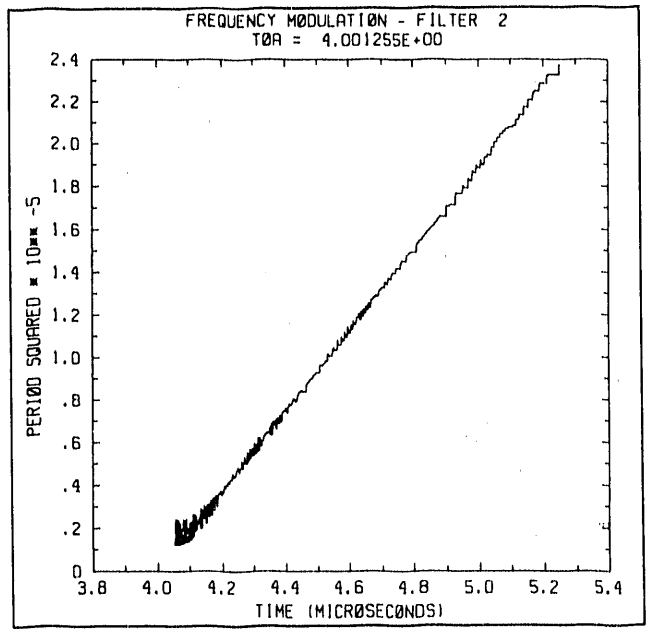

Figure 27a

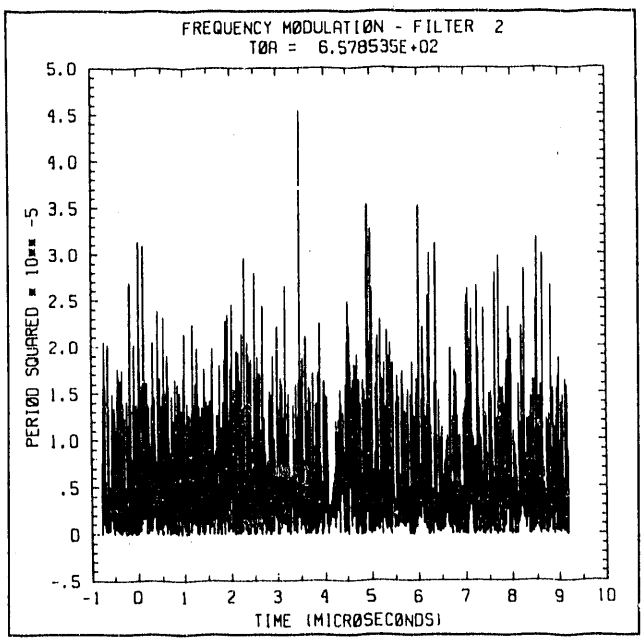

Figure 27b

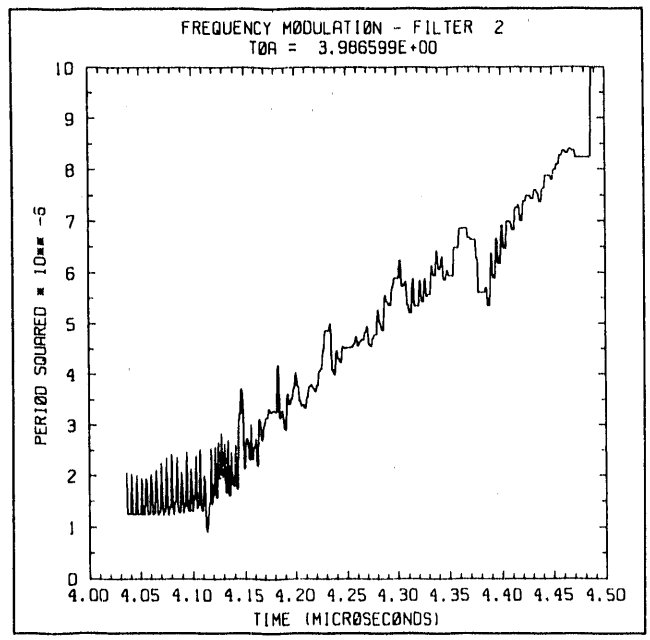

Figure 27c

Figures 27a–c The period squared computed by the FMTOA algorithm plotted vs. time for the narrowband filter shown in Figure 24b. The default delta function, a TEC of 1×10^{13} cm^{-2} and the three sets of parameters (a) infinite coherence bandwidth and 70 dB SNR, (b) infinite coherence bandwidth and 30 dB SNR, and (c) 10 MHz coherence bandwidth and 70 dB SNR, are assumed for the three figures, respectively.

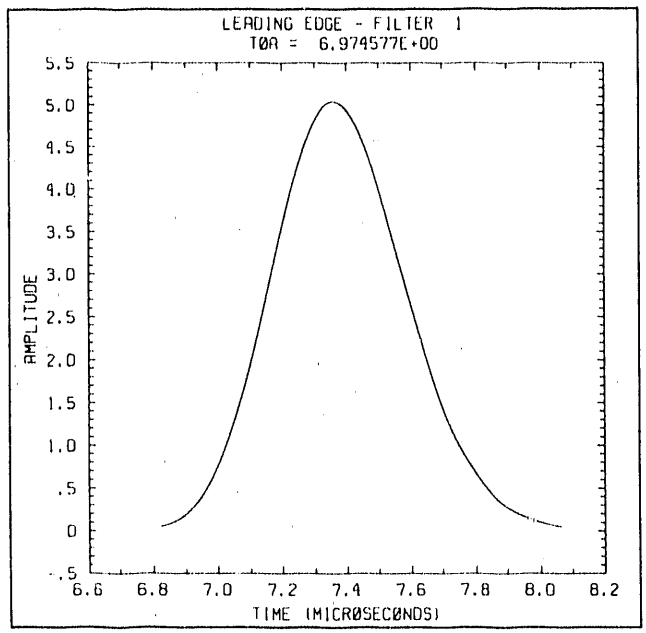

Figure 28a

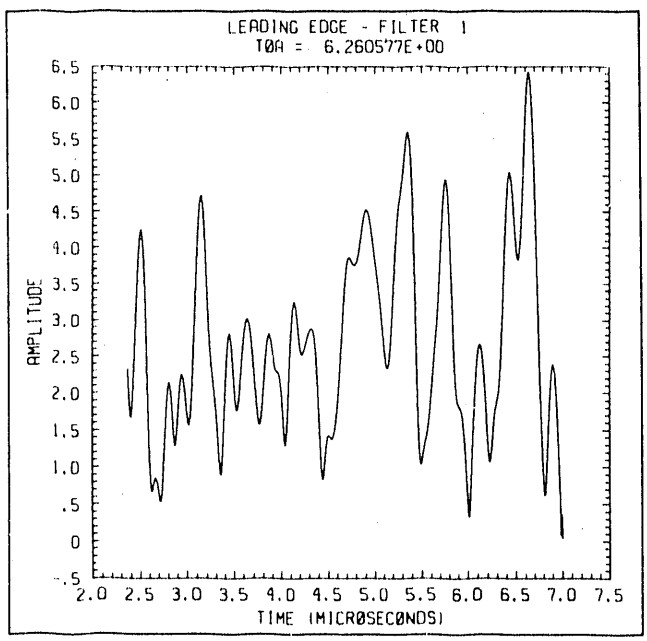

Figure 28b

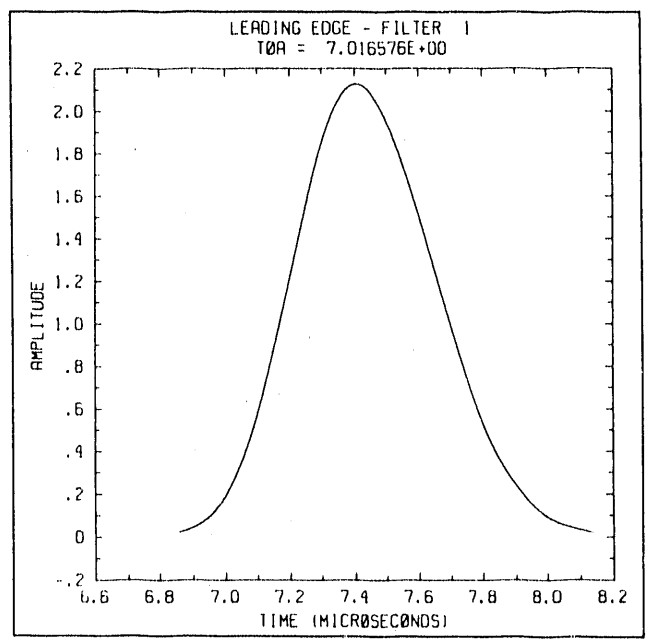

Figure 28c

Figures 28a–c Plot of the envelope of the received signal used by the leading-edge technique to time-tag the arrival of the signal in the wideband filter shown in Figure 24a. The default delta function, a TEC of 1×10^{13} cm^{-2} and the three sets of parameters (a) infinite coherence bandwidth and 70 dB SNR, (b) infinite coherence bandwidth and 30 dB SNR, and (c) 10 MHz coherence bandwidth and 70 dB SNR, are assumed for the three figures, respectively.

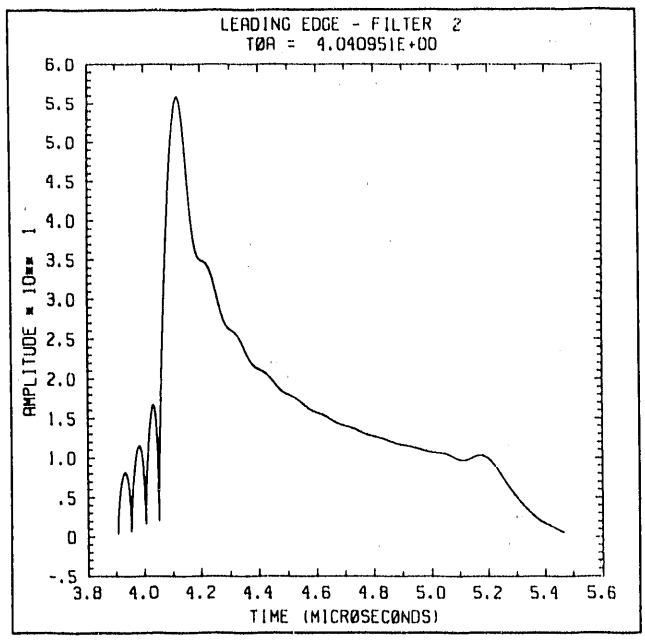

Figure 29a

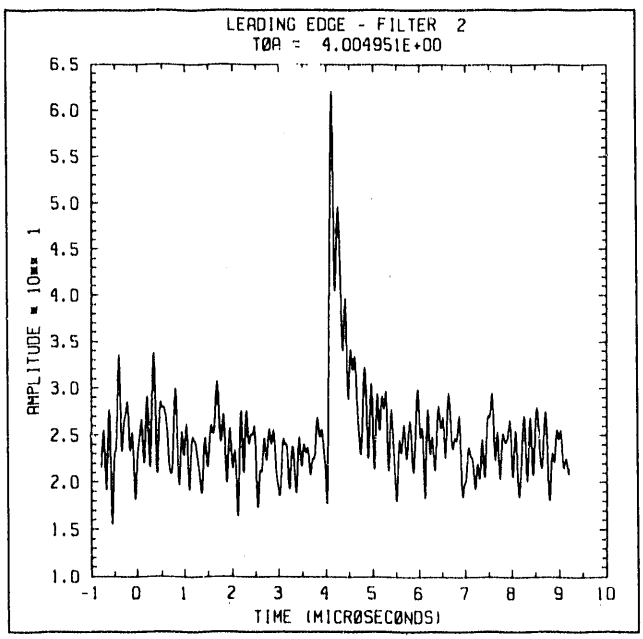

Figure 29b

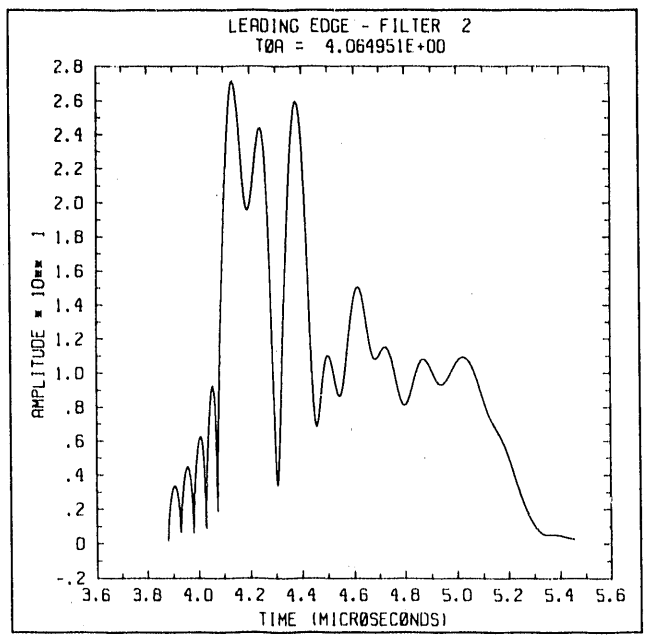

Figure 29c

Figures 29a–c Plot of the envelope of the received signal used by the leading-edge technique to time-tag the arrival of the signal in the narrowband filter shown in Figure 24b. The default delta function, a TEC of 1×10^{13} cm^{-2} and the three sets of parameters (a) infinite coherence bandwidth and 70 dB SNR, (b) infinite coherence bandwidth and 30 dB SNR, and (c) 10 MHz coherence bandwidth and 70 dB SNR, are assumed for the three figures, respectively.

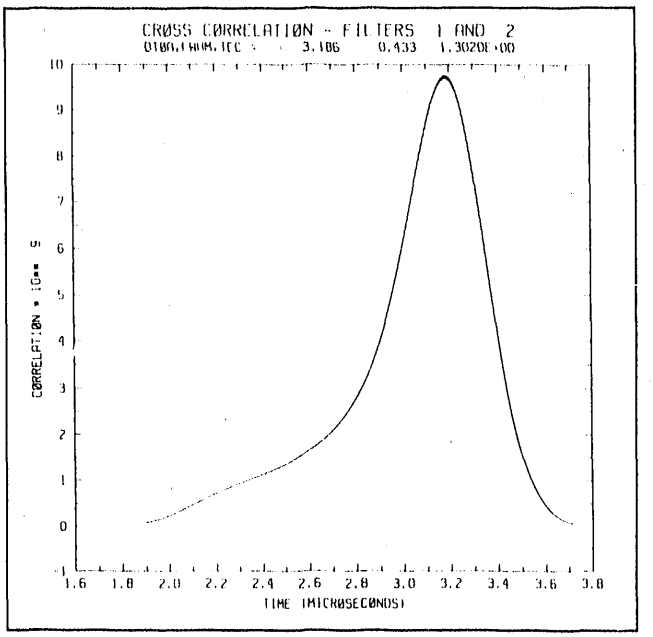

Figure 30a

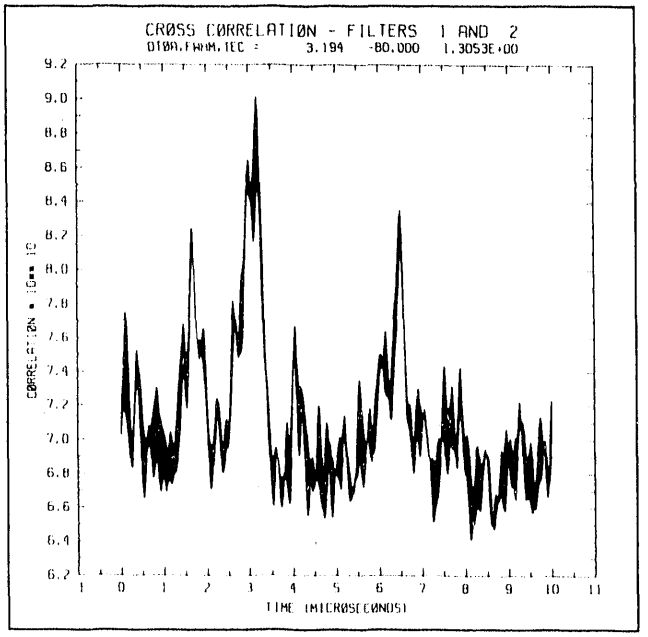

Figure 30b

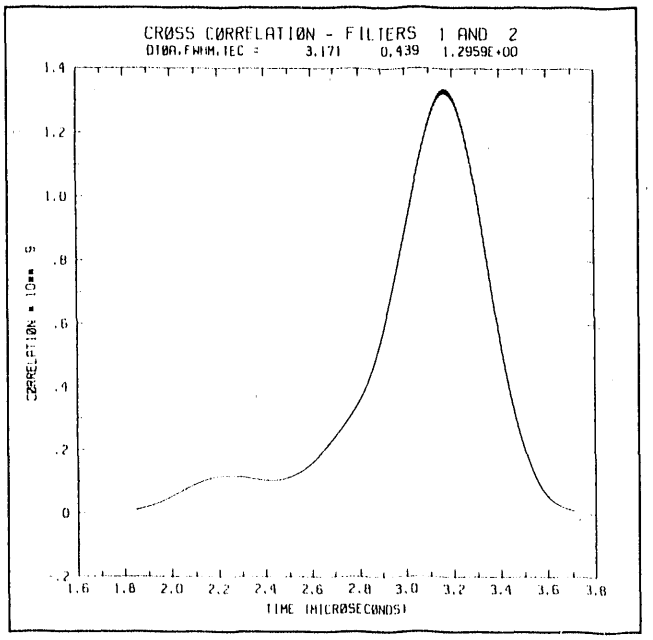

Figure 30c

Figures 30a–c Plots of the cross-correlations of the wideband and narrowband filters shown in Figures 24a and b as a function of the temporal shift (t) between receivers. The default delta function, a TEC of 1×10^{13} cm^{-2} and the three sets of parameters (a) infinite coherence bandwidth and 70 dB SNR, (b) infinite coherence bandwidth and 30 dB SNR, and (c) 10 MHz coherence bandwidth and 70 dB SNR, are assumed for the three figures, respectively.

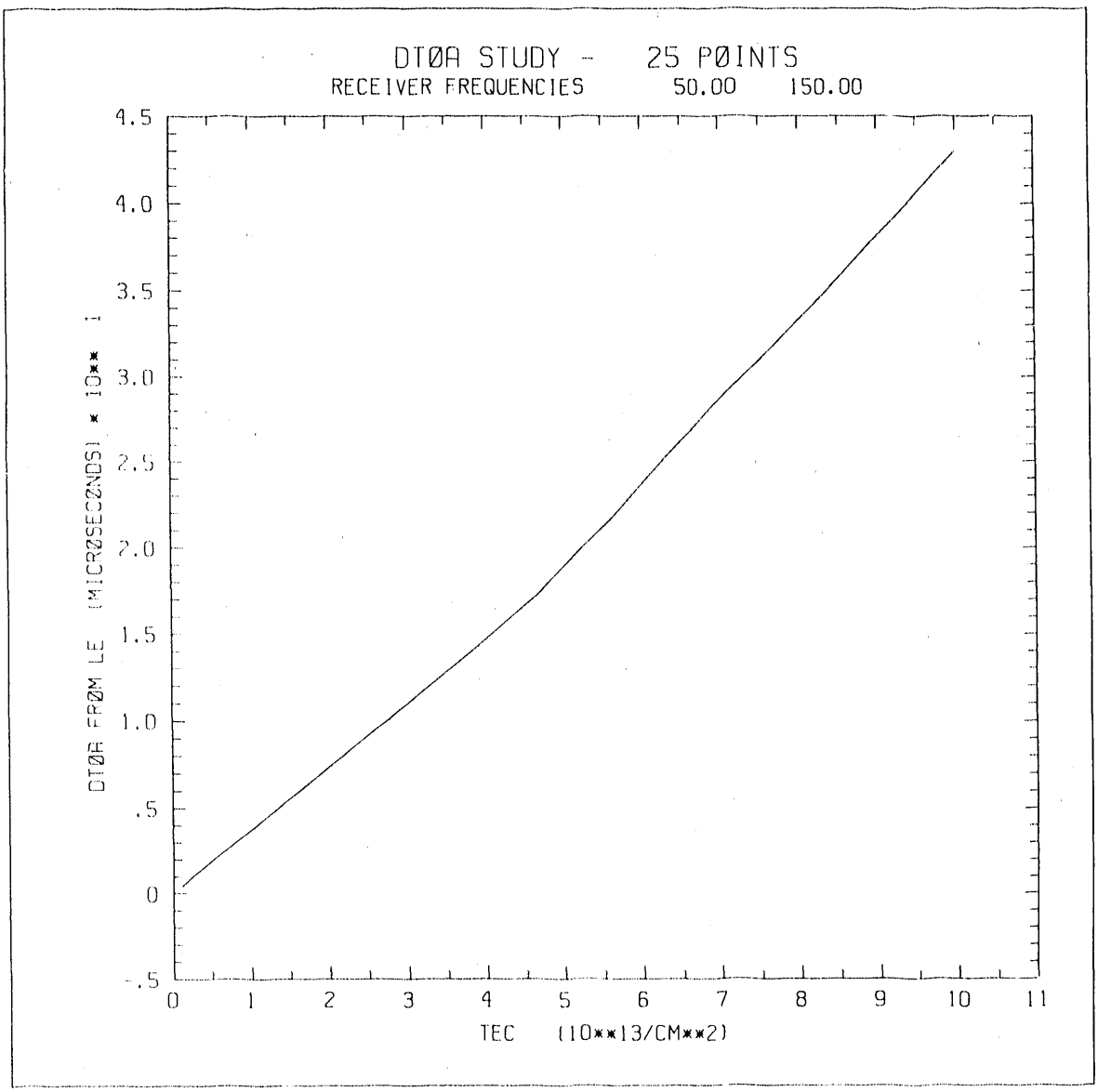

Figure 31 Plot of DTOA, obtained with the leading-edge algorithm for two narrowband receivers with central frequencies of 150 and 50 MHz and a FWHM of 6 MHz vs. TEC for 25 values ranging from 0.1–10 × 10^{13} cm^{-2}. The default delta function, $F_{coh} = \infty$, and SNR = 70 dB were used as input parameters.

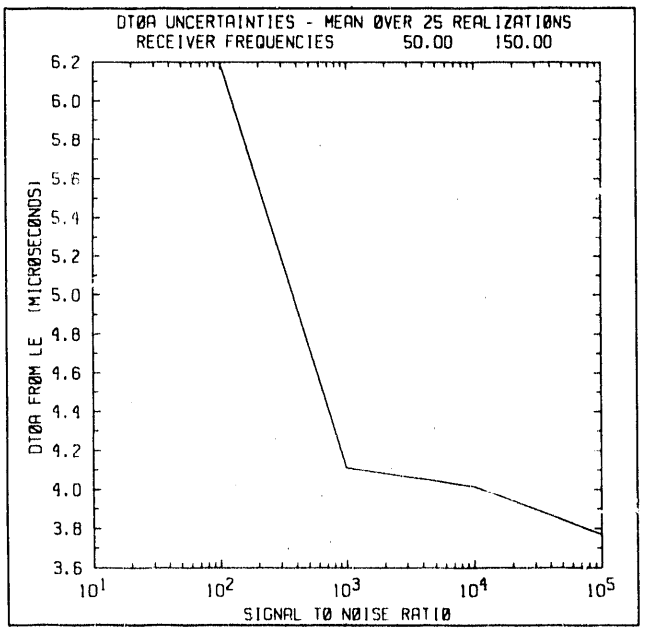

Figure 32a

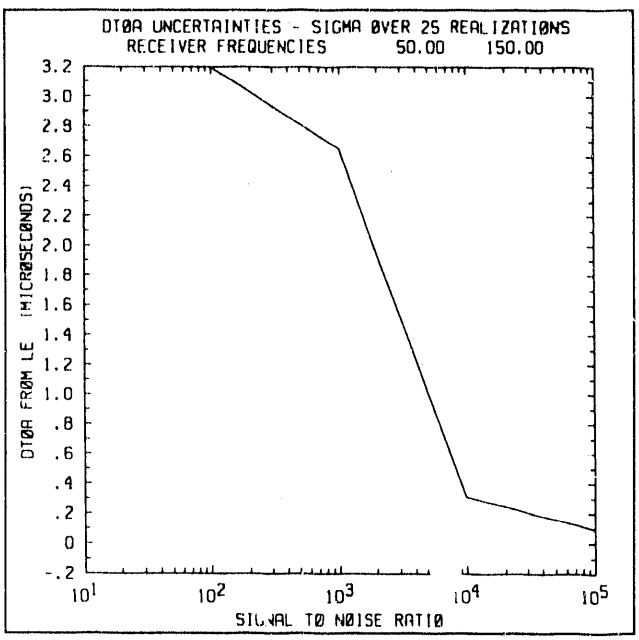

Figure 32b

Figure 32a–b Results of the DTOA uncertainty study for two narrowband receivers with central frequencies of 150 and 50 MHz and a FWHM of 6 MHz. The TEC was fixed at 1×10^{13} cm^{-2} and Gaussian white noise was added to the input signal. The resulting filtered transionospheric signal was time-tagged with the leading-edge algorithm for SNRs ranging from 102 to 105. (a) Mean DTOA and (b) standard deviation from the mean were calculated using 25 realizations of each SNR. The default delta function and $F_{coh} = \infty$ were used as input parameters.

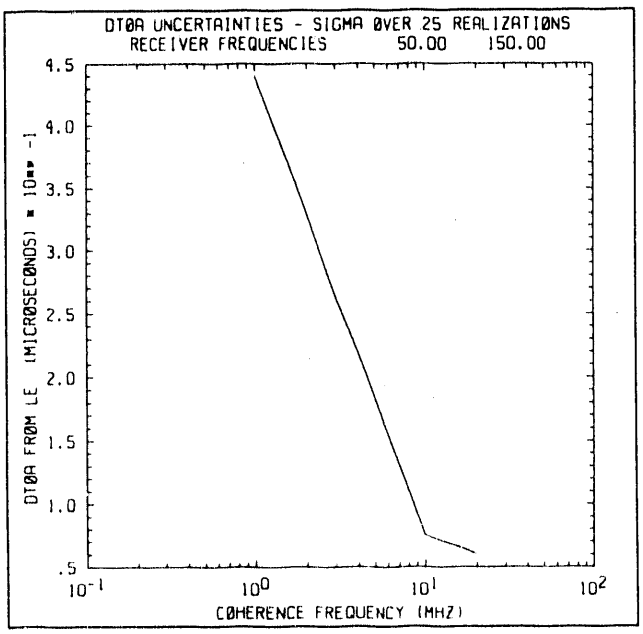

Figure 33a

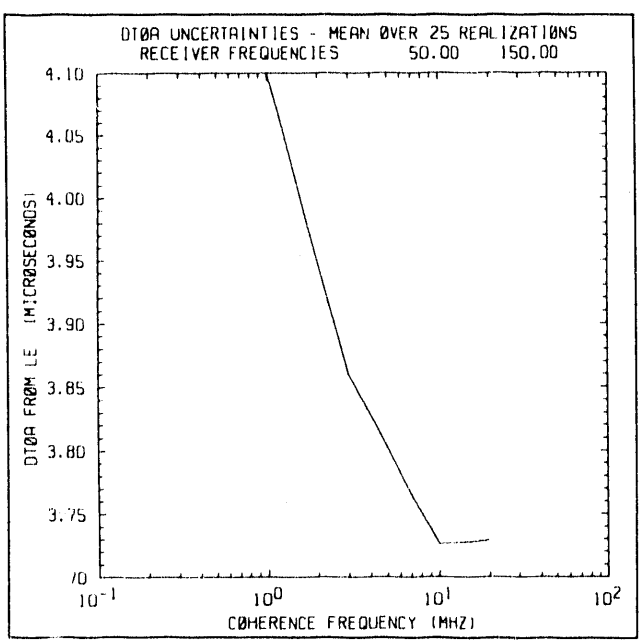

Figure 33b

Figure 33a-b Results of the DTOA uncertainty study for two narrowband receivers with central frequencies of 150 and 50 MHz and a FWHM of 6 MHz. The TEC was fixed at 1×10^{13} cm^{-2} and the effects of ionospheric scattering were included. The resulting filtered transionospheric signal was time-tagged with the leading-edge algorithm for F_{coh} ranging from 1 to 20 MHz. (a) Mean DTOA and (b) standard deviation from the mean were calculated using 25 realizations of each F_{coh}. The default delta function and an SNR of 70 dB were used as input parameters.

*U.S. GOVERNMENT PRINTING OFFICE 1990-0-573-036/20112

-END-

DATE FILMED

11/28/90