NASA Technical Memorandum 56045

A STATISTICAL PACKAGE FOR COMPUTING

TIME AND FREQUENCY DOMAIN ANALYSIS

James Brownlow

August 1978

NASA

1 Report No NASA TM-56045:	2 Government Accession No	3 Recipient's Catalog No
4 Title and Subtitle A STATISTICAL PACKAGE FOR COMPUTING TIME AND FREQUENCY DOMAIN ANALYSIS		5 Report Date August 1978
		6 Performing Organization Code H-981
7 Author(s) James Brownlow		8 Performing Organization Report No
		10 Work Unit No 992-01-00
9 Performing Organization Name and Address NASA Dryden Flight Research Center P O. Box 273 Edwards, California 93523		
		11 Contract or Grant No
		13 Type of Report and Period Covered Technical Memorandum
12 Sponsoring Agency Name and Address National Aeronautics and Space Administration Washington, D.C. 20546		
		14 Sponsoring Agency Code

15 Supplementary Notes

16 Abstract

    The spectrum analysis (SPA) program is a general purpose digital computer program designed to aid in data analysis. The program does time and frequency domain statistical analyses as well as some preanalysis data preparation. The capabilities of the SPA program include linear trend removal and/or digital filtering of data, plotting and/or listing of both filtered and unfiltered data, time domain statistical characterization of data, and frequency domain statistical characterization of data.

17 Key Words (Suggested by Author(s)) Time series analysis Spectrum analysis Digital filtering Fourier analysis	18 Distribution Statement Unclassified—Unlimited Category 65
19 Security Classif (of this report) Unclassified	20 Security Classif (of this page) Unclassified	21	22

NASA Technical Memorandum 56045

A STATISTICAL PACKAGE FOR COMPUTING

TIME AND FREQUENCY DOMAIN ANALYSIS

James Brownlow

Dryden Flight Research Center
Edwards, California

SYMBOLS

A	area of a histogram or fourier coefficient
AR_{XY}	amplitude ratio
ARG	principal angle of a complex number
a	real number
B	fourier coefficient
B_e	bandwidth of the estimate
B_w	half bandwidth for a bandpass digital filter
b	real number
C_{XY}	coincident spectrum
$C_X(\tau)$	autocovariance function
E_i	expected frequency in the ith class for χ^2-statistic
e	base of natural logarithms
F	fourier transform operator or F–statistic
F^{-1}	inverse fourier transform operator
f	frequency variable
f_c	center frequency for bandpass filter
f_N	Nyquist frequency
f_0	break frequency
f_1, f_2	values of the frequency variable
G_X	one-sided power spectrum of input X
G_{XY}	complex-valued cross spectrum

G_Y	one-sided power spectrum of output Y
$G(s)$	transfer function, S-plane
$G(z)$	transfer function, Z-plane
$H_{XY}(f)$	frequency response function, X input, Y output
$H(e^{i\omega})$	complex-valued transfer function
$H(z)$	transfer function
Im	imaginary part of a complex number
i	index, dummy variable, or $\sqrt{-1}$
j, k	indexes
m	dummy variable or mean square
m_1, m_2, m_3, m_4	first, second, third, and fourth central moments, respectively
n	number of samples, number of intervals, or order of a digital filter
$\binom{n}{j}$	combinations, n things taken j at a time
n_1, n_2	degrees of freedom for the F-test
O_i	observed frequency in the ith class for the χ^2-statistic
$P(X \leq x)$	probability the random variable X takes a value less than or equal to x
p_1, p_2, ..., p_n	poles of a transfer function
Q_{XY}	quadrature spectrum
$R_X(\tau)$	autocorrelation function
Re	real part of a complex number
$R(\tau)$	ensemble average of the $R_X(\tau)$
r	dummy variable

2

$\hat{r}$	confidence interval estimate
r_k	roots of a trigonometric polynomial
S	Laplace transform variable
s	standard deviation
s_X	sample standard deviation for random variable X
$s_X{}^2$	sample variance for random variable X
T	sampling period, or total length (in time) of a sample
TF_{XY}	transfer function gain, X input, Y output
t	time parameter, dummy variable for integration, or Student's t-test statistic
U(f)	fourier transform of u(t)
u	real variable
u(k)	discrete sinusoidal function
u(t)	boxcar function: $u(t) = \begin{cases} 1 & -T < t < T \\ 0 & \text{Otherwise} \end{cases}$
v	real variable
X	random variable or input signal
$\bar{X}$	sample mean value
X_t	input, at time t
$X_1, X_2, \ldots, X_n$	collection of random variables
X(f)	fourier transform of x(t)
X(z)	input as a function of z
x, y	real numbers
x_t	sampled (discrete) real-valued function of time
x(t)	continuous real-valued function of time

Y	output signal
$Y(f)$	fourier transform of $y(t)$
$Y(z)$	output as a function of z
y_t	output (discrete) at time t
$y(k)$	discrete output
$y(t)$	continuous, real-valued function of time
Z	Z-transform variable or standard normal random variable
z	complex variable
α	real number; coefficient of a digital filter or a probability value
β	real number or coefficient of a digital filter
Γ	gamma function: $\Gamma(x) = \int_0^\infty t^{x-1} e^{-t} dt$
$\gamma_{XY}{}^2$	coherence function
Δ	increment
Δt	time interval
$\delta(f - f_1 - f_2)$	delta function at $f - f_1 - f_2$
ε_r	normalized standard error for power spectrum
θ	phase angle
θ_{XY}	phase angle, X input, Y output
μ	population mean
μ_X	population mean for random variable X
μ_ω	mean of a frequency domain parameter at frequency ω
ν	degrees of freedom for χ^2- or F-statistic

4

σ	population standard deviation
σ^2	population variance
$\sigma_X{}^2$	variance for random variable X
$\sigma_\omega{}^2$	variance of a frequency-domain parameter at frequency ω
τ	lag variable for correlation functions; dummy variable for integration
$\varphi(f)$	phase angle
χ^2	test statistic
ω	frequency variable
ω_0	break frequency for low-pass filter
ω_1	break frequency for high-pass filter

Subscript:

0	initial term

An asterisk indicates a complex conjugate.

A superscripted caret indicates an estimated value.

A prime indicates the addition of a dummy variable.

INTRODUCTION

The statistical problems with which the spectrum analysis (SPA) program is concerned arise when data come in the form of a time series; that is, a collection of numerical observations made at equally spaced time intervals. These data can be analyzed in two ways. First, the time series can be examined for usual statistical properties, like mean, variance, form of the distribution the data come from, and so on. Such analysis is called time domain analysis; the time series is simply treated as a collection of independent observations. Secondly, the fact that the data were collected in order, at specified time intervals, can be used to examine the time series for periodicities. Further, more than one time series may be examined for possible interactions or cause-effect relationships.

In almost all practical applications, the data collected are actually the sum of the desired signal and noise. In most cases noise may be represented (or, at worst, approximated) as stationary, normal, and random with zero mean. The noise-plus-signal functions can be usefully studied in terms of the autocorrelation or power spectrum of this signal.

To determine the properties of a process exactly, however, requires data to be measured perfectly for an infinite time and computing equipment of infinite precision. Such requirements are, of course, impractical. As a result, the parameters sought (mean, power spectrum, or whatever) are estimated from finite-precision, finite-length records, and thus the results are actually estimates and their confidence intervals. Statistical estimation is what SPA is all about. The only difficulty is the extensiveness of the data required for highly precise estimates and the suitable interpretation of the estimated quantity.

A function of time $x(t)$ generated by some random, stationary process (stationary being the mean value and power spectrum independent of when a sample is taken) has a value which may be viewed as a random variable with a probability distribution. A given $x(t)$ may also be considered to be only one of an ensemble of random functions which might be generated by the stationary random process. A given $x(t)$ is assumed to be representative of the entire ensemble. The importance of normality for the distribution of the error signal comes, of course, from the statistics of confidence interval estimation. The usual statistical tests (t, x^2, and so forth) have at their foundation the normality assumption. Additional uncertainties in actual variability due to either nonnormality in the error portion of the signal or to changing conditions among runs (or both) can often be alleviated successfully by prior treatment of the data. Such procedures as detrending and digital filtering can serve to make the time series more stationary. The final precision of an overall result is more wisely judged by the observed consistency of repeated measurements (as by analysis of variance) than by any purely theoretical variability argument based on a normal distribution assumption.

As mentioned earlier, practical considerations in the estimate of power spectra can involve enhancing the stationarity of the data. The autocovariance function $R_X(\tau)$ defined at a lag value τ is given by the following equation for a zero mean process, $x(t)$:

6

$$R_X(\tau) = \lim_{T \to \infty} \frac{1}{T} \int_{-\frac{T}{2}}^{\frac{T}{2}} x(t) x(t + \tau) dt$$

In ensemble terms, $R(\tau)$ is the average of the $R_X(\tau)$ for each ensemble. The auto-correlation function is the normalized autocovariance function; namely, $R_X(\tau)/R_X(0)$. This autocorrelation function is used throughout this presentation. For a process with nonzero mean, the formula is given in the section entitled Program Use.

The autocovariance and power spectrum are fourier transforms of each other. The autocovariance function is

$$R_X(\tau) = \lim_{T \to \infty} \frac{1}{T} \int_{-\frac{T}{2}}^{\frac{T}{2}} x(t) x(t + \tau) dt$$

and the power spectrum, $G_X(f)$, is as follows:

$$G_X(f) = \lim_{T \to \infty} \frac{1}{T} \left| \int_{-\frac{T}{2}}^{\frac{T}{2}} x(t) e^{-i2\pi ft} dt \right|^2$$

$$= \lim_{T \to \infty} \int_{-\frac{T}{2}}^{\frac{T}{2}} x(t) e^{-i2\pi ft} dt \int_{-\frac{T}{2}}^{\frac{T}{2}} x(t) e^{i2\pi ft} dt$$

where the complex absolute value is a number times its conjugate. Then, by letting u equal $t - t'$ and v equal t',

$$G_X(f) = \lim_{T \to \infty} \int_0^T \left[\frac{1}{T} \int_{-\frac{T}{2}}^{\frac{T}{2}} x(v) x(v + u) dv \right] e^{-i2\pi fu} du + \int_{-T}^0 \left[\frac{1}{T} \int_{-\frac{T}{2} - u}^{\frac{T}{2}} x(v) x(v + u) dv \right] e^{-i2\pi fu} du$$

$$= \lim_{T \to \infty} \int_{-T}^T \frac{1}{2T} x(v) x(v + u) e^{-i2\pi fu} du$$

$$= \lim_{T \to \infty} \frac{1}{T} \int_{-\frac{T}{2}}^{\frac{T}{2}} R_X(\tau) e^{-i2\pi fu} du$$

At first glance selecting which of the estimates (power spectrum or autocorrelation) will be used appears to be purely a matter of convenience. In fact, when both quantities must be estimated, the use of the power spectrum is preferred, since the statistical effects of random noise (confidence intervals) and finite time intervals are better known.

The power spectrum

$$G_X(f) = \lim_{T \to \infty} \frac{1}{T} \left| \int_{-\frac{T}{2}}^{\frac{T}{2}} x(t) e^{-i2\pi ft} dt \right|^2$$

describes the contribution to variance of the original time series from f to $f + df$.

Notice now that if the time series examined violates the stationarity assumption by exhibiting trend (that is, the mean value is a function of time), the frequency spectrum will be contaminated by the summing into the spectrum of the original signal the additional frequency components necessary to resolve the ramp or trend in the time domain.

Digital filtering can serve to isolate frequency ranges and to examine the spectra of the filtered signal. But even low-pass filtering is not sufficient to eliminate the effects of trend completely; the frequency components above the break frequency (for the low-pass case) would remain.

The examination of a finite time length series amounts to looking at an infinite fourier transform of a truncated function. Consider the function x_T for a finite

length of time $\lim_{T \to \infty} \frac{1}{T} \left| \int_{-\frac{T}{2}}^{\frac{T}{2}} x(t) e^{-i2\pi ft} dt \right|^2$ which is equivalent to

$$\left| \int_{-\infty}^{\infty} x(t) u(t) e^{-i2\pi ft} dt \right|^2 \quad \text{where} \quad u(t) = \begin{cases} 1 & -\frac{T}{2} < t < \frac{T}{2} \\ 0 & \text{Otherwise} \end{cases}. \quad \text{We are, in point of}$$

fact, transforming a product of functions.

Let F be the fourier transform operator; then

$$F[x(t)u(t)] = F\{F^{-1}[X(f)]F^{-1}[U(f)]\}$$

where $X(f) = F[x(t)]$, $U(f) = F[u(t)]$. So we have

$$\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} X(f_1)U(f_2)\exp\left[-i2\pi(ft - f_1 t - f_2 t)\right]dt df_1 df_2 \quad \text{and hence}$$

$$F[x(t)u(t)] = \int_{-\infty,}^{\infty}\int_{-\infty}^{\infty} X(f_1)U(f_2)\int_{-\infty}^{\infty} \exp\left[-i2\pi(t - f_1 - f_2)\right]dt df_1 df_2$$

$$= \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} X(f_1)U(f_2)\delta(f - f_1 - f_2)df_1 df_2$$

where $\delta(f)$ is the impulse function, which is the fourier transform of $f(t) = 1$. Then

the above becomes $\int_{-\infty}^{\infty} X(f_1)\int_{-\infty}^{\infty} U(f_2)\delta(f - f_1 - f_2)df_1 df_2$ which equals

$$\int_{-\infty}^{\infty} X(f_1)U(f - f_1)df_1.$$

This is known as the convolution of X and U; hence, the product of our original signal and the boxcar function in the time domain result in the convolution of the fourier transform of the original signal with the fourier transform of the boxcar

function. If $u(t) = \begin{cases} 1 & -T < t < T \\ 0 & \text{Otherwise} \end{cases}$ then

$$U(f) = \int_{-\infty}^{\infty} u(t)e^{-i2\pi ft}dt$$

$$= \int_{-T}^{T} e^{-i2\pi ft}dt$$

$$= \frac{1}{-i2\pi f}e^{-i2\pi ft}\Big|_{t=-T}^{t=T}$$

$$= \frac{-1}{i2\pi f}\left[\cos 2\pi ft - i\sin 2\pi ft - \cos(-2\pi ft) + i\sin(-2\pi ft)\right]$$

$$= \frac{2\sin(2\pi fT)}{2\pi f}$$

9

Therefore U(f) resembles the sketch below.

U(f)
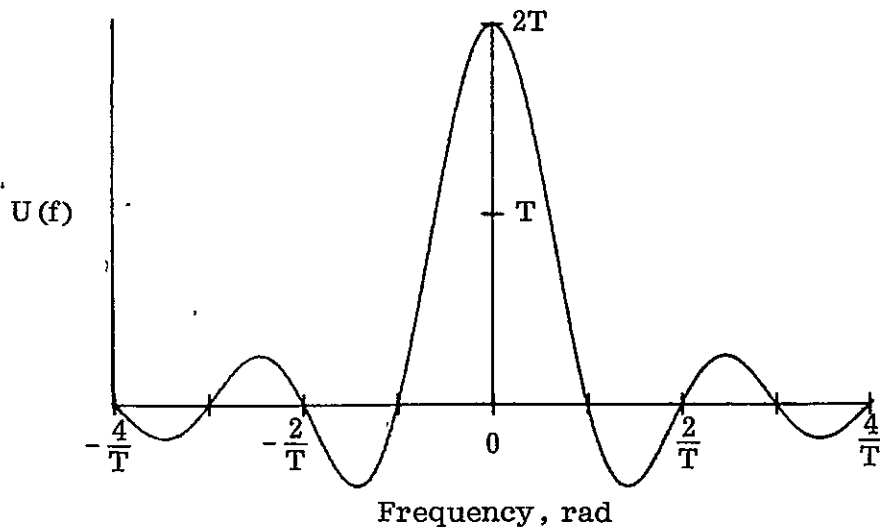
Frequency, rad

Hence we convolve the fourier transform of our original function with the above (sin x)/x function. The side lobes (beyond $\frac{1}{T}$ area) amount to smearing power from one frequency to adjacent frequencies. To alleviate this smearing, the data are windowed. This amounts to lowering the power spread to the side lobes at the expense of wider resolution.

U(f)
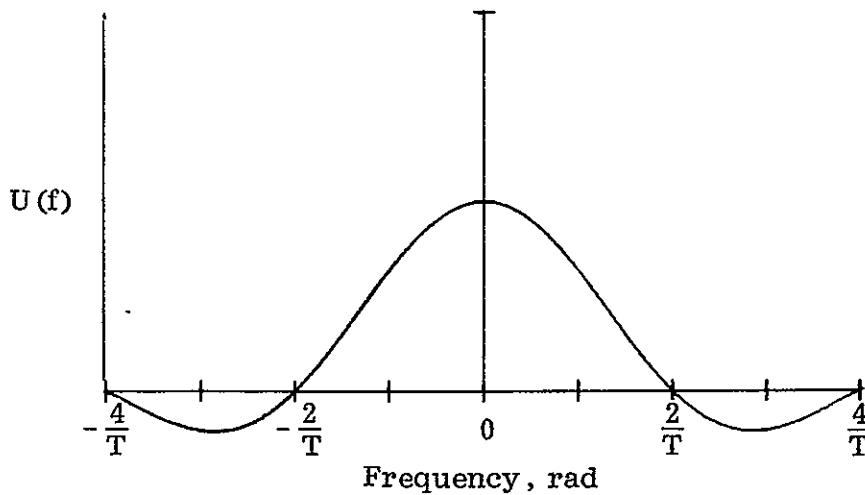
Frequency, rad

As a final note, in the SPA program records of arbitrary length may be analyzed. If more than 2048 points are to be analyzed, the program does the fourier transform in parts. At most, 1024 points will be put in the frequency domain. The resolution in the frequency domain will be samples per second divided by the number of points in the time domain, unless the number of points in the time domain is greater than 2048. In that case the SPA program will break up the record into parts and fourier transform each part. This is equivalent to frequency averaging an estimate to increase the number of degrees of freedom. The windows are then passed over the averaged spectrum. This will be dealt with more extensively in the following section.

10

Inputs to the SPA routine can be in either of two forms: data cards that indicate what type of analysis is to be performed on what parameters at what times, and actual data on a temporary or permanent file in a standard form (one expected by SPA).

Data Card Input

The data card input to the SPA program is described in detail in Program Use. Briefly, the input cards are: namelist $CARDIN, namelist $FLTRIN, namelist $STATIN, namelist $RMSIN, namelist $PSDIN, and namelist $CSIN.

Cards in all categories except namelist $CARDIN are optional, depending upon which tasks the SPA program is to accomplish. For example, if only power spectral densities (PSD's) are required, there could be no namelist $FLTRIN, $STATIN, or $RMSIN cards. Namelist $CARDIN directs the main program in selecting what tasks are to be performed.

Data Files

When data files are used, the data input to the SPA routine is discrete (sampled) and the sampling rate is assumed to be constant. The sampling rate must be at least twice the highest frequency present in the data.

The user must previously have stored his data on a file in a format expected by SPA. The user may use his own program for this purpose or one previously designed. An example of one of these programs, used in this case to read pulse code modulation (PCM) tapes, is shown in figure 1..

The SPA program then reads and takes data from this file (file 2) and puts it on a working file (file 4) for the SPA program to analyze. The data are unformatted. File 4 is a temporary file and is lost when the job terminates.

The data files managed by SPA are organized in the following manner (file 1 is reserved for input and file 3 is reserved for output). File 2 contains all data to be analyzed (multiple time intervals). File 4 contains a single time interval of data (used for processing by SPA). File 5 contains the same time interval as file 4, except that file 5 contains filtered data. All files are written unformatted.

File 2.—File 2 is created in the first job step. The file contains all the data to be analyzed; the data are organized by time interval. There may be as many time intervals as desired. File 2 is written as follows:

```
TITLE                                                     ⎫
NCH,SPS                                                    ⎪
CID(1),CID(2),...,CID(NCH)                                 ⎪
T1,T2,XMILLI,DATA(1),DATA(2),...,DATA(NCH)                 ⎪
T1,T2,XMILLI,DATA(1),DATA(2),...,DATA(NCH)                 ⎬  First time
              .                                           ⎪     interval
              .                                           ⎪
              .                                           ⎪
T1,T2,XMILLI,DATA(1),DATA(2),...,DATA(NCH)                 ⎪
EOF                                                       ⎪
ST(1),ST(2),ST(3),ST(4),ET(1),ET(2),ET(3),ET(4)          ⎭


TITLE                                                     ⎫
NCH,SPS                                                    ⎪
CID(1),CID(2),...,CID(NCH)                                 ⎪
T1,T2,XMILLI,DATA(1),DATA(2),...,DATA(NCH)                 ⎪
T1,T2,XMILLI,DATA(1),DATA(2),...,DATA(NCH)                 ⎬  Second time
              .                                           ⎪     interval
              .                                           ⎪
              .                                           ⎪
T1,T2,XMILLI,DATA(1),DATA(2),...,DATA(NCH)                 ⎪
EOF                                                       ⎪
ST(1),ST(2),ST(3),ST(4),ET(1),ET(2),ET(3),ET(4)          ⎭


TITLE                                                     ⎫
NCH,SPS                                                    ⎪
CID(1),CID(2),...,CID(NCH)                                 ⎪
T1,T2,XMILLI,DATA(1),DATA(2),...,DATA(NCH)                 ⎪
T1,T2,XMILLI,DATA(1),DATA(2),...,DATA(NCH)                 ⎪
              .                                           ⎬  Last time
              .                                           ⎪     interval
              .                                           ⎪
T1,T2,XMILLI,DATA(1),DATA(2),...,DATA(NCH)                 ⎪
EOF                                                       ⎪
ST(1),ST(2),ST(3),ST(4),ET(1),ET(2),ET(3),ET(4)          ⎪
EOF                                                       ⎭
```

where

TITLE	eight-word array containing title information for that interval
NCH	number of channels being processed for that interval; 10 or fewer are allowed
SPS	sampling rate (samples per second)
CID(I)	channel identification for the ith channel

T1,T2   time for those data in alphanumeric code; for example,

$$HR-MN-SC-MSC--------$$

$$\underbrace{}_{T1}\quad\underbrace{}_{T2}$$

XMILLI   time for those data in total milliseconds, as follows:

$$XMILLI = MSC + 1000(SC + 60(MN + 60HR))$$

DATA(I)   data value for the ith channel

ST   four-word array which contains the start time (hours, minutes, seconds, milliseconds) for that time interval

ET   four-word array which contains the end time (hours, minutes, seconds, milliseconds) for that time interval.

File 4 and file 5.—Files 4 and 5 are working files created in SPAMAIN from the master file (file 2). File 4 is made up of unfiltered data. File 5 is made up of filtered data. These files are used by SPA for actual processing and hence must be contiguous in time. The form for both files is

    T1,T2,XMILLI,DATA(1),DATA(2),...,DATA(NCH)
    T1,T2,XMILLI,DATA(1),DATA(2),...,DATA(NCH)
        .
        .
        .
    T1,T2,XMILLI,DATA(1),DATA(2),...,DATA(NCH)

where T1, T2, XMILLI, and DATA(I) are as in file 2. Figure 1 is the PCM front end (the job that creates a SPA file 2 from PCM input tapes).

OUTPUT

   The types and forms of output for the SPA program are many. Output is both listed (printed) and plotted. Each input card to the SPA program is printed so that the integrity of the input information can be verified.

   The first thing the SPA program does is to filter and/or detrend the input data. The filtered data may be listed (by specifying PRINT=T in namelist $FLTRIN). In addition, or as an alternative, both the filtered and unfiltered data may be plotted (on a common time axis) by specifying PLOTFL=T in namelist $CARDIN. The time domain statistical parameters that may be listed are mean value, variance, standard deviation, root mean square (rms), data maximum, data minimum, number of observations, skewness, and kurtosis.

   When a histogram is requested, a frequency histogram is produced and a normal curve with the same mean and variance as the data is fit to it. A statistic showing

13

```
      PROGRAM PCMIO (INPUT,OUTPUT,TAPE1=INPUT,TAPE2,PCM1=8,PCM2=8)
      COMMON/PCMCOM/KEY,NFILE,NCH,NFR,LFR,ST,ET,AST,AET,DAY,RATE,SYSTEM
      DIMENSION CID(10),TITLE(8),DATA(10,1)
      REAL MILLI(1)
      INTEGER ST(4),ET(4),AST(4),AET(4),CH(10),TV(1),TIME(2,1),EU(10),
     -  SYSTEM
      LOGICAL PRINTIT,ENDTIME
      NAMELIST /PCMIN/ ST,ET,NFILE,NCH,CH,PRINTIT,SPS,SYSTEM
      NFR = 1
    1 READ (1,101) TITLE
  101 FORMAT (8A10)
      IF (EOF(1).NE.0.) GO TO 999
      WRITE (2) TITLE
      READ PCMIN
      PRINT PCMIN
      WRITE (2) NCH,SPS
      READ 101, (CID(I),I=1,NCH)
      WRITE (2) (CID(I),I=1,NCH)
      ENDTIME = .FALSE.
      LINE = 0
      KEY = 0
      RATE = SPS
    2 CALL PCMOUT (DATA,CH,MILLI,TIME,TV,EU), RETURNS(6,7,8)
    3 IF (.NOT.PRINTIT) GO TO 5
      IF (MOD(LINE,55).NE.0) GO TO 4
      PRINT 102, TITLE
  102 FORMAT (1H1,8A10)
      PRINT 103
  103 FORMAT (/T15,"UNFILTERED DATA FROM PCM TAPE")
      PRINT 104, (CID(I),I=1,NCH)
  104 FORMAT (/T16,9(A10,2X),A10)
    4 PRINT 105, (TIME(J,1),J=1,2),(DATA(I,1),I=1,NCH)
  105 FORMAT (1X,A10,A2,10E12.5)
      LINE = LINE + 1
    5 WRITE (2) (TIME(J,1),J=1,2),MILLI(1),(DATA(I,1),I=1,NCH)
      IF (ENDTIME) GO TO 9
      GO TO 2
    6 ENDTIME = .TRUE.
      GO TO 3
    7 PRINT 106, ET,ST,NFILE
  106 FORMAT (" END TIME",4I4," BEFORE START TIME",4I4," IN FILE",I4)
      GO TO 999
    8 PRINT 107, ST,NFILE
  107 FORMAT (4I4," NOT IN FILE",I4," PASS ABORTED.")
      GO TO 999
    9 ENDFILE 2
      WRITE (2) AST,AET
      GO TO 1
  999 ENDFILE 2
      REWIND 2
      END
```

Figure 1. Program that reads PCM data and writes data in
standard SPA format.

how well the observed histogram fits a normal curve with the same mean and variance (the goodness of fit statistics) is produced and listed. This statistic and the time domain statistical parameters listed above are included in the histogram plot.

When output is printed (as opposed to plotted), the first 10 lag values of the auto-correlation values are listed in addition to the parameters mentioned above. (The autocorrelation value for a lag of 0 is, by definition, 1.)

If the full range of statistical parameters and histograms is not required, one may choose to print only the rms values and the variance by using the $RMSIN namelist. The data from which the root mean square is computed may also be listed. Both the statistical summary and the rms analysis may be done over any subinterval (or the entire interval) of the data; the data may be either filtered or unfiltered.

The spectrum analysis parameters may be printed or plotted. Here there is a wide range of choices. Power spectral densities may be of raw or filtered data (any time interval). The data may be smoothed by a Hamming, Hann-Tukey, Bartlett, or moving average smoothing window and may be frequency averaged by specifying additional degrees of freedom for each estimate. They may also be left unsmoothed and unaveraged, if desired. The number of points in the smoothing window (from three to 35) may be selected by the user. The same is true of cross-spectrum, transfer function, amplitude ratio, and phase angle estimates.

Since the coherence function is meaningful only for smoothed estimates, it will be smoothed automatically. The user still has the option of selecting the window to be used for smoothing and of increasing the number of degrees of freedom of the estimate by frequency averaging. This is done simply by specifying the minimum number of degrees of freedom desired.

The sections of this report entitled General Description of Spectrum Analysis Program Options and SPECTRUM ANALYSIS PROGRAM TECHNICAL DESCRIPTION provide more details about the above options.

PROGRAM USE

General Description of Spectrum Analysis Program Options

SPAMAIN.—The SPA program is driven by SPAMAIN. This routine reads the namelist $CARDIN with the parameters FILTER, DETREND, PLOTFL, STATIST, HGRAPH, RMS, and PSD. All these parameters are logical. DETREND=T if data are to be detrended; FILTER=T if data are to be digitally filtered; PLOTFL=T if filtered and unfiltered data are to be plotted against a common time axis; STATIST=T if a time domain statistical analysis of the data is to be made; HGRAPH=T if a histo-gram of the data is desired; and PSD=T if any spectrum analysis is to be done. All these logical variables guide SPAMAIN in selecting the subroutines to be called.

Notice that if PLOTFL=T, FILTER must also be specified as true, and that if HGRAPH=T, STATIST must also be specified as true. This routine also takes the user's data one set at a time from file 2 and puts it on the temporary working file (file 4) for processing.

Detrending.—The SPA program can detrend the data before digital filtering. This option is exercised by specifying DETREND=T in the $CARDIN namelist. This operation is recommended for three reasons. First, linear trends introduce frequency components across the entire spectrum, and high-pass digital filtering will remove only the low frequency components, leaving the high frequency components. Secondly, removing trend will remove the mean value from the data, thus eliminating a delta function at zero. Thirdly, spectrum analysis is predicated upon stationary data—data which have mean and autocorrelation functions that are not time dependent. Detrending will make the mean value less time dependent, thus enhancing the stationarity of the data.

Digital filtering.—The SPA program has Butterworth design digital filtering capabilities. It will high pass, low pass, or bandpass, and the filters available are order 1, order 3, and order 4. The order of the filter is the number of poles the filter transfer function has in the left half of the Z-plane. (The order will therefore determine the number of terms in the recursive filter.) An order 3 filter resembles the following equation:
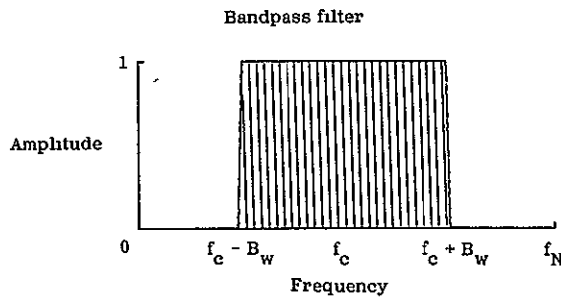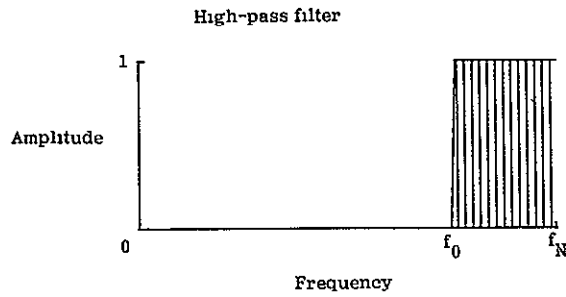
$$y_t = ax_t + \sum_{i=1}^{3} \alpha_i x_{t-i} + \beta_i y_{t-i} \qquad (1)$$

where

y_t output of filter at time t

x_t input to the filter at time t

y_{t-i} output of the filter at time t – i

x_{t-i} input to the filter at time t – i

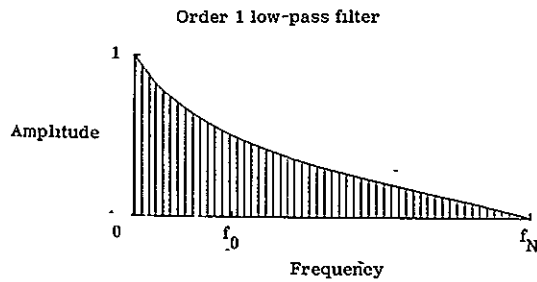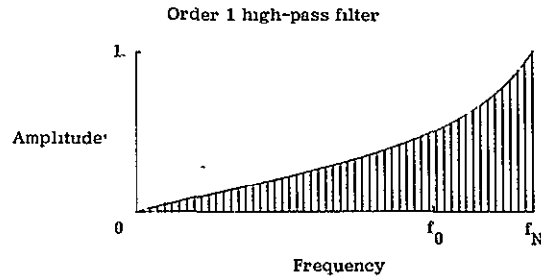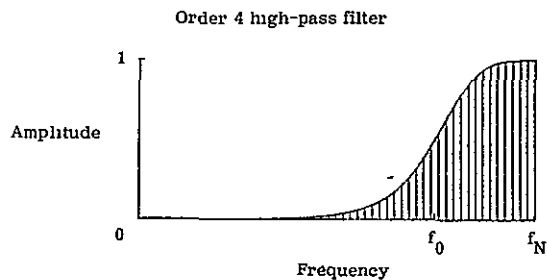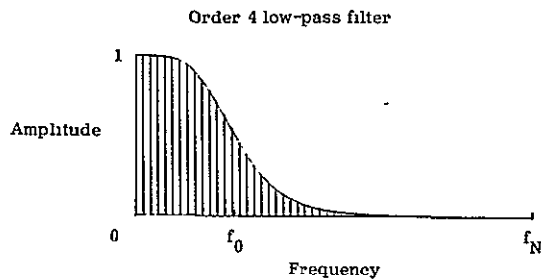The form of the transfer function of the ideal filters is as follows:

Low-pass filter

16

High-pass filter
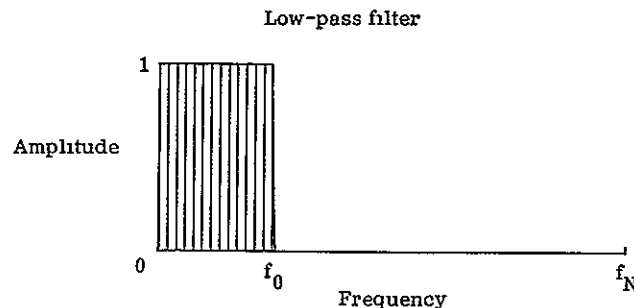
Bandpass filter

where

f_0 break frequency

f_N Nyquist frequency

f_c center frequency for bandpass filter

B_W half bandwidth for bandpass filter

The order of the digital filter determines how closely the actual filter transfer function approximates these ideal forms. In other words, the order 1 high-pass and low-pass digital filters have transfer functions which resemble

Order 1 low-pass filter

Order 1 high-pass filter

Amplitude

0   f_0 f_N

Frequency

and the order 4 digital filters have transfer functions which resemble

Order 4 low-pass filter

Amplitude

0   f_0 f_N

Frequency

Order 4 high-pass filter

Amplitude

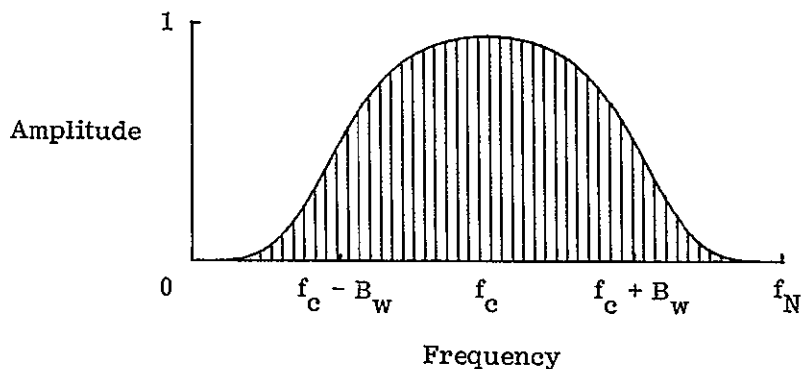0 f_0 f_N

Frequency

The higher-order filters have better frequency response characteristics, but they are more susceptible to stability problems; if the break frequency is set too close to zero or f_N, they require slightly more computer time, and they have a longer rise time.

The Butterworth filters are down 3 decibels at the break frequency, f_0. Changing the order only affects the filter rolloff. Stability problems are the result of finite precision arithmetic. The number held in core is a rational approximation of the true value of the coefficients α_i and β_i in equation (1). Hence, with a large number of recursive terms, the errors can propagate extensively.

The filter rise time is the time it takes the filter to begin outputting reasonable values. When the first data point is read in, the lag values $(x_{t-1}, x_{t-2}, \ldots, y_{t-1}, y_{t-2}, \ldots)$ are set to zero (for high-pass filters) or the first data point (for low-pass

18

filters), so the first few points filtered do not represent the time series being filtered accurately. The number of points the filter requires before it starts to produce acceptable results depends upon the filter order and the break frequency. In general, the higher the order and the closer the break frequency to zero or f_N, the greater the rise time.

The bandpass filter is a cascaded fourth-order high-pass and low-pass filter. The transfer function resembles

Amplitude — (vertical axis from 0 to 1)

Horizontal axis (Frequency): $f_c - B_W$, f_c, $f_c + B_W$, f_N

Frequency

The rise time for the bandpass filter is comparable to the rise time for the fourth-order high-pass or low-pass filters; it also depends on how close $f_c + BW$ is to f_N and how close $f_c - BW$ is to zero.

Data plotting routine.—The data plotting routine plots filtered and unfiltered data on a common time axis. The routine is called by setting PLOTFL=T in the $CARDIN namelist. There is no namelist for this routine. If this option is to be exercised, FILTER=T must also be specified in $CARDIN and there must be a namelist card $FLTRIN specifying how the data are to be filtered.

Statistical analysis routines.—The statistical analysis routines are divided into three sections: rms analysis, general statistical analysis, and histograms.

The rms analysis section produces only rms and variance values for either filtered or unfiltered data for any time interval. To obtain rms and variance values, one must specify RMS=T in $CARDIN and include a $RMSIN namelist. The time interval specified in this $RMSIN namelist card specifies the time period for which the rms and variance values will be computed. For rms and variance values of unfiltered data, one specifies ND=4 on this card; for filtered data, ND=5.

The general statistical analysis includes mean, mean square, rms, variance, standard deviation, skewness, kurtosis, data editing, and the first 10 lag values for the autocorrelation function. To obtain these parameters one must specify STATIST=T in the namelist $CARDIN and include a $STATIN namelist. In $STATIN one specifies the time interval desired and whether the data are filtered or unfiltered. One may also specify maximum and minimum values for data editing. Values beyond the minimum and maximum are not included in the analysis, and no substitution for these points is made.

Histograms of the data are obtained by specifying HGRAPH=T in namelist $CARDIN. The histograms of the data are fitted with a normal curve (based on data mean and variance). In addition, when plotting histograms the SPA program will divide the range of the data into 20 intervals. More or fewer intervals may be specified in $STATIN. From one to 40 intervals may be requested for the histograms.

The goodness of the fit of the normal curve to the histogram (the χ^2-statistic) is printed above the histogram, along with the parameter's mean, mean square, root mean square, variance, standard deviation, data maximum and minimum, skewness, and kurtosis.

Spectrum analysis routines.—The spectrum analysis routines are divided into two portions: autospectrum and cross spectrum. The autospectrum (sometimes called the power spectrum or power spectral density) sections produce only auto-spectra. The cross-spectrum section produces cross spectral density, phase angle, transfer function, amplitude ratio, and coherence functions. These functions can be plotted, listed, or both. The spectrum routines are called by specifying PSD=T in namelist $CARDIN and by including $PSDIN namelist cards (for autospectra) or $CSIN namelist cards (for cross spectra). In these namelist cards one specifies the time interval desired, the type of data desired (ND=4 for unfiltered data, ND=5 for filtered data) and the parameters desired. The spectrum estimates may be plotted with linear log or decibel scales and smoothed with Hamming, Hann-Tukey, Bartlett, or moving average smoothing windows. For more details on spectrum functions, see the section entitled SPECTRUM ANALYSIS PROGRAM TECHNICAL DESCRIPTION.

Job Card Setup

The job card setup below is for two-step SPA execution using the PCM example shown in figure 1.

```
PCMSPA,NT1,SSSS,FTN.
ATTACH(PCMFRNT,ID=SPA,MR=1)
REQUEST (PCMSj,NT,MF,PE,E,VSN=XXXX)
LABEL(PCMj,R,M=PCMSj,L=ZZZZ) RING OUT
FILE(PCMj,BT=C,RT=W,EO=AD,MBL=5120,MRL=5110)
LDSET(FILES=PCMj)
LOAD(PCMFRNT)
EXECUTE.
ATTACH(SPALIB,ID=SPA,MR=1,PW=RD)
ATTACH(SPADIR,ID=SPA,MR=1,PW=RD)
UNLOAD(PCMSj)
RFL(120000)
LDSET(LIB=SPALIB)
SEGLOAD(I=SPADIR)
EXECUTE.
REWIND(TAPE13)
RETURN(TAPE2,TAPE4,TAPE5)
LABEL(SPAPLT,W,L=SPAPLOTTAPE,D=HD,VSN=YYY) RING IN
COPY(TAPE13,SPAPLT)
7/8/9
[Enter PCM setup cards]
7/8/9
[Enter SPA setup cards]
6/7/8/9
```

where

SSSS   subtask number

XXXX   volume serial number

j   PCM system number

ZZZZ   PCM tape label

YYYY   volume serial number

Data Card Setup

The data setup for the SPA program consists of six sections: namelist $CARDIN, namelist $FLTRIN, namelist $STATIN, namelist $RMSIN, namelist $PSDIN, and namelist $CSIN.

Namelist $CARDIN.—Namelist $CARDIN is the input card to the SPA program that allows the user to select options.

The namelist name must begin with the $ in column 2 of the input card. The $ must be in column 2 for all namelists in SPA.

The parameters in $CARDIN are DETREND, FILTER, STATIST, RMS, PSD, PLOTFL, and HGRAPH. All variables are logical, and all have default value false.

DETREND is a logical variable used for decision on the call to the data detrending routine. DETREND=T will cause linear trends to be removed from the data written in file 4 (unfiltered data). DETREND=F will cause the data on file 4 not to be detrended. The detrending is done before any filtering is done. Hence, if the data are digitally filtered by SPA, the data filtered will be either detrended or not detrended, depending on the variable DETREND.

FILTER is a logical variable used for decision on the call to the digital filtering routine. If FILTER=T, the data on file 4 will be digitally filtered; if FILTER=F, the data will not be filtered. If FILTER=T, namelist $FLTRIN must be in the setup.

STATIST is a logical variable used for decision on the call to the statistical analysis routine. If STATIST=T, either the filtered or the unfiltered data (or both) may be analyzed, depending on what is specified in the namelist card $STATIN. If STATIST=T, there must be at least one $STATIN namelist, and there may be more.

RMS is a logical variable used for decision on the call to the rms analysis routine. If RMS=T, the rms analysis routine will be called; if RMS=F it will not be called. If RMS=T, there must be at least one $RMSIN namelist (there may be more). Either filtered data or unfiltered data (or both) may be analyzed.

PSD is a logical variable used for decision on the call to the spectrum analysis routines. If PSD=T, the spectrum analysis routines will be called; if PSD=F,

21

spectrum analysis will not be called. If PSD=T, there must be at least one $PSDIN namelist card and at least one $CSIN namelist card.

PLOTFL is a logical variable used for decision on the call to the routine that plots the filtered and unfiltered data on a common time axis. If PLOTFL=T, a plot for each channel of the data will be made (each SPA channel, file 4, the working file). If PLOTFL=T, FILTER=T must be specified.

HGRAPH is a logical variable used for decision on the call to the routine which plots a histogram of data (either filtered or unfiltered). If HGRAPH=T, histograms will be produced for each STATIN card as well as a statistical summary printout. If HGRAPH=T, STATIST=T must also be specified.

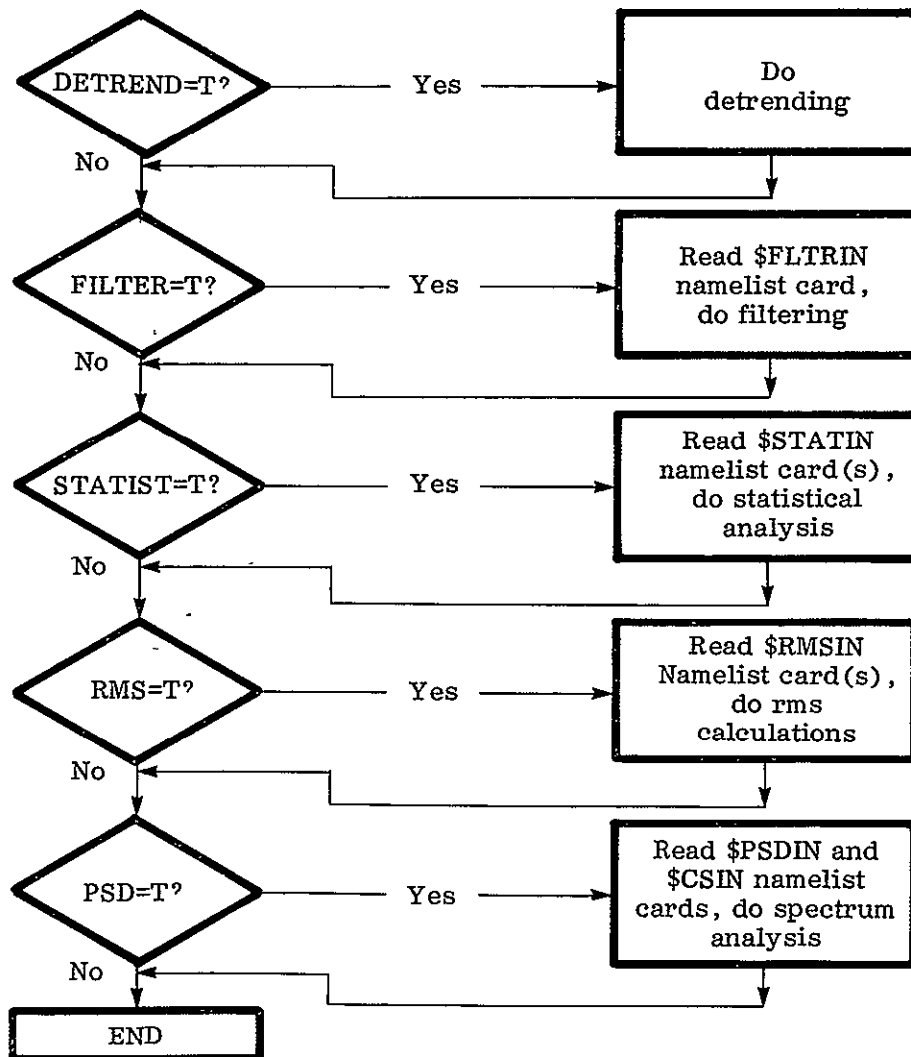The organization of the namelist SPA input, based on parameters in the $CARDIN namelist, is shown in figure 2.

Figure 2. $SCARDIN namelist.

Namelist $FLTRIN.—If FILTER=T is specified in namelist $CARDIN, there must be one namelist $FLTRIN card. The parameters in $FLTRIN are HIPASS, LOPASS, BANDPAS, ORDER, BREAK, FC, BW, and PRINT. ORDER is an integer, and BREAK, FC, and BW are real numbers; all the remaining variables are logical.

HIPASS is a logical variable. HIPASS=T yields high-pass filtering; HIPASS=F indicates that high-pass filtering is not desired.

LOPASS is a logical variable. LOPASS=T yields low-pass filtering; LOPASS=F indicates that low-pass filtering is not desired.

BANDPAS is a logical variable. BANDPAS=T yields bandpass filtering; BANDPAS=F indicates that bandpass filtering is not desired.

ORDER is an integer that indicates the order of the filter for high-pass or low-pass filtering. First-, third-, or fourth-order digital filters may be employed. For bandpass filtering the program automatically selects fourth order, no matter what value ORDER has.

BREAK is a real number that indicates which break frequency, in hertz, is to be used whenever high-pass or low-pass filtering is selected.

FC is a real number that indicates the center frequency of the band to be passed if bandpass filtering is selected.

BW is a real number that indicates the half bandwidth of the bandpass filter. The frequencies passed by the bandpass filter are those in the interval FC – BW to FC + BW.

PRINT is a logical variable. PRINT=T implies that filtered data will be listed. PRINT=F suppresses such output.

The default values for $FLTRIN are HIPASS=T, BANDPAS=F, ORDER=3, LOPASS=F, PRINT=F, and BREAK=3. FC and BW have no default values and must be provided.

Namelist $STATIN.—If STATIST=T is specified in namelist $CARDIN, there must be at least one namelist $STATIN card; there may be as many of these cards as desired. The last $STATIN card, however, must have the parameter STOP=T specified. The parameters in $STATIN are ST, ET, ND, TSTVAL, INTRVL, and STOP. All are integers, except for TSTVAL, which is a real number, and STOP, which is a logical variable.

ST is a four-word integer array containing the start time in hours, minutes, seconds, and milliseconds for statistical analysis.

ET is a four-word integer array containing the end time for statistical analysis.

ND is an integer specifying whether filtered or unfiltered data will be analyzed. ND=4 implies that the data to be analyzed will be taken from file 4, which contains

the unfiltered data. ND=5 implies that the data to be analyzed will be taken from file 5, which contains the filtered data.

TSTVAL is a real array with dimensions of 2 X 10. With TSTVAL, the user may censor data. TSTVAL(1,1) and TSTVAL(2,1) specify minimum and maximum values for data on the first parameter (that is, data outside the range from TSTVAL(1,1) to TSTVAL(2,1) will be censored). Similarly, TSTVAL(1,2) and TSTVAL(2,2) specify minimum and maximum values for data on the second parameter, and so forth. If no values are specified for TSTVAL, the program will do no data censoring.

INTRVL is an integer array with dimension of 10, which specifies the number of intervals the data range will be broken into for the histogram. INTRVL(1) is the number of intervals for the first parameter, INTRVL(2) is the number of intervals for the second parameter, and so forth. The default value for each INTRVL array member is 20.

STOP is a logical variable used to indicate that the last $STATIN card has been read. Thus, STOP is omitted from each $STATIN card except the last one, where STOP=T must be specified.

The default values for namelist $STATIN are ND=5; STOP=F; TSTVAL(I,J)=0, I=1,2, J=1,2,...,10; INTRVL(I)=20, I=1,2,...,10. Once again, setting all TSTVAL values at zero implies that data censoring will not take place.

Namelist $RMSIN.—If RMS=T was specified in namelist $CARDIN, there must be at least one namelist $RMSIN card; there may be as many $RMSIN cards as desired. The last card must specify STOP=T. The parameters in $RMSIN are ST, ET, ND, PRINT, and STOP. ST, ET, and ND are integers; PRINT and STOP are logical variables.

ST is a four-word integer array that specifies the start time in hours, minutes, seconds, and milliseconds for rms analysis.

ET is a four-word integer array that specifies the end time for rms analysis.

ND is an integer disk number specifying whether filtered or unfiltered data will be analyzed. ND=4 implies that the data for rms analysis will be taken from file 4 (the unfiltered data). ND=5 implies that the data for rms analysis will be taken from file 5 (the filtered data).

PRINT is a logical variable. PRINT=T will cause the data analyzed for rms value to be printed out. PRINT=F will suppress such output.

STOP is a logical variable used to indicate that the last $RMSIN card has been read. STOP=T should be specified only in the last $RMSIN namelist card.

The default values for $RMSIN are STOP=T, ND=4, and PRINT=F.

Namelist $PSDIN.—If PSD=T was specified in namelist $CARDIN, there must be at least one $PSDIN and at least one $CSIN card. All the $PSDIN cards precede all the

$CSIN cards. The last $PSDIN card must be $PSDIN STOP=T $, and the last $CSIN card must be $CSIN STOP=T $. The parameters in $PSDIN are ST, ET, XTYPE, YTYPE, ISMO, NW, PSDLSTR, PSDLSTS, PSDPLTR, PSDPLTS, CH, ND, DOF, YMIN, YMAX, and STOP. YMIN and YMAX are real numbers, PSDLSTR, PSDLSTS, PSDPLTR, PSDPLTS, and STOP are all logical, with default value FALSE; and the rest are integers.

ST is a four-word integer array containing the start time in hours, minutes, seconds, and milliseconds for PSD analysis.

ET is a four-word integer array containing the end time for PSD analysis.

XTYPE is an integer variable determining the type of X-axis (the frequency axis) on the plotted output. XTYPE=1 implies that the frequency axis will be linear and XTYPE=2 that the frequency axis will be logarithmic.

YTYPE is an integer variable determining the type of Y-axis (power) for auto-spectrum, cross-spectrum, amplitude ratio, and transfer function plots. YTYPE=1 implies that the Y-axis will be linear, YTYPE=2 that the Y-axis will be logarithmic, and YTYPE=3 that the Y-axis will be in decibels.

ISMO is an integer variable determining the type of smoothing window on the frequency domain plots. ISMO=1 will result in the frequency domain estimate being smoothed with the Hann-Tukey window, ISMO=2 will result in the smoothing being done by the Bartlett window, ISMO=3 will result in the smoothing being done by the moving average window, and ISMO=4 will result in the smoothing being done by the Hamming window.

NW is an integer indicating the number of points in the smoothing window. In general, the greater the number of points in the window, the smoother the frequency domain estimate will be (see Smoothing Windows). NW must be odd and may take on from three to 35 values, except for the Hamming window, which may have from three to seven points in it.

Setting the following four variables equal to true, T, will cause the specified output to be produced.

PSDLSTR is a logical variable determining whether or not the raw estimate of the power spectrum will be printed. Here raw means unsmoothed by any of the various windows.

PSDLSTS is a logical variable determining whether or not the smoothed power-spectrum estimate will be listed.

PSDPLTR is a logical variable determining whether or not the raw power-spectrum estimate will be plotted.

PSDPLTS is a logical variable determining whether or not the smoothed power-spectrum estimate will be plotted.

CH is a parameter indicating the channel of data to be analyzed. Since this is an SPA parameter number, CH will be an integer value from 1 to 10. CH=1 will cause the first parameter to be analyzed, and so forth.

ND is an integer variable specifying whether filtered or unfiltered data will be analyzed. ND=4 will cause unfiltered data to be analyzed; ND=5 will cause the digitally filtered data to be analyzed.

DOF is an integer variable indicating the minimum number of degrees of freedom of the estimate desired. DOF relates to bandwidth and number of points analyzed. With n data points, resolution will be SPS/n (where SPS is sampling rate, in samples per seconds), and there will be two degrees of freedom. To increase DOF, one must decrease resolution bandwidth. If SPS/n is greater than 1024, SPA will do some of the bandwidth decreasing (DOF increasing) automatically. There are practical limitations on DOF, of course; if a request is made for more degrees of freedom than can be derived, the program will yield the maximum number of degrees of freedom that is practical.

YMIN is a real variable that indicates the minimum value for the power axis.

YMAX is a real variable that indicates the maximum value for the power axis. By using YMIN and YMAX one can cause all the power-spectrum plots to have identical scaling, making them easily comparable. Values on the plot that exceed the range (YMIN, YMAX) are redefined to the limiting value.

STOP is a logical variable used to terminate power-spectrum analysis. The last $PSDIN card must be in the form $PSDIN STOP=T, $.

Namelist $CSIN.—All $PSDIN namelist cards must precede the $CSIN namelist cards. The parameters in $CSIN are ST, ET, XTYPE, YTYPE, ISMO, NW, ND, CSDLSTR, CSDPLTR, CSDLSTS, CSDPLTS, PHLSTR, PHLSTS, PHPLTR, PHPLTS, COHERL, COHERP, TFLSTR, TFLSTS, TFPLTR, TFPLTS, ARLSTR, ARLSTS, ARPLTR, ARPLTS, YMIN, YMAX, DOF, CHIN, CHOUT, and STOP.

ST and ET are used exactly as in namelist $PSDIN. They specify the start and stop times for cross-spectrum analysis. XTYPE, YTYPE, ISMO, NW, ND, YMIN, YMAX, and DOF are also exactly as in $PSDIN. The exceptions are that phase angle and coherence have fixed axes. The phase angle axis runs from -180 to 180; the coherence axis runs from 0 to 1. The phase angle and coherence axes are always linear; specifying YTYPE does not affect this. Setting all of the following variables equal to true, T, will cause the specified output to be produced.

CSDLSTR is a logical variable determining whether or not the unsmoothed cross-spectrum estimate will be listed.

CSDPLTR is a logical variable determining whether or not the unsmoothed cross-spectrum estimate will be plotted.

CSDLSTS is a logical variable determining whether or not the smoothed cross-spectrum estimate will be listed.

CSDPLTS is a logical variable determining whether or not the smoothed cross-spectrum estimate will be plotted.

PHLSTR is a logical variable determining whether or not the unsmoothed phase angle estimate will be listed.

PHLSTS is a logical variable determining whether or not the smoothed phase angle estimate will be listed.

PHPLTR is a logical variable determining whether or not the unsmoothed phase angle estimate will be plotted.

PHPLTS is a logical variable determining whether or not the smoothed phase angle estimate will be plotted.

COHERL is a logical variable determining whether or not the coherence function will be listed.

COHERP is a logical variable determining whether or not the coherence function will be plotted.

TFLSTR is a logical variable determining whether or not the unsmoothed transfer function estimate will be listed.

TFLSTS is a logical variable determining whether or not the smoothed transfer function estimate will be listed.

TFPLTR is a logical variable determining whether or not the unsmoothed transfer function estimate will be plotted.

TFPLTS is a logical variable determining whether or not the smoothed transfer function estimate will be plotted.

ARLSTR is a logical variable determining whether or not the unsmoothed amplitude ratio estimate will be listed.

ARLSTS is a logical variable determining whether or not the smoothed amplitude ratio estimate will be listed.

ARPLTR is a logical variable determining whether or not the unsmoothed amplitude ratio estimate will be plotted.

ARPLTS is a logical variable determining whether or not the smoothed amplitude ratio estimate will be plotted.

CHIN is the channel number (similar to CH in $PSDIN) for input data. CHOUT is the channel number for output data.

The last $CSIN card must be $CSIN STOP=T , $.

SPECTRUM ANALYSIS PROGRAM ERROR PROCEDURES

Computational Procedures

In general, the SPA program is rather weak in terms of pointing out the location of error interrupts to processing. Almost no checks of denominator (prior to division) are made: When a processing interrupt occurs, the best approach is to examine the loader map to determine what routine the interrupt occurred in.

Statistical Domain Computations

The error messages described below are associated with statistical domain computations.

If, when specifying the Y-axis (the statistical parameter) MAX and MIN values, the user specifies a MAX less than the specified minimum, the MAX/MIN values will be ignored. The message YMAX IS LESS THAN YMIN – SO THEY WILL BE IGNORED is produced, and processing continues.

If the specified start time is later than the specified end time for a given interval, an appropriate error message is produced, and processing is terminated.

If the number of points to be analyzed is 0 (or if start time equals end time), an appropriate error message is produced, and processing is terminated.

In the computation of the weights for smoothing windows, if an undefined operation occurs in combinations, $\binom{2}{3}$, for example, an appropriate error message is produced.

In estimates of frequency domain statistical parameters, if DOF=0, the frequency domain parameters are meaningless. DOF is one of the $PSDIN namelist parameters.

If there is an error in the fourier transform, the message ARRAY BOUNDS EXCEEDED IN FFT will be produced.

The appendix deals with computing the confidence levels for the statistical parameters estimated in SPA.

LIMITATIONS

Only 10 channels of information (parameters) may be analyzed at one time. There may be as many time intervals (with possibly different parameters) on file 2 as desired.

The SPA program will analyze arbitrarily many points in the time domain and, if requested, plot a time history. This may generate a plotted output of unacceptable

length. As mentioned above, the program will put a maximum of 1024 points in the frequency domain (for spectrum estimate parameters—PSD, CROSS SPECTRUM, COHERENCE, and so forth).

An estimate of the time required for a SPA run can be arrived at by using the following algorithm:

$$\text{TIME (SECONDS)} = \text{NCH} \times 100 + 20\text{LN (NPTS)}$$

where

NCH   number of channels

NPTS   number of points being analyzed

LN   natural log

If only a statistical analysis is being done, or if the data are not being filtered, the time estimate may be revised downward.

SPECTRUM ANALYSIS PROGRAM TECHNICAL DESCRIPTION

Smoothing Windows

The estimates of the frequency domain parameters may be smoothed by any one of four windows: Hann-Tukey, Hamming, Bartlett, or moving average. These functions are called windows because they window the data in the time domain, or, equivalently, convolve the data in the frequency domain.

Hann-Tukey smoothing window.—The first window we shall consider is the three-point Hann-Tukey window. In the frequency domain it amounts to the weighted moving average given by the following equation:

$$X(f) = [X(f - 1) + 2X(f) + X(f + 1)]/4$$

We can view this (and all the other windows) as a digital filter and analyze its effect on a frequency domain estimate by using the expression

$$Y(z) = [X(z - 1) + 2X(z) + X(z + 1)]/4$$

where $Y(z)$ is the output from the $X(z)$ input and z is the time lag variable.

Then, from the real translation theorem,

$$Y(z) = \left[X(z)z^{-1} + 2X(z) + X(z)z\right]/4$$

This implies that $Y(z) = X(z)\left(z^{-1} + 2 + z\right)/4$, so $\frac{Y(z)}{X(z)} = H(z) = \left(z^{-1} + 2 + z\right)/4$.

29

On the unit circle, $Z = e^{i\omega}$, so

$$H\left(e^{i\omega}\right) = \left(e^{-i\omega} + 2 + e^{i\omega}\right)/4$$

$$= \left[2 + 2\left(\frac{e^{i\omega} + e^{-i\omega}}{2}\right)\right]/4$$

$$= (2 + 2\cos\omega)/4$$

$$= (1 + \cos\omega)/2$$

where $-\pi < \omega < \pi$.

Therefore, the transfer function of the digital filter with which we are smoothing is

$$H\left(e^{i\omega}\right) = (1 + \cos\omega)/2$$

The result of this equation is real.

It is common to plot the transfer function modulus squared (also called squared gain) as follows:

$$\left|H\left(e^{i\omega}\right)\right|^2 = \left[(1 + \cos\omega)^2\right]/4$$

Table 1 lists the values of the squared gain of all the SPA smoothing window transfer functions at various frequencies. Figures 3 to 6 are plots of the same information.

TABLE 1 — SQUARED GAIN OF SMOOTHING WINDOW TRANSFER FUNCTIONS

| Frequency	Smoothing window							
	Hann–Tukey		Hamming		Bartlett		Moving average	
	Number of points in window							
	3	5	3	5	3	5	3	5
	Squared gain							
---	1 000	1 000	1 000	1 000	1 000	1 000	1 000	1 000
$\pi/16$	0 981	0 962	0 982	0 965	0 981	0 950	0 975	0 925
$\pi/8$	0 925	0 856	0 931	0 867	0 925	0 812	0 901	0 727
$3\pi/16$	0 839	0 704	0 851	0 724	0 839	0 621	0 788	0 470
$\pi/4$	0 729	0 531	0 749	0.561	0 729	0 419	0 648	0 233
$5\pi/16$	0 605	0 366	0 633	0.401	0.605	0 245	0 495	0 072
$3\pi/8$	0 478	0 228	0 513	0 263	0 478	0 120	0 346	0 005
$7\pi/16$	0 357	0.127	0 397	0 157	0.357	0 046	0.215	0 008
$\pi/2$	0 250	0 063	0 292	0 085	0 250	0 012	0.111	0 040
$9\pi/16$	0 162	0 026	0 203	0 041	0 162	0 002	0 041	0 061
$5\pi/8$	0 095	0 009	0 132	0 018	0 095	0	0 005	0 056
$11\pi/16$	0 049	0 002	0 081	0 007	0 049	0	0 001	0.031
$3\pi/4$	0 021	0	0 046	0 002	0 021	0	0 019	0.007
$13\pi/16$	0 007	0	0 025	0 001	0 007	0 002	0 049	0
$7\pi/8$	0 001	0	0 013	0	0 001	0 006	0 080	0 013
$15\pi/16$	0	0	0 008	0	0	0 011	0 103	0 031
π	0	0	0 006	0	0	0 012	0 111	0 040
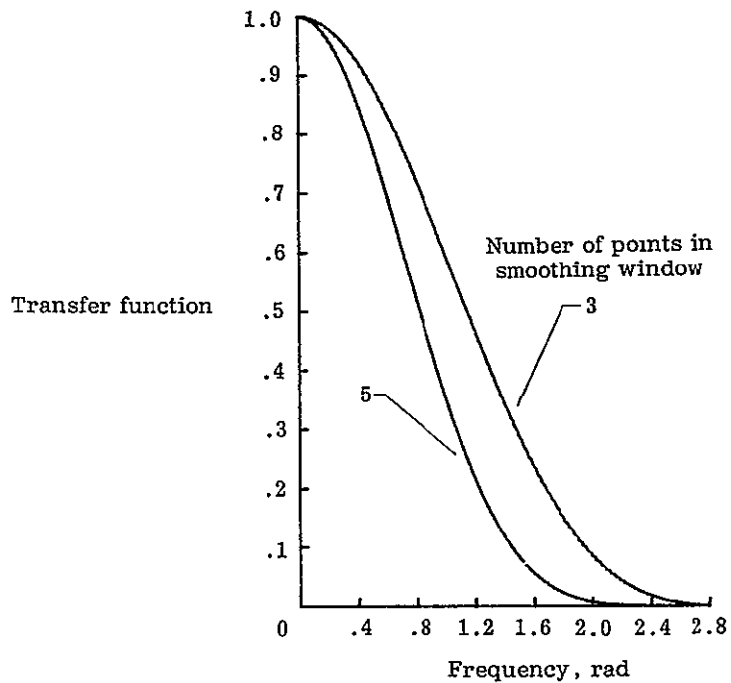
Figure 3. Squared gain of Hann-Tukey transfer function versus frequency.
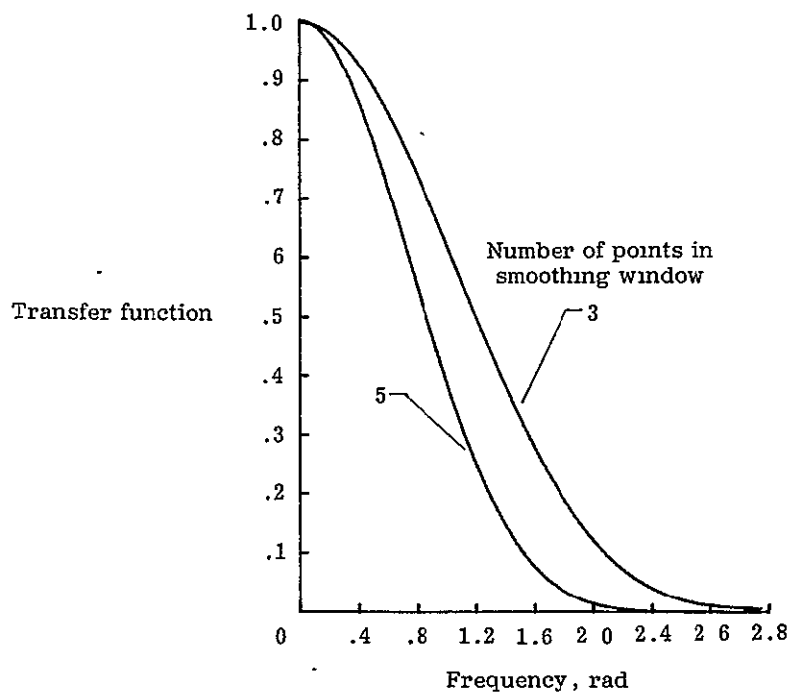

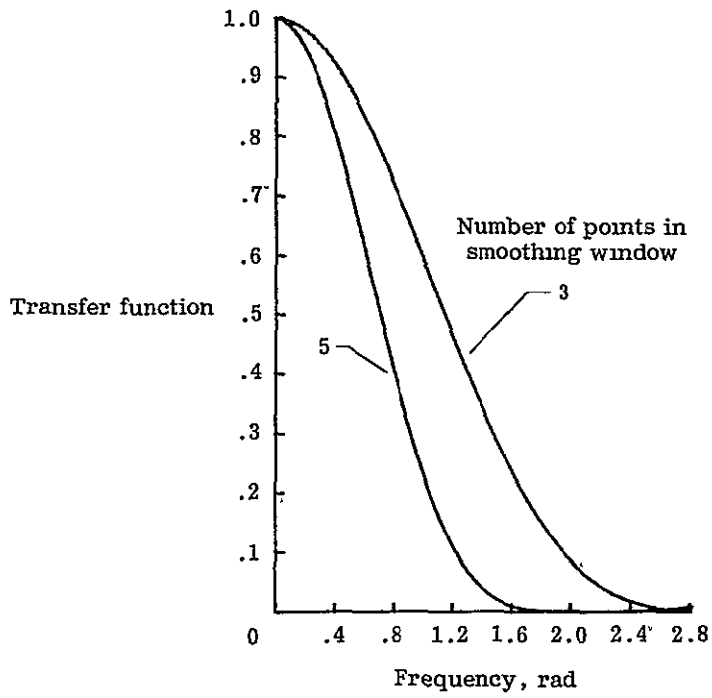Figure 4. Squared gain of Hamming transfer function versus frequency.

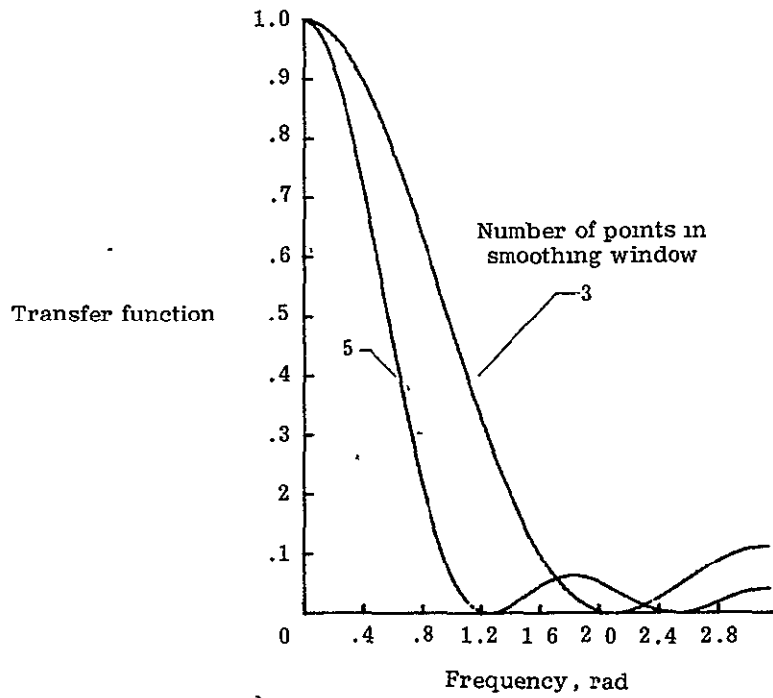Figure 5. Squared gain of Bartlett transfer function versus frequency.

Figure 6. Squared gain of moving average transfer function versus frequency.

The one-half power point, or break frequency (which corresponds to down 3 decibels) occurs at $\left| H\left(e^{i\omega}\right)\right|^2 = 1/2$. Solving $\left[(1 + \cos \omega)^2\right]/4 = 1/2$ for ω, the one-half power point for the three-point Hann-Tukey window is at 1.144 radians (65.53°).

The five-point Hann-Tukey window is given by the following equation:

$$X(f) = \left[X(f - 2) + 4X(f - 1) + 6X(f) + 4X(f + 1) + X(f + 2)\right]/16$$

and has a squared gain, $\left| H\left(e^{i\omega}\right)\right|^2$, equal to $\left[(3 + 4\cos \omega + \cos 2\omega)^2\right]/8$, with a break frequency at 0.8206 radian (47.02°).

Hamming smoothing window.—The three-point Hamming smoothing window is given by the expression

$$X(f) = 0.23X(f - 1) + 0.54X(f) + 0.23X(f + 1)$$

Deriving $\left| H\left(e^{i\omega}\right)\right|$ in a manner similar to that described above, we have

$$H\left(e^{i\omega}\right) = 0.54 + 0.46\cos \omega$$

where $-\pi < \omega < \pi$.

The squared gain of the Hamming window, $\left| H\left(e^{i\omega}\right)\right|^2$, is $(0.54 + 0.46\cos \omega)^2$ (table 1, fig. 4). The break frequency is at $\left| H\left(e^{i\omega}\right)\right|^2 = 1/2$.

Solving $(0.54 + 0.46\cos \omega)^2 = 1/2$ for ω, the break frequency for the three-point Hamming window is at $\omega = 1.1990$ radians or 68.70°. The Hann-Tukey window, unlike the Hamming smoothing window, does not completely eliminate the high frequency of the estimate.

The five-point Hamming window is given by the following equation:

$$X(f) = 0.0529X(f - 2) + 0.2484X(f - 1) + 0.3974X(f)$$
$$+ 0.2484X(f + 1) + 0.0529X(f + 2)$$

This expression is algebraically equivalent to smoothing with a three-point Hamming window and then smoothing the smoothed estimate with the same three-point Hamming window again.

The five-point Hamming window has a squared gain, $\left| H\left(e^{i\omega}\right)\right|^2$, equal to $(0.1058\cos 2\omega + 0.4968\cos \omega + 0.3974)^2$ with a break frequency at 0.8578 radian or 49.15°.

Bartlett smoothing window.—The three-point Bartlett smoothing window is the same as the three-point Hann–Tukey window. It is given by the expression

$$X(f) = [X(f - 1) + 2X(f) + X(f + 1)]/4$$

and has a squared gain, $\left|H\left(e^{i\omega}\right)\right|^2$, equal to $\left[(1 + \cos \omega)^2\right]/4$. The break frequency is at 1.144 radians (65.53°) (table 1, fig. 5).

The five-point Bartlett smoothing window is given by the equation

$$X(f) = [X(f - 2) + 2X(f - 1) + 3X(f) + 2X(f + 1) + X(f + 2)]/9$$

and has a squared gain, $\left|H\left(e^{i\omega}\right)\right|^2$, equal to $\left[(2\cos 2\omega + 4\cos \omega + 3)^2\right]/9$. The break frequency is at 0.7054 radian (40.42°).

Moving average window.—The three-point moving average window is given by

$$X(f) = [X(f - 1) + X(f) + X(f + 1)]/3$$

and has a squared gain, $\left|H\left(e^{i\omega}\right)\right|^2$, equal to $\left[(1 + \cos \omega)^2\right]/9$. The break frequency is at 0.9756 radian (55.89°).

The five-point moving average window is given by

$$X(f) = [X(f - 2) + X(f - 1) + X(f) + X(f + 1) + X(f + 2)]/5$$

and has a squared gain, $\left|H\left(e^{i\omega}\right)\right|^2$, equal to $\left[(1 + 2\cos \omega + 2\cos 2\omega)^2\right]/25$. The break frequency is at 0.63969 radian (36.6514°) (table 1 and fig. 6).

Linear Trend Removal

A special correction may be required to remove trend from the data. Trend is a frequency component of period greater than record length. This component is only partially removed by high-pass digital filtering. Frequency components above the break frequency will remain. (Because the trend frequency is lower than the lowest, or fundamental, frequency, it will be expressed as the sum of the fundamental and its harmonics.) The SPA program will remove linear trend by a least-squares fit; a linear regression line for the entire data record is estimated and then subtracted from the data.

34

Digital Filtering

If a stable system with transfer function $H(z)$ is driven by a sinusoidal input, $u(k)$, is equal to $\sin(k\omega T)$ where $k = 0,1,2,\ldots$ the resultant steady-state response is given by the expression

$$y(k) = \left| H\left(e^{i\omega T}\right) \right| \sin(k\omega T + \theta)$$

where θ equals ARG $H\ e^{i\omega T}$. That is, the steady-state sinusoidal response can easily be obtained by evaluating the system's transfer function at $Z = \exp(i\omega T)$ where ω is the frequency in radians of the sinusoidal input and T is the sampling period, $1/sps$.

A filter that severely attenuates sinusoids with frequencies outside the pass band is essential. The break frequency for the pass band is defined as the half-power point, that is, the frequency at which the amplitude of the system response is down 3 decibels.

The design of the fourth-order low-pass digital filter in the SPA program is predicated upon the assumption that the squared gain factor of the transfer function $H(z)$ is as follows:

$$\left| H\left(e^{i\omega T}\right) \right|^2 = \frac{1}{1 + \dfrac{\tan^{2n}(\omega T/2)}{\tan^{2n}\left(\omega_0 T/2\right)}}$$

where ω_0 is the break frequency and n is the order of the filter. The ideal low-pass filter would have a squared gain factor which resembled the following sketch (for $\omega_0 = \frac{\pi}{9} T$ radians):

Frequency, rad
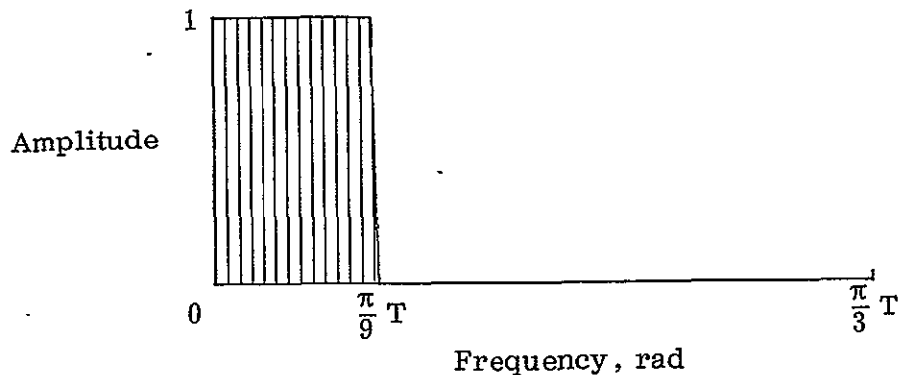
For the fourth-order case ($n = 4$; that is, there are four poles in the transfer function) the actual squared gain factor can be computed from equation (1) by substituting various values of ω. The actual squared gain factor resembles
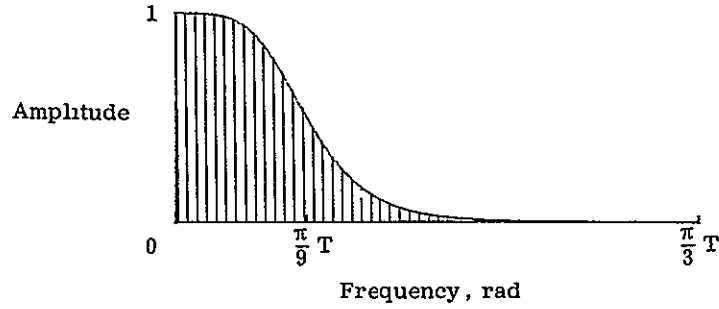
The transfer function for the low-pass filter specified above is arrived at by the following substitutions:

$$\tan\left(\frac{\omega T}{2}\right) = \frac{\sin\left(\frac{\omega T}{2}\right)}{\cos\left(\frac{\omega T}{2}\right)} = \frac{e^{-i\omega T/2}}{i e^{-i\omega T/2}}\left(\frac{e^{i\omega T} - 1}{e^{i\omega T} + 1}\right) = \frac{1}{i}\left(\frac{e^{i\omega T} - 1}{e^{i\omega T} + 1}\right)$$

Hence

$$\left|H\left(e^{i\omega T}\right)\right|^2 = \frac{\tan^{2n}\left(\omega_0 T/2\right)}{\tan^{2n}\left(\omega_0 T/2\right) + (-1)^n\left(\dfrac{e^{i\omega T} - 1}{e^{i\omega T} + 1}\right)}$$

Letting Z equal $e^{i\omega T}$, with ω_0 fixed,

$$\left|H(z)\right|^2 = \frac{\tan^{2n}\left(\omega_0 T/2\right)}{\tan^{2n}\left(\omega_0 T/2\right) + (-1)^n(z - 1)/(z + 1)}$$

Multiplying numerator and denominator by $(z + 1)^{2n}$,

$$\left|H(z)\right|^2 = \frac{\tan^{2n}\left(\omega_0 T/2\right)(z + 1)^{2n}}{\tan^{2n}\left(\omega_0 T/2\right)(z + 1)^{2n} + (-1)^n(z - 1)^{2n}}$$

Using then the binomial expansion for $(z + 1)^{2n}$, the denominator of the above expression may be written as follows:

$$\left[\tan^{2n}\left(\omega_0 T/2\right) + (-1)^n\right]\left[z^{2n} + \beta\binom{2n}{1}z^{2n-1} + \binom{2n}{2}z^{2n-2}\right.$$
$$\left. + \beta\binom{2n}{3}z^{2n-3} + \binom{2n}{4}z^{2n-4} + \ldots + \beta\binom{2n}{2n-1}z + 1\right]$$

36

where

$$\beta = \frac{\tan^{2n}\left(\omega_0 T/2\right) - (-1)^n}{\tan^{2n}\left(\omega_0 T/2\right) + (-1)^n}$$

The $2n$ roots of the expanded form of the denominator are given by the following expression:

$$r_k = \frac{1 - \tan^2\left(\omega_0 T/2\right) + i2\tan\left(\omega_0 T/2\right)\sin\theta k}{1 - 2\tan\left(\omega_0 T/2\right)\cos\theta_k + \tan^2\left(\omega_0 T/2\right)}$$

where $k = 1, 2, \ldots, 2n$ and θ_k equals $\left(\dfrac{k-1}{n}\right)\pi$ when n is odd and $\left(\dfrac{2k-1}{n}\right)\pi$ when n is even.

Thus, the squared gain factor may be written in the following form:

$$|H(z)|^2 = \frac{\tan^{2n}\left(\omega_0 T/2\right)}{\tan^{2n}\left(\omega_0 T/2\right) + (-1)^n}\left[\frac{(z+1)^{2n}}{\left(z - r_1\right)\left(z - r_2\right)\cdots\left(z - r_{2n}\right)}\right]$$

If $r = ae^{i\omega}$ is a root of this polynomial, so are $ae^{-i\omega}$, $\dfrac{1}{a}e^{i\omega}$, and $\dfrac{1}{a}e^{-i\omega}$.

Hence, the function $|H(z)|^2$ will have n poles inside the unit circle and n poles outside the unit circle (for $r \neq 1$). Letting $p_1, p_2, \ldots, p_n$ denote the poles inside the unit circle, the transfer function that has the desired squared gain factor is found as follows:

$$H(z) = \frac{b(z+1)^n}{\left(z - p_1\right)\left(z - p_2\right)\cdots\left(z - p_n\right)} \tag{2}$$

where b is selected so that the steady-state unit step response has a magnitude of one (that is, $H(1) = 1$). In other words,

$$b = \frac{\left(1 - p_1\right)\left(1 - p_2\right)\cdots\left(1 - p_n\right)}{2^n}$$

Having specified the transfer function $H(z)$, we derive the associated difference equation by the real translation theorem, which says that $Y(z)z$ equals $Y(z+1)$, or that $Y(z)z^{-1}$ equals $Y(z-1)$. Thus, $H(z)$ equals $\dfrac{Y(z)}{X(z)}$, or $X(z)H(z)$ equals $Y(z)$, and using the translation theorem, the entire function can be expressed in

terms of $X(z - m)$, $Y(z - m)$, and $M = 0,1,2,\ldots,n$. For example, suppose that $H(z)$ were given by $\dfrac{b(z + 1)}{(z - p_1)}$. Then b equals $\dfrac{z - 1}{2}$, $H(z)$ equals $\dfrac{Y(z)}{X(z)}$ equals $\dfrac{b(z + 1)}{(z - p_1)}$, and finally $Y(z)\big(z - p_1\big)$ equals $bX(z)(z + 1)$. Therefore

$$Y(z)z - p_1 Y(z) = b[X(z)z + X(z)]$$

and

$$Y(z + 1) - p_1 Y(z) = b[X(z + 1) + X(z)]$$

or, translating 1 unit,

$$Y(z) = p_1 Y(z - 1) + b[X(z) + X(z - 1)]$$

This is the desired difference equation, in which $X(z)$ is the current input, $X(z - 1)$ is the last input (input lagged 1 unit), $Y(z)$ is the present output, and $Y(z - 1)$ is the last output (output lagged 1 unit).

The design procedure for the high-pass digital filter is, of course, similar. The squared gain of the transfer function is found as follows:

$$\left|H\big(e^{i\omega T}\big)\right|^2 = \dfrac{1}{1 + \dfrac{\cot^{2n}(\omega T/2)}{\cot^{2n}\big(\omega_1 T/2\big)}}$$

where, once again, ω_1 is the break frequency and n is the order. What this amounts to is simply a rotation of the low-pass filter poles and zeros through an angle of π radians, where the half-power point for the low-pass design, ω_0, is taken to be $\dfrac{\pi}{2} - \omega_1$. The SPA program utilizes this information to use the same routine for high-pass and low-pass filter determination: As shown in the sketch below, the bandpass filter in SPA is simply a cascaded low-pass—high-pass system.
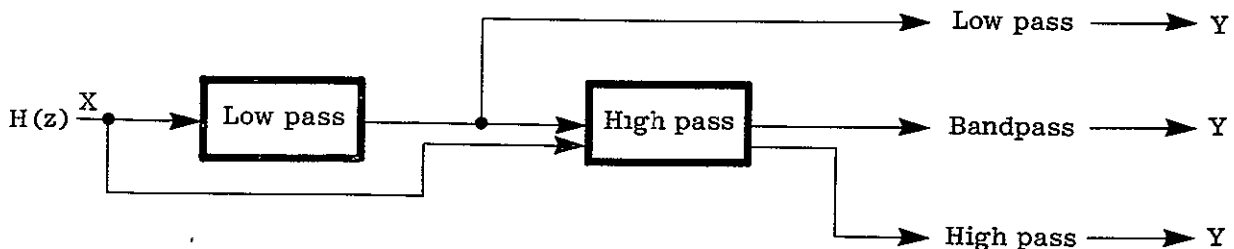
The fourth-order design of the filter cascade scheme was selected for the SPA program because it gives good attenuation outside the pass band without having so many recursive terms that word length contributes to instability or that rise time becomes a problem.

The SPA program also has first- and third-order Butterworth design low-pass and high-pass filters. The first-order designs suffer because the frequency response characteristics are not as square as in the fourth-order design. That is, the distinction between the frequencies that are passed and those that are attenuated is not as good as in the fourth-order case. However, first-order designs have a rapid rise time and only three recursive terms instead of four. The first-order filter characteristics resemble the following sketches.

First-order low-pass filter

First-order high-pass filter
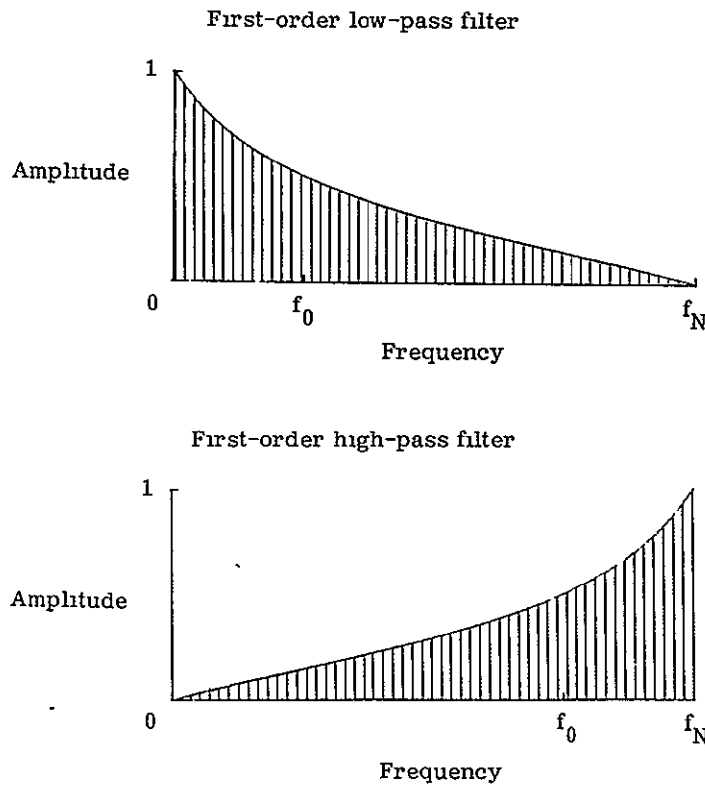
The first-order filters are derived in the following manner. The S-plane form for the first-order filters has only one pole, α. The transfer function, $G(s)$, of the low-pass filter is equal to $\frac{\alpha}{s + \alpha}$, and that of the high-pass filter is equal to $\frac{s}{s + \alpha}$.

For the low-pass case (by the bilinear transform mentioned earlier),

$$G(z) = \frac{k(z + 1)}{z - e^{-\alpha T}}$$

where k is selected so that $G(z)\big|_{z=1}$ = Gain = 1. Hence $k = \left(1 - e^{-\alpha T}\right)/2$ and the difference equation for the first-order low-pass case is found as follows:

$$G(z) = \frac{Y(z)}{X(z)} = \frac{k(z + 1)}{z - e^{-\alpha T}}$$

$$\frac{Y(z)}{X(z)} = \frac{k\left(1 + z^{-1}\right)}{1 - e^{-\alpha T}}$$

$$Y(z)\left[1 - e^{-\alpha T}z^{-1}\right] = X(z)k\left[1 + z^{-1}\right]$$

Therefore,

$$Y(z) = e^{-\alpha T}Y(z - 1) + k\left[X(z) + X(z - 1)\right]$$

where, once again, the sampling period, T, equals $1/\text{sps}$.

The design considerations and implementation for the third-order filters are similar to those for the first order. However, the transfer functions are three-pole, as follows. For the high-pass filter,

$$G(s) = \frac{s}{s + \alpha}\ \frac{s^2}{(s + a)^2 + b^2}$$

For the low-pass filter,

$$G(s) = \frac{\alpha}{s + \alpha}\ \frac{a^2 + b^2}{(s + a)^2 + b^2}$$

where f_0 is the break frequency and the transfer functions are derived by using the following expressions: $\alpha = 2\pi f_0$, $a = 2\pi f\cos \pi/3$, and $b = 2\pi f\sin \pi/3$.

The third-order filters expressed in the Z-plane as transfer functions are as follows. For the high-pass filter,

$$G(z) = \frac{k(z - 1)\left(z^2 - 2z + 1\right)}{\left(z - e^{-\alpha T}\right)\left(z^2 - 2e^{-aT}\cos bTz + e^{-2aT}\right)}$$

For the low-pass filter,

$$G(z) = \frac{k(z + 1)\left(z^2 - 2z + 1\right)}{\left(z - e^{-\alpha T}\right)\left(z^2 - 2e^{-aT}\cos bTz + e^{-2aT}\right)}$$

40

The transformation from the S- to the Z-plane is accomplished by the bilinear transform (ref. 1). The difference equations for the filters are then derived in the usual manner.

Time Domain Statistical Analysis

Summary of time domain statistical parameters.—The SPA program will compute the mean, $\overline{X}$; variance, s^2; standard deviation, s; mean square, m; and root mean square, rms. It also lists data maximum, data minimum, and number of observations. All of the above may be produced for either filtered or unfiltered data.

For example, let X_1, X_2, ..., X_n be n mutually stochastically independent random variables from a given distribution. Then

$$\overline{X} = \frac{1}{n} \sum_{i=1}^{n} X_i$$

$$s^2 = \frac{1}{n-1} \sum_{i=1}^{n} \left(X_i - \overline{X} \right)^2$$

$$s = \sqrt{s^2}$$

$$m = \frac{1}{n} \sum_{i=1}^{n} X_i^{\ 2}$$

$$rms = \sqrt{m}$$

In addition, the SPA program will produce a histogram of the time domain data (with from one to 40 intervals), and, based on $\overline{X}$ and s^2, fit a normal curve to the histogram by using the following expression:

$$y = \frac{\text{Area of histogram}}{s\sqrt{2\pi}} \exp\left[-\frac{1}{2}\left(\frac{X - \overline{X}}{s} \right)^2 \right]$$

The following χ^2 goodness-of-fit test is performed to examine the hypothesis that the data are normally distributed:

$$\chi^2 = \sum_{i=1}^{n} \frac{\left(O_i - E_i \right)^2}{E_i}$$

where O_i is the observed frequency in the ith class and E_i is the expected frequency in the ith class.

41

If the histogram is not different from a normal curve at the 0.05 level, an asterisk is printed at the χ^2-value.

The skewness and kurtosis of the distribution are also computed. Skewness is the third moment about the mean divided by standard deviation cubed, or

$$m_3/s^{3/2} = \frac{1}{n} \sum_{i=1}^{n} \left(X_i - \overline{X}\right)^3 / s^{3/2}$$

Kurtosis is the fourth moment about the mean divided by the standard deviation raised to the fourth power, or

$$m_4/s^4 = \frac{1}{n} \sum_{i=1}^{n} \left(X_i - \overline{X}\right)^2 / s^4$$

Details of time domain statistical analysis.—Given a sampled data sequence X_i, $i = 1, 2, \ldots, n$, the mean value (first moment) of the data is computed as follows:

$$m_1 = \frac{1}{n} \sum_{i=1}^{n} X_i$$

The second moment, m_2, is defined as follows

$$m_2 = \frac{1}{n} \sum_{i=1}^{n} \left(X_i - \overline{X}\right)^2$$

and is computed by the formula

$$m_2 = \left[\sum_{i=1}^{n} X_i^2 - \frac{1}{n} \left(\sum_{i=1}^{n} X_i \right)^2 \right] / n$$

The variance of the data is defined as follows:

$$s^2 = \text{var} = \frac{\displaystyle\sum_{i=1}^{n} X_i^2 - \frac{1}{n} \left(\sum_{i=1}^{n} X_i \right)^2}{n-1} = \frac{m_2 n}{n-1}$$

42

where we divide $\sum \left(X_i - \overline{X}\right)^2$ by $n - 1$ instead of by n because $\overline{X}$ is an estimate of the population mean, μ.

In other words, the sampled sequence $\{X_i\}_{i=1}^n$ is taken to be a finite sample from a larger (and possibly infinite) population. As a result, $\overline{X}$ will be an estimate of μ; that is, $\overline{X} - \mu = 0$, necessarily. Therefore

$$\sum_{i=1}^n \left(X_i - \overline{X}\right)^2 = \sum_{i=1}^n \left(X_i - \mu + \mu - \overline{X}\right)^2$$

$$= \sum_{i=1}^n \left[\left(X_i - \mu\right) - \left(\overline{X} - \mu\right)\right]^2$$

$$= \sum_{i=1}^n \left[\left(X_i - \mu\right)^2 - 2\left(\overline{X} - \mu\right)\left(X_i - \mu\right) + \left(\overline{X} - \mu\right)^2\right]$$

$$= \sum_{i=1}^n \left(X_i - \mu\right)^2 - 2\left(\overline{X} - \mu\right) \sum_{i=1}^n \left(X_i - \mu\right) + \sum_{i=1}^n \left(\overline{X} - \mu\right)^2$$

$$= \sum_{i=1}^n \left(X_i - \mu\right)^2 - 2\left(\overline{X} - \mu\right)\left(n\overline{X} - n\mu\right) + n\left(\overline{X} - \mu\right)^2$$

$$= \sum_{i=1}^n \left(X_i - \mu\right)^2 - 2n\left(\overline{X} - \mu\right)^2 + n\left(\overline{X} - \mu\right)^2$$

$$= \sum_{i=1}^n \left(X_i - \mu\right)^2 - n\left(\overline{X} - \mu\right)^2$$

which is to say that $\sum_{i=1}^n \left(X_i - \overline{X}\right)^2$ is always less than or equal to $\sum_{i=1}^n \left(X_i - \mu\right)^2$.
To correct for this bias, we divide by $n - 1$ instead of by n.

The third moment about the mean is defined as follows.

$$m_3 = \sum_{i=1}^{n} \frac{\left(X_i - \overline{X}\right)^3}{n}$$

and is computed by the following equation:

$$m_3 = \left[\sum_{i=1}^{n} X_i^3 - \frac{3}{n} \sum_{i=1}^{n} X_i^2 \sum_{i=1}^{n} X_i + \frac{2}{n^2} \left(\sum_{i=1}^{n} X_i \right)^3 \right] / n$$

Skewness is defined as $m_3/s^{3/2}$.

The fourth moment about the mean, m_4, is defined as follows

$$m_4 = \frac{\sum_{i=1}^{n} \left(X_i - \overline{X}\right)^4}{n}$$

and is computed by using the following equation:

$$m_4 = \left[\sum_{i=1}^{n} X_i^4 - \frac{4}{n} \left(\sum_{i=1}^{n} X_i^3 \right) \left(\sum_{i=1}^{n} X_i \right) + \frac{6}{n^2} \left(\sum_{i=1}^{n} X_i^2 \right) \left(\sum_{i=1}^{n} X_i \right)^2 - \frac{3}{n^3} \left(\sum_{i=1}^{n} X_i \right)^4 \right] / n$$

Kurtosis is defined as m_4/s^4.

The mean square value of the data is equal to $\frac{1}{n} \sum_{i=1}^{n} X_i^2$ and the root mean square

is equal to $\left(\frac{1}{n} \sum_{i=1}^{n} X_i^2 \right)^{1/2}$

The autocorrelation estimate for the first 10 lag values is given by the following equation:

$$R_X(\tau) = \left[\frac{1}{n - |\tau| - 1} \sum_{i=1}^{n-|\tau|-1} X(i)X(i + \tau) \right] / R_X(0)$$

where $\tau \neq 0$ and $\{X_i\}_{i=1}^{n}$ is assumed to have a zero mean. Computing more than 10 lag values by this technique would be time consuming. The autocorrelation function gives the same information the autospectrum does. Since the autospectrum is computed with greater resolution, more values of $R_X(\tau)$ are not computed.

The autocorrelation function obeys $R_X(\tau) \leq R_X(0)$ for all values of τ. All autocorrelation functions are normalized by dividing by $R_X(0)$. The formula used for data with a nonzero mean is as follows:

$$R_X(\tau) = \frac{1}{n - |\tau| - 1} \left[\sum_{i=1}^{n-|\tau|-1} \left(X_i X_{i+\tau} \right) - (n - |\tau|)\bar{X} \right] / R_X(0)$$

so the mean does not have to be first subtracted from the data. In addition, frequency histograms are computed and plotted for the data. The ordinate scale can be partitioned into from one to 40 intervals. Based on the area, A; mean, $\bar{X}$; and variance, s^2, of the data, a normal curve is fitted to the histogram. The equation for the curve is as follows:

$$y = \frac{A}{\sqrt{2\pi s^2}} \, e^{-\frac{1}{2}\left(\frac{X - \bar{X}}{s}\right)^2} \tag{3}$$

where A is the area of the fitted histogram.

The goodness of fit of the normal curve defined by equation (3) to the histogram is determined by the χ^2-statistic defined as follows:

$$\chi^2 = \sum_{i=1}^{n} \frac{\left(O_i - E_i\right)^2}{E_i}$$

where

O_i number of observations in the ith interval

E_i number of observations expected in the ith interval

n number of intervals (number of degrees of freedom plus 1)

The histogram plot prints out many of the time domain statistics (mean, variance, mean square, root mean square, skewness, kurtosis, and so forth) and prints the χ^2 goodness-of-fit statistic. In addition, if the χ^2 test reveals that the histogram

does not differ from the normal curve (with 95 percent probability), an asterisk is printed above the x^2-statistic.

Table 2, which is used in the SPA program for the evaluation of the x^2-statistic, was taken from reference 2 for the first 30 degrees of freedom. For degrees of freedom from 31 to 39, the x^2 density function was integrated numerically as follows:

$$P(X \leq x) = \int_0^x \frac{1}{\Gamma(v/2)2^{v/2}} \omega^{v/2-1}e^{-\omega/2}d\omega$$

Therefore, for 31 degrees of freedom at the 0.025 level, one would solve the following equation for x:

$$P(X \leq x) = \int_0^x \frac{1}{\Gamma(31/2)2^{31/2}} \omega^{31/2-1}e^{-\omega/2}d\omega = 0.025$$

Frequency Domain Statistical Analysis

Summary of frequency domain statistical analysis.—The Spa program computes power spectrum, cross spectrum, coherence, phase angle, amplitude ratio, and transfer function. In addition, it produces the integral rms and maximum and minimum values of the frequency domain parameters.

Power spectrum: The theoretical power spectrum

$$G_X(f) = 2\lim_{T \to \infty} \frac{1}{T} \left| \int_{-\frac{T}{2}}^{\frac{T}{2}} x(t)e^{-i2\pi ft}dt \right|^2$$

is defined by the fourier transform operation. The digital fourier transform is given by

$$G_X(f_1) = 2\Delta t \left| \sum_{k=0}^{n-1} x(t)e^{-i2\pi f_1 k/n} \right|^2$$

TABLE 2.—x^2 TABLE

n	Probability	
	0 975	0 025
	x^2	
1	0 00098	5 02
2	0 0506	7 38
3	0 216	9.35
4	0.484	11.14
5	0 831	12 83
6	1.24	14 45
7	1.69	16 01
8	2 18	17 53
9	2 70	19 02
10	3 25	20 48
11	3.82	21 92
12	4 40	23 34
13	5 01	24 74
14	5 63	26 12
15	6 26	27 49
16	6 91	28.85
17	7 56	30 19
18	8.23	31.53
19	8.91	32 85
20	9.59	34 17
21	10 28	35 48
22	10 98	36 78
23	11 69	38.08
24	12 40	39 36
25	13 12	40.65
26	13 82	41 92
27	14.57	43 19
28	15.31	44 46
29	16 05	45 72
30	16 79	46 98
31	17 54	48 25
32	18 29	49.48
33	19 05	50.72
34	19 81	51.97
35	20.57	53.20
36	21 34	54 44
37	22 11	55 67
38	22.88	56 89
39	23.66	58.12
40	24 43	59 34

Time domain data of arbitrary length can be used to produce power-spectrum estimates by fourier transforming the data in sections. This amounts to static averaging the full frequency domain estimate.

Cross spectrum: The cross spectrum is defined as follows:

$$G_{XY}(f) = X^*(f)Y(f)$$

where

    $X(f)$ fourier transform of $x(t)$, the input signal

    $Y(f)$ fourier transform of $y(t)$, the output signal

    *   complex conjugate

Thus, the cross spectrum will, in general, be a complex-valued function. As a result, it is represented as a modulus and phase angle, as follows:

$$G_{XY}(f) = \left| G_{XY}(f) \right| e^{-i\theta_{XY}(f)}$$

where the phase angle, $\theta_{XY}(f)$, is

$$\theta_{XY}(f) = \tan^{-1}\left[\frac{Q_{XY}(f)}{C_{XY}(f)} \right]$$

Thus, $Q_{XY}(f)$, the so-called quad spectrum, is $Im[Y(f)]Re[X(f)] - Re[Y(f)]Im[X(f)]$ and $C_{XY}(f)$, the so-called co spectrum, is $Re[X(f)]Re[Y(f)] + Im[X(f)]Im[Y(f)]$.

Both power- and cross-spectrum estimates can be smoothed. The smoothing windows available are the Hann-Tukey, Hamming, Bartlett, and moving average windows. From three to 35 points may be specified in each window.

Coherence function: The coherence function is computed as follows:

$$CF_{XY}(f) = \frac{G_{XY}^{2}(f)}{G_X(f)G_Y(f)}$$

It is meaningful, of course, only for smoothed estimates of $G_{XY}(f)$, $G_X(f)$, and $G_Y(f)$. This is automatically taken care of in the SPA program.

Amplitude ratio: Amplitude ratio is computed as follows:

$$AR_{XY}(f) = \frac{G_{XY}(f)}{G_X(f)}$$

Amplitude ratio is also computed with smoothed values of $G_{XY}(f)$ and $G_X(f)$.

Transfer function: Transfer function is computed as follows:

$$TF_{XY}(f) = G_{XY}{}^2(f)/G_X{}^2(f)$$

where $x(t)$ is input and $y(t)$ is output. The value of $TF_{XY}(f)$ is also computed with smoothed estimates of $G_{XY}(f)$ and $G_X(f)$.

Details of frequency domain statistical analysis. —A time series or time history is a random (nondeterministic) function X of an independent variable, usually time. The most important assumption made about a time series is that the stochastic process from which the series arises is stationary; that is, that the first moment and the autocovariance function are independent of time.

The autocovariance function is given by the following equation

$$C_X(\tau) = \frac{1}{n - |\tau| - 1} \sum_{i=1}^{n-|\tau|} \left(X_i - \overline{X}\right)\left(X_{i+\tau} - \overline{X}\right)$$

and the autocorrelation function is computed as follows:

$$R_X(\tau) = C_X(t)/C_X(0)$$

As we shall see, this equivalent to requiring the mean and power spectrum to be independent of time.

At the heart of frequency domain analysis is the fact that it is possible to represent arbitrary periodic functions as a linear combination of any class of orthogonal periodic functions. Furthermore, with suitable interpretation it is possible to represent nonperiodic functions as a sum of periodic functions. In fourier analysis, the periodic functions used are sine and cosine functions.

Thus, given an arbitrary function on the interval $[0,n]$, which is a sampled function $\left\{X_i\right\}_{i=1}^n$, we may express $\left\{X_i\right\}_{i=1}^n$ as follows:

$$A_0 + 2 \sum_{m=1}^{n-1} \left\{A_m \cos(2\pi mk/n) + B_m \sin(2\pi mk/n)\right\}$$

48

where $k = -n, -(n-1), \ldots, 0, \ldots, n-1, n$. This is to say that we consider n sine and cosine components and express $\{X_i\}_{i+1}^{n}$ as a sum of these components.

Notice that each of these sine/cosine components has period n. The frequency $1/n$ is in some sense a fundamental frequency and corresponds to a period of a length equal to the length of the record. The other frequencies are then harmonics or multiples of the fundamental frequency.

Notice that this transformation (known in common parlance as a fourier transform) reveals a considerable amount of information; namely, the relative proportion of the variance of the original time series that can be attributed to each of the harmonics, k/n where $k = 2, 3, 4, \ldots, n-1$.

Displaying the sum of the squares of A_m and B_m versus frequency m/n is nothing more than the power spectral density at that frequency (or, as it is sometimes called, the autospectrum).

Computing the so-called fourier coefficients, A_m and B_m, is facilitated by the fact that the sines and cosines are orthogonal; that is,

$$\sum_{r=-n}^{n} \sin\left(\frac{2\pi kr}{n}\right) \cos\left(\frac{2\pi mr}{n}\right) = \begin{cases} 0 & \begin{array}{l} k, m, \text{integers}, \\ \underbrace{-n, -n+1, \ldots, 0, \ldots, n-1, n}_{2n \text{ terms}} \end{array} \end{cases}$$

$$\sum_{r=-n}^{n} \sin\left(\frac{2\pi kr}{n}\right) \sin\left(\frac{2\pi mr}{n}\right) = \begin{cases} 0 & k \neq m \\ n/2 & k=m, m \neq 0, n \\ 0 & k=m=0, n \end{cases}$$

$$\sum_{r=-n}^{n} \cos\left(\frac{2\pi kr}{n}\right) \cos\left(\frac{2\pi mr}{n}\right) = \begin{cases} 0 & k \neq m \\ n/2 & k=m, m \neq 0, n \\ n & k=m=0, n \end{cases}$$

Hence, the fourier coefficients can be estimated as follows:

$$X(m) = A_0 + 2 \sum_{k=1}^{n-1} \left\{ A_m \cos\left(\frac{2\pi mk}{n}\right) + B_m \sin\left(\frac{2\pi mk}{n}\right) \right\}$$

$$X(m)\cos\left(\frac{2\pi kr}{n}\right) = A_0 \cos\left(\frac{2\pi kr}{n}\right) + \sum_{k=1}^{n-1} \left\{ A_m \cos\left(\frac{2\pi mk}{n}\right) \cos\left(\frac{2\pi rk}{n}\right) \right.$$

$$\left. + B_m \sin\left(\frac{2\pi mk}{n}\right) \cos\left(\frac{2\pi rk}{n}\right) \right\}$$

By applying the above orthogonality relations, we get

49

$$2 \sum_{k=0}^{n-1} X_k \cos\left(\frac{2\pi mk}{n}\right) = nA_m$$

and

$$2 \sum_{k=0}^{n-1} X_k \sin\left(\frac{2\pi mk}{n}\right) = nB_m$$

This yields a convenient method for finding the fourier coefficients:

$$A_m = \frac{2}{n} \sum_{k=0}^{n-1} X_k \cos\left(\frac{2\pi mk}{n}\right)$$

$$B_m = \frac{2}{n} \sum_{k=0}^{n-1} X_k \sin\left(\frac{2\pi mk}{n}\right)$$

Proceeding directly (as outlined on the previous page) has a significant drawback in that computing the coefficients A_m and B_m requires on the order of $2n^2$ operations; for a large value of n this would be prohibitively time consuming. To alleviate this problem, Cooley and Tukey presented an algorithm for computing the fourier coefficients in a much more efficient manner—by decomposing n (ref. 3). This result is known as a fast fourier transform (FFT), and it makes the computation of the fourier coefficients practical for even a large number of data points. The SPA program uses the fast fourier transform routine written by R.C. Singleton (ref. 4).

The fourier transform of a function X (if it exists) is as follows:

$$X(f) = \int_{-\infty}^{\infty} x(t)e^{-i2\pi ft}dt$$

Recalling the earlier formula

$$e^{i\omega} = \cos \omega + i \sin \omega$$

this equation reveals that the fourier transform is nothing more than an extension of the method of computing fourier coefficients to the complex plane. Most theoretical discussions use the fourier transform, since it is easily specified and the complex plane is (algebraically) closed. The fourier transform, $X(f)$, of a function $x(t)$ exists if the square of $x(t)$ is Lebesque integrable (ref. 5).

The fourier transform of $x(t)$ is given by

$$X(f) = F[x(t)] = \int_{-\infty}^{\infty} x(t)e^{-i2\pi ft}dt$$

50

and the inverse fourier transform is

$$x(t) = F^{-1}[X(f)] = \int_{-\infty}^{\infty} X(f)e^{i2\pi ft}df$$

Problems with the existence of the fourier transform have been handled largely with delta functions (ref. 6).

In any case, finite fourier transforms always exist. Delta functions at a point are presented by the peak of finite height; this is because the finite fourier transform

$$X(f) = \frac{1}{n}\sum_{k=0}^{n-1} x(k)e^{-i2\pi kf/n}$$

has finite resolution, $1/n$, rather than infinitesimal resolution, as in the infinite case.

The parameters of interest in the frequency domain fall into two broad categories: those that concern a single signal and those that concern input/output relationships. For parameters in the first category, the power spectral density function or auto-spectrum is the most important. For parameters in the second category the parameters of primary importance are transfer function, coherence, and phase angle.

To estimate the power spectrum, first the fourier transform of the time series is taken. Then the modulus of the resulting complex-valued function is determined and normalized by dividing it by the frequency bandwidth of the estimate. Records of arbitrary length (that is, signals of arbitrary length) in the time domain are fourier transformed in parts and then averaged. Mathematically, this amounts to fourier transforming the entire record and then averaging over frequencies to enhance the statistical quality of the estimate.

The power-spectrum estimate which is defined by

$$G_X(y) = 2\lim_{T\to\infty} \frac{1}{T} \left| \int_{-\frac{T}{2}}^{\frac{T}{2}} x(t)e^{-i2\pi ft}dt \right|^2$$

is actually estimated by a discrete fourier transform:

$$\hat{G}_X(f_i) = 2\Delta t \left| \sum_{i=0}^{n-1} x(t_i)e^{-i2\pi ft_i/n} \right|^2$$

The estimate produced by the SPA program is one sided; that is, powers at negative frequencies are expressed as their positive counterparts. All n values of $X(f)$ in the time domain are used to compute each PSD value at the frequencies $\frac{SPS}{n}, \frac{2SPS}{n}, \ldots, \frac{(n/2)SPS}{n}$.

The cross-spectrum estimate, unlike the power-spectrum estimate, is not a real quantity. Hence it is expressed as a gain and phase angle at each frequency, f_i. The gain is defined as follows:

$$G_{XY}(f_i) = X^*(f_i)Y(f_i)$$

where $X(f)$ is the fourier transform of the input signal, $Y(f)$ is the fourier transform of the output, and * denotes a complex conjugate.

The phase angle at f is given by the expression

$$\theta_{XY}(f_i) = \tan^{-1}\left[\frac{Q_{XY}(f_i)}{C_{XY}(f_i)}\right]$$

where $Q_{XY}(f_i)$ is the so-called quad spectrum, $\text{Im}Y(f_i)\text{Re}X(f_i) - \text{Re}Y(f_i)\text{Im}X(f_i)$, and $C_{XY}(f)$ is the coincident spectrum and is defined by the following equation:

$$C_{XY}(f_i) = \text{Re}X(f_i)\text{Re}Y(f_i) + \text{Im}X(f_i)\text{Im}Y(f_i)$$

Once again, for long time domain signals, the co and quad (C and Q) spectrum estimates are computed over subintervals and then averaged, just as with the power spectral density estimate. Coherence, $\gamma_{XY}^2(f_i)$, is given by the following equation:

$$\gamma_{XY}^2(f_i) = \frac{\left|G_{XY}(f_i)\right|^2}{G_X(f_i)G_Y(f_i)}$$

It is smoothed to increase the number of degrees of freedom, since a two-degree-of-freedom coherence estimate is identically 1.

Internal Data Arrays and Definition of Variables

The definitions in the following section refer to the actual FORTRAN program. Copies of this program are available from the author.

Common variables.—The variables defined below are common to several subroutines in the SPA program. The line numbers mentioned refer to the line coding in the subroutines.

COMMON/INFO/TITLE(8),CID(10),NCH,SPS,JULIAN:

/INFO/ — used in nearly every SPA routine.

TITLE — array of title information (alphanumeric) used in SPA output.

CID — array of SPA channel identification names in file 4 (unfiltered data) or file 5 (filtered data). The first parameter has the name CID(10), and so on.

NCH — number of SPA channels in file 4 (or file 5).

52

SPS — samples per second, sampling rate of data in file 4 (or file 5).

JULIAN — date of SPA run.

COMMON/TAPEOP/OPEN: Logical variable used to determine whether plotter has been opened.

COMMON/TYMAXS/ST(4),ET(4): Start and stop times (integers) of data in file 4. Used for plotting time axis in filtered and unfiltered data plots.

COMMON/FLTER/HIPASS,LOPASS,BANDPAS,FC,BW,PRINT,BREAK: HIPASS, LOPASS,BANDPAS, and PRINT are all logical variables determining filter type and whether or not the filtered data are to be printed. FC is the center frequency, and BW is the bandwidth for the BANDPAS filter (FC − BW to FC + BW). BREAK is the break frequency for HIPASS or LOPASS filters.

COMMON/PLTINF/RATE,BRK,ORD: These variables are used for output in SPA plots. RATE is the sampling rate of the data, BRK is the break frequency, and ORD is the order.

COMMON/FINF/XARRAY(8): XARRAY contains alphanumeric information about the type of filter selected.

COMMON/STAT/TSTVAL(2,10),STM,ETM,ND,INTRVL(10):

TSTVAL — array used for data editing in the computation of the statistical parameters. Data values from the ith channel less than TSTVAL(1,i) or greater than TSTVAL(2,i) will not be used in computing the various statistical parameters.

STM — start time for statistical analysis (time of day, in total milliseconds).

ETM — end time for statistical analysis (in total milliseconds).

ND — integer disk number. ND=4 for unfiltered data; ND=5 for filtered data.

INTRVL(I) — number of intervals the ith channel will be broken up into for a histogram $1 \leq INTRVL(I) \leq 40$.

COMMON/RESULTS/XMEAN(10),VAR(10),STDEV(10),MEANSQ(10),RMS(10), SKEW(10), KURT(10),XMIN(10),XMAX(10),XNPT(10),AUTO(10,10):

XMEAN(I) — mean of the ith channel

VAR(I) — variance of the ith channel.

STDEV(I) — standard deviation of the ith channel.

MEANSQ(I) — mean square value of the ith channel.

RMS (I) — root mean square of the ith channel.

SKEW (I) — skewness of the ith channel.

KURT (I) — kurtosis estimate of the ith channel.

XMIN (I) — minimum value of the ith channel.

XMAX (I) — maximum value of the ith channel.

XPNT (I) — number of points used in the statistical derivations of the ith channel.

AUTO (I,J) — jth value of the autocorrelation function for the ith channel.

COMMON/FFTINFO/ST,ET,XTYPE,YTYPE,NW,PSDLSTR,PSDLSTS,PSDPLTR, PSTPLTS,CSDLSTR,CSDLSTS,CSDPLTR,CSDPLTS,PHLSTR,PHLSTS,PHPLTR, PHPLTS,COHERL,COHERP,TFLSTR,TFLSTS,TFPLTR,TFPLTS,ARLSTR,ARLSTS, ARPLTR,ARPLTS,DOF,YMIN,YMAX:

ST — four-word integer array used for the start time of spectrum analysis (HR-MIN-SEC-MILLISEC).

ET — four-word integer array used for end time of spectrum analysis. (HR-MIN-SEC-MILLISEC).

XTYPE — integer variable used to determine frequency.

YTYPE — integer variable used to determine ordinate axis type (choices are log linear and decibel).

NW — number of points in the smoothing window.

PSDLSTR — logical variable for listing unsmoothed power spectral density.

PSDLSTS — logical variable for listing smoothed power spectral density.

PSDPLTR — logical variable for plotting unsmoothed power spectral density.

PSDPLTS — logical variable for plotting smoothed power spectral density.

CSDLSTR — logical variable for listing unsmoothed cross-spectrum amplitude.

CSDLSTS — logical variable for listing smoothed cross-spectrum amplitude.

CSDPLTR — logical variable for plotting unsmoothed cross-spectrum amplitude.

CSDPLTS — logical variable for plotting smoothed cross-spectrum amplitude.

PHLSTR — logical variable for listing unsmoothed phase angle estimate.

PHLSTS — logical variable for listing smoothed phase angle estimate.

PHPLTR — logical variable for plotting unsmoothed phase angle estimate.

PHPLTS — logical variable for plotting smoothed phase angle estimate.

COHERL — logical variable for listing coherence estimate.

COHERP — logical variable for plotting coherence estimate.

TFLSTR — logical variable for listing unsmoothed transfer function estimate.

TFLSTS — logical variable for listing smoothed transfer function estimate.

TFPLTR — logical variable for plotting unsmoothed transfer function estimate.

TFPLTS — logical variable for plotting smoothed transfer function estimate.

ARLSTR — logical variable for listing unsmoothed amplitude ratio estimate.

ARLSTS — logical variable for listing smoothed amplitude ratio estimate.

ARPLTR — logical variable for plotting unsmoothed amplitude ratio estimate.

ARPLTS — logical variable for plotting smoothed amplitude ratio estimate.

DOF — integer specifying minimum number of degrees of freedom desired for frequency domain statistical estimate.

YMIN — minimum value for ordinate axis of power spectral density or cross-spectrum density plot. May be used to force many different parameters to have the same scale.

YMAX — maximum value for ordinate axis of power spectral density or cross-spectrum density plot. May be used to force many different parameters to have the same scale.

COMMON/STUFF/PSD(1024),SMOPSD(1024): The common area /STUFF/ is used as a device for saving core.

COMMON/ACTIME/AST(4),AET(4): AST and AET are the actual start and stop times for the spectrum analysis.

COMMON/TRAR/A(2048),B(2048),AA(2048),BB(2048),TYME(2),MILLI,U(50): This common area is used to save core; the A, B, AA, and BB arrays are used for doing fourier transforms.

COMMON/LBL/T,ISMO,XNW: T is the bandwidth of the estimate; ISMO is an integer used to select the type of smoothing window; and XNW is the number of points in the window for plotted output.

COMMON/FRQ/FREQ(1024): FREQ is the array of frequency (abscissa) points for spectrum plots.

COMMON/CSPSD/PSDIN(1024),PSDOUT(1024): PSDIN is the power spectral density estimate of the input channel, and PSDOUT is the power spectral density estimate of the output channel. They are computed in subroutine FFTCSD and passed to subroutine CSDINF, where they are used to derive other spectrum parameters.

COMMON/TRAR2/COIN(1024),QUADRA(1024): COMMON/TRAR2/ is used to save core.

SPAMAIN.—The following definitions are used in the main program.

NAMELIST/CARDIN/RMS,PSD,FILTER,STATIST,DETREND are all logical variables used for the selection of the SPA options to be exercised.

Subroutine TIMARY uses the following variables:

XMILLI — total milliseconds (real) passed in.

ATIME — array passed out (real). ATIME(1) contains hours; ATIME(2) contains minutes; ATIME(3) contains seconds; and ATIME(4) contains milliseconds of the total millisecond time.

N — integer variable used for truncation purposes in TIMARY.

Subroutine TIMECON uses the following variables:

I — integer input array, consisting of time (four words), I(1) denotes hours, I(2) denotes minutes, I(3) denotes seconds, and I(4) denotes milliseconds.

X — total milliseconds (returned).

Subroutine DTRND uses the following variables, among others. SUMX is the sum of the values of the independent variable (in this case a dummy variable, a counter), and SUMX2 is the sum of the squared values of the independent (dummy) variable. SUMY and SUMY2 are similarly defined for the dependent variables (the data being detrended). SUMXY is the sum of the product of the independent and dependent variables. The A array contains the constants of the linear regression, and the B array contains the coefficients of the independent parameter. Notice that there are as many A and B values as there are data channels in SPA file 4 (NCH (number of channels), in fact).

Subroutine FILTERS uses the variables described in common areas /FLTER/ and /PLTINF/.

Subroutine BUTTER1 is a first-order high-pass/low-pass recursive digital filter routine. One of its salient features is that as many as 10 channels may be filtered simultaneously. The variables ALPHA and COEF come from the derivation of the filter, as does K. Notice that K (in BUTTER1) is a real variable.

Y0 and U0 arrays contain the lagged output and input of the filter, respectively. The variable SGN is set to +1 or -1, depending upon the type of filter (high pass or low pass).

Subroutine BUTTER3 is similar to BUTTER1 except that it is a third-order recursive digital high-pass/low-pass filter. BUTTER3 also filters as many as 10 channels simultaneously; A, B, K, COEF, COEFA, and COEF2A are all coefficients used in the actual filtering process. X1, X2, and X3 contain the lagged values of the input; Y1, Y2, and Y3 contain the lagged values of the output.

Subroutine BUTTER4 is a fourth-order digital filter—high pass, low pass, or bandpass. As in BUTTER1 and BUTTER3, as many as 10 channels may be filtered simultaneously.

XLAGL and YLAGL are the input and output lag values for the low-pass filter, respectively.

XLAGL(7,3) is the third lag value for the seventh data channel, and so on; XLAGH and YLAGH are similarly defined for the high-pass filter. BLOW and BHIGH are the arrays that contain the actual filter coefficients.

XZ is the input to the filter (present value); YZ is the output; and YINT is an intermediate value held for the cascaded bandpass filter.

Subroutine COEF computes coefficients for the fourth-order filter. The P array holds the (complex) pole locations of the filter (within the unit circle). The B array is the array of computed coefficients.

Subroutines RECURH and RECURL are the actual recursion routines for the fourth-order filter. RECURH is the high-pass recursion routine; RECURL is the low-pass routine.

X — input (once again, as many as 10 channels may be filtered simultaneously) to the filter.

Y — output of the filter.

B — array of coefficients.

XLAG — lagged values of the input.

YLAG — lagged values of the output.

Subroutine PRINTO has two important variables—T(2), which contains time in alphanumeric code (two words), and Y (10-word array), which contains the data being printed.

Subroutine OVERLAY plots the filtered and unfiltered data on a common time axis. SPI is the number of samples per interval, and TS is the time scale (seconds per centimeter).

Subroutine DATAPLT plots data from either file 4 or file 5 (that is, filtered or unfiltered data).

U is the variable array for data read in. Z and W are used as (scaled) beginning location coordinates for moving the pen to do plotting. Each point is plotted separately; no arrays are used for the actual plotting. Z and U are used for plotting the points. The variable INDX determines which of the SPA channels is being plotted.

Subroutine RTMNSQ computes only the root mean square and the variance of time domain data; the P array contains the input data (as many as 10 channels). The CHT and CHS arrays contain the sum of the parameter values and the sum of the squared parameter values, respectively. The RMSVAL array contains the computed rms; the VAR array contains the computed variance.

Subroutine HISGRAM is the driving routine for the (time domain) statistical parameter estimations. The important variables for this routine are described in common area /STAT/.

Subroutine STAT5 does the actual parameter computations: mean, variance, mean square, standard deviation, root mean square, data maximum and minimum, skewness, kurtosis, and the first 10 values of the autocorrelation function. Variables THIRD and FOURTH are used in computing the third and fourth moments about the mean. SUMX is the sum of the parameter values, SUMX2 is the sum of the squared parameter values, and so on. The SAVE array contains lagged values of the input. SAVE(I,J) contains the jth lag value of the ith channel.

Subroutine STATOUT does the output for the statistical parameters. The important variables are described in the common area /STAT/.

Subroutine FREQFIL computes the number of observations within the intervals for histogram output. Once again, many of the important variables are described in common area /STAT/. XLEN is the length of the interval in a histogram. FREQ(I,J) holds the number of observations in the jth interval of the ith parameter.

Subroutine HG5 does the actual construction of the histogram. Array CH105 contains the 5 percent critical values for the χ^2-statistic; CH105(I,J) for I=1 contains the low value, for I=2 the high value for J degrees of freedom. The PY array is used in a DO-loop to move the plotter pen to the correct position for printing the various statistical estimates. The array JIM is used in a similar manner to determine the number of decimal places in the output. The E array (two words) is used for scaling the frequency (vertical) axis. The D array (also two words) is used for scaling the data range (horizontal axis). The X, Y, and Z variables (lines 83 to 100) are used to plot the actual histogram. Then the histogram produced is fit to a normal curve with the same first and second moments. The normal curve ordinate values are put in the array ORD. Three hundred values are plotted. The normal curve abscissa values are put in the array ABSS. The variable COEF is the scaling variable (area) that insures that the normal curve plotted has the same area as the histogram. The code in lines 160 to 180 contains the computational algorithm for the χ^2 goodness-of-fit statistic. XNORM(JK) is the average value of the normal curve in the JKth interval.

Subroutine SPECTR is the driver routine for spectrum analysis. CH is the channel number for power spectral density computation; for cross-spectrum computations, CHIN is the input channel and CHOUT is the output channel.

Subroutine FFTPSD computes the power spectral density of the indicated channel. NPT is the number of points being fourier transformed at one time; NPT02 is the number of points in the frequency domain. As mentioned in the discussion of common area /TRAR/, A and B arrays are used to do the fourier transforming. XST and XET are the requested start and stop times (in total milliseconds) for the analysis. The power spectral density array holds the power spectral density estimate. The code between lines 119 and 140 does the frequency averaging to increase the number of degrees of freedom. The A and B arrays are used to do the actual averaging.

Subroutine FFTCSD computes the principal components of the cross-spectrum estimates. As in FFTPSD, XST and XET are the (requested) start and stop times for the analysis (total milliseconds). A, B, AA, and BB arrays are used to do the actual fourier transform, NPT is the number of points being transformed at one time, and NPT02 is the number of points in the frequency domain. COIN and QUADRA are the arrays that hold the coincident and quadrature spectrum estimates, respectively. The array phase holds the phase angle estimates. The code in lines 165 to 189 does the frequency averaging to increase the number of the degrees of freedom. The A, B, AA, and BB arrays are used for the averaging, as is the cross-spectrum density array (line 175). The cross-spectrum density array first holds the averaged quadrature spectrum estimate; on line 182 it is filled with the cross-spectrum amplitude estimate.

Subroutine POINTS computes the number of time domain points to be fourier transformed at one time. This number is called NPT.

Subroutine FACTR modifies the number of points to be fourier transformed to make the number compatible with the fast fourier transform routine. At most, 2048 points will be transformed. Array IPRIME contains the prime numbers from 29 to 1024. (NPT may have no prime factor greater than 23, lines 25 to 38). The IFAC array counts the number of times the prime factors appear in NPT; IFAC(1) is the number of 2's, IFAC(2) is the number of 3's, and so on. Lines 60 to 66 examine the square-free portion of the factors; this must be less than or equal to 210 for the fast fourier transform routine.

Subroutine CSDINF computes many of the cross-spectrum parameters, such as coherence, transfer function, and amplitude ratio. The array SMOPSDI contains the smoothed power spectral density estimate of the input channel; SMOPSDO contains the smoothed power spectral density estimate of the output channel; SMUCSD contains the smoothed estimate of the cross-spectrum amplitude; and SMOANG contains the smoothed estimate of phase angle. The COHER array contains the coherence estimate, AR the amplitude ratio, TF the transfer function, SMOAR the smoothed amplitude ratio, and SMOTF the smoothed transfer function. The myriad logical variables are discussed in the section in the common block, FFTINFO.

Subroutine SMOOTHS smooths a spectrum estimate with any of four different windows. N is the number of points being smoothed, and NW is the number of points in the window. The end points of the estimate are smoothed by extending the

raw estimate by linear regression. SUMX1, SUMY1, SUMX2, SUMY2, SUMX1Y1, and so forth are various accumulated sums for computing the coefficients of the linear regression. A linear regression is done at each end of the curve being smoothed. A1 and A2 are the constants of the linear regression; B1 and B2 are the linear coefficients. The W array holds the weights for the window. The R array holds the lagged values of the raw input. SMOOTH is the output array.

Function COMB computes $\binom{n}{j}$, which equals $\frac{n!}{j!(n-j)!}$. In the routine, TP is the numerator; and BT is the denominator.

Subroutine OUTPT manages the output for the various spectrum routines. The logical variables LR, PR, LS, and PS correspond to listing the raw data, plotting the raw data, listing the smoothed data, and plotting the smoothed data, respectively. The variable INTEG computes the (discrete) integral of the power spectral density. CUM and PCUM are cumulative and percent cumulative power for power spectral densities. The AFREQ array is the input frequency array. According to XTYPE, the FREQ array is then filled as a linear, logarithmic, or decibel transform of AFREQ. The same is true for power spectral density and PPSD. The variables MINFRQ, MINPSD, MAXFRQ, and MAXPSD are used in the call to the logarithmic axis routine.

Subroutine AX90 draws the vertical axes. The important variables are detailed in lines 5 to 11.

Subroutine LABEL puts most of the written annotation on the spectrum estimate plots. ITYPE identifies the type of plot. XINC is used to reposition the pen for successive lines of output.

Subroutine FFT performs the fast fourier transform. NFAC is the number of prime factors. NP, AT, CK, BT, and SK are used for temporary storage. The latter four are used for storing the prime factors in the FFT computation. For a detailed analysis of the way FFT works, see reference 4.

Dryden Flight Research Center
  National Aeronautics and Space Administration
    Edwards, Calif., May 16, 1977

APPENDIX—CONFIDENCE INTERVALS FOR
THE STATISTICAL PARAMETERS

Random Error in the Estimates

The first section of this appendix deals with the computational formulas used in estimating the time and frequency domain parameters. This section deals with the confidence intervals for the various estimates. The number produced by the SPA program is, in some sense, the best estimate of a given parameter (mean

A more meaningful interpretation of the SPA program's output can be obtained by examining an interval within which the true (population) parameter lies (with typically 95 percent or 99 percent confidence). These confidence intervals will be discussed for both time and frequency domain estimates. Confidence intervals indicate the uncertainty attached to a parameter estimate: Hence, they reflect the amount of random error (sometimes also referred to as sampling error) that accompanies the estimate of that parameter. The reason for this error is that a statistical estimate is made from a sample (not the entire population) and generalized. As a general rule, increasing the sample size lowers the random error in all SPA-estimated parameters. This is true of even the frequency domain estimates, as discussed in the next section.

Confidence Intervals for Time Domain Statistical Parameters

The time domain parameters for which confidence intervals will be specified are mean, variance (and hence the standard deviation), mean square (and hence root mean square), skewness, kurtosis, and autocorrelation function.

Mean value.—The mean values, $\overline{X}$, found on the statistical output are, as mentioned above, sample means. A confidence interval for the true (population) parameter, μ_X, can be based on the sample estimates $\overline{X}$ and s_X, by using the t-test as follows.

Let n be the number of samples, $\overline{X}$ the sample mean, s_X the sample standard deviation, $t_{n,\alpha/2}$ the t-score (see table 3), and $(1 - \alpha)$ the confidence level.

$$\overline{X} + \frac{s_X t_{n,\alpha/2}}{\sqrt{n}} \leq \mu_X \leq \frac{\overline{X} + s_X t_{n,1-\alpha/2}}{\sqrt{n}}$$

which is to say that the true (population) mean value falls within the above range with a confidence of $100(1 - \alpha)$ percent. Notice that use of the t-test is predicated on the assumption that the data are approximately normally distributed.

Variance and mean square estimates.—For the variance, a $100(1 - \alpha)$ percent confidence interval can be estimated by letting n be the number of observations,

TABLE 3 —PERCENTAGE POINTS OF STUDENT t-DISTRIBUTION

$$\left[\text{Value of } t_{\nu,\alpha} \text{ such that Prob}\left(t_\nu > t_{\nu,\alpha}\right) = \alpha\right]$$
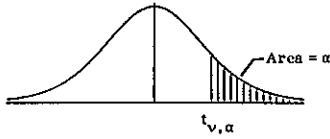
ν	α				
	0 100	0 050	0 025	0 010	0 005
	t-score				
1	3 078	6 314	12 706	31 821	63 657
2	1 886	2 920	4 303	6 965	9 925
3	1 638	2 353	3 182	4 541	5 841
4	1 533	2 132	2 776	3 747	4 604
5	1 476	2 015	2 571	3 365	4 032
6	1 440	1 943	2 447	3 143	3 707
7	1 415	1 895	2 365	2 998	3 499
8	1 397	1 860	2 306	2 896	3 355
9	1 383	1 833	2 262	2 821	3 250
10	1 372	1 812	2 228	2 764	3 169
11	1 363	1 796	2 201	2 718	3 106
12	1 356	1 782	2 179	2 681	3 055
13	1 350	1 771	2 160	2 650	3 012
14	1 345	1 761	2 145	2 624	2 977
15	1 341	1 753	2.131	2 602	2 947
16	1 337	1 746	2 120	2 583	2 921
17	1 333	1 740	2 110	2 567	2 898
18	1 330	1 734	2 101	2 552	2 878
19	1 328	1 729	2 093	2 539	2 861
20	1 325	1 725	2 086	2 528	2 845
21	1 323	1 721	2 080	2 518	2 831
22	1 321	1 717	2 074	2 508	2 819
23	1 319	1 714	2 069	2 500	2 807
24	1 318	1 711	2 064	2 492	2 797
25	1 316	1 708	2 060	2 485	2 787
26	1 315	1 706	2 056	2 479	2 779
27	1 314	1 703	2 052	2 473	2 771
28	1 313	1 701	2 048	2 467	2 763
29	1 311	1 699	2 045	2 462	2 756
30	1 310	1 697	2 042	2 457	2 750
40	1 303	1 684	2 021	2 423	2 704
60	1 296	1 671	2 000	2 390	2 660
120	1 289	1 658	1 980	2 358	2 617

$\alpha = 0\ 995,\ 0\ 990,\ 0\ 975,\ 0\ 950,\ \text{and } 0\ 900 \text{ follow from } t_{\nu,\,1-\alpha} = -t_{\nu,\alpha}$

$s_X{}^2$ the sample variance, and $\chi_{\mu,\alpha}{}^2$ the χ^2-statistic for n degrees of freedom at the α level of confidence. Then

$$\frac{ns_X{}^2}{\chi_{\nu,\alpha/2}{}^2} \leq \sigma_\chi{}^2 \leq \frac{ns_X{}^2}{\chi_{\nu,\,(1-\alpha/2)}{}^2}$$

where $\nu = n - 2$.

The χ^2-value is found from table 4.

TABLE 4 —PERCENTAGE POINTS OF χ^2-DISTRIBUTION

Value of $\chi_{\nu,\alpha}^2$ such that $\text{Prob}\left(\chi_\nu^2 > \chi_{\nu,\alpha}^2\right) = \alpha$, ref 2

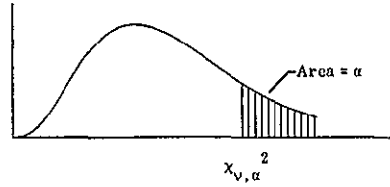| ν	α									
	0 995	0 990	0 975	0 950	0 900	0 10	0 05	0 025	0 010	0 005
	Percentage points for χ^2-distribution									
1	0 000039	0 00016	0 00098	0 0039	0 0158	2 71	3 84	5 02	6 63	7 88
2	0 0100	0 0201	0 0506	0 103	0 211	4 61	5 99	7 38	9 21	10 60
3	0 0717	0 115	0 216	0 352	0 584	6 25	7 81	9 35	11 34	12 84
4	0 207	0 297	0 484	0 711	1 06	7 78	9 49	11 14	13 28	14 86
5	0 412	0 554	0 831	1 15	1 61	9 24	11 07	12 83	15 09	16 75
6	0 676	0 872	1 24	1 64	2 20	10 64	12 59	14 45	16 81	18 55
7	0 989	1 24	1 69	2 17	2 83	12 02	14 07	16 01	18 48	20 28
8	1 34	1 65	2 18	2 73	3 49	13 36	15 51	17 53	20 09	21 96
9	1 73	2 09	2 70	3 33	4 17	14 68	16 92	19 02	21 67	23 59
10	2 16	2 56	3 25	3 94	4 87	15 99	18 31	20 48	23 21	25 19
11	2 60	3 05	3 82	4 57	5 58	17 28	19 68	21 92	24 73	26 76
12	3 07	3 57	4 40	5 23	6 30	18 55	21 03	23 34	26 22	28 30
13	3 57	4 11	5 01	5 89	7 04	19 81	22 36	24 74	27 69	29 82
14	4 07	4 66	5 63	6 57	7 79	21 06	23 68	26 12	29 14	31 32
15	4 60	5 23	6 26	7 26	8 55	22 31	25 00	27 49	30 58	32 80
16	5 14	5 81	6 91	7 96	9 31	23 54	26 30	28 85	32 00	34 27
17	5 70	6 41	7 56	8 67	10 08	24 77	27 59	30 19	33 41	35 72
18	6 26	7 01	8 23	9 39	10 86	25 99	28 87	31 53	34 81	37 16
19	6 84	7 63	8 91	10 12	11 65	27 20	30 14	32 85	36 19	38 58
20	7 43	8 26	9 59	10 85	12 44	28 41	31 41	34 17	37 57	40 00
21	8 03	8 90	10 28	11 59	13 24	29 62	32 67	35 48	38 93	41 40
22	8 64	9 54	10 98	12 34	14 04	30 81	33 92	36 78	40 29	42 80
23	9 26	10 20	11 69	13 09	14 85	32 01	35 17	38 08	41 64	44 18
24	9 89	10 86	12 40	13 85	15 66	33 20	36 42	39 36	42 98	45 56
25	10 52	11 52	13 12	14 61	16 47	34 38	37 65	40 65	44 31	46 93
26	11 16	12 20	13 84	15 38	17 29	35 56	38 88	41 92	45 64	48 29
27	11 81	12 88	14 57	16 15	18 11	36 74	40 11	43 19	46 96	49 64
28	12 46	13 56	15 31	16 93	18 94	37 92	41 34	44 46	48 28	50 99
29	13 12	14 26	16 05	17 71	19 77	39 09	42 56	45 72	49 59	52 34
30	13 79	14 95	16 79	18 49	20 60	40 26	43 77	46 98	50 89	53 67
40	20 71	22 16	24 43	26 51	29 05	51 81	55 76	59 34	63 69	66 77
60	35 53	37 48	40 48	43 19	46 46	74 40	79 08	83 30	88 38	91 95
120	83 85	86 92	91 58	95 70	100 62	140 23	146 57	152 21	158 95	163 65

For $\nu > 120$, $\chi_{\nu,\alpha}^2 \approx \nu\left(1 - \dfrac{2}{9\nu} + z_\alpha\sqrt{\dfrac{2}{9\nu}}\right)^3$ where z_α is desired percentage point for a standardized normal distribution

Confidence intervals for the mean and variance estimates presented here are two-sided; that is, the population parameter is as likely to be greater than as it is to be less than the estimated value.

Skewness and kurtosis.—As mentioned previously in this appendix, skewness and kurtosis are related to the third and fourth moments about the mean, respectively.

In general, skewness is a measure of how the distribution is spread about the mean; positive skewness indicates a distribution skewed on the positive (right) side of the distribution, while negative skewness indicates a distribution skewed on the negative (left) side of the distribution. The skewness of a symmetric distribution is 0. Kurtosis measures the spread of the data. A large kurtosis (a value greater than 3) indicates that the distribution is pointed; that is, that there are more data near the middle of the distribution than in the tails compared with a normal distribution of the same mean and variance. A small kurtosis (a value less than 3) indicates that there are more data in the tails of the distribution than in the middle compared with a normal curve with the same mean and variance. Of course, the kurtosis of a normal distribution is 3.

In general, skewness and kurtosis are used as measures of how closely a given distribution fits a normal distribution. When a distribution is estimated from a sample, skewness and kurtosis are subject to random error. To determine (with 95 percent or 99 percent confidence) whether an observed skewness or kurtosis is from a normal distribution, consult tables 5 and 6, respectively. If, for example,

TABLE 5 — PROBABILITY THAT SKEWNESS
EXCEEDS LISTED VALUE IN POSITIVE
DIRECTION FOR A GIVEN SAMPLE SIZE

n	Skewness with respect to a normal distribution at an α of—	
	0 05	0.01
	Critical values for skewness	
25	0.711	1 061
30	0.661	0 982
35	0 621	0.921
40	0.587	0.869
45	0.558	0.825
50	0.533	0.787
60	0.492	0.723
70	0.459	0 673
80	0.432	0.621
90	0 409	0.596
100	0 389	0 567
125	0.350	0.508
150	0.321	0.464
175	0.298	0.430
200	0.280	0.403
250	0 251	0.360
300	0 230	0.329
400	0.200	0.285
500	0.179	0 255
750	0 146	0.208
1000	0.127	0.180

TABLE 6 —PROBABILITY THAT KURTOSIS EXCEEDS
LISTED VALUES

Kurtosis limits with respect to a normal distribution at an α of—			
0 01		0.05	
Critical values for kurtosis			
Lower limit	Upper limit	Lower limit	Upper limit
2 37	3 98	2.51	3 57
2 42	3.87	2 55	3.52
2 46	3.79	2 59	3 47
2 57	3 60	2.67	3.37
2 68	3 41	2.76	3.26
2.77	3 28	2.83	3.18
2.85	3 17	2 89	3 12

the statistical summary from the SPA program indicates the skewness for 500 points
to be 0.016 and the kurtosis to be 4.23, we may conclude with 99 percent confidence
that the distribution sample is not normal. The skewness is not significantly
different from 0 (0.016 is less than 0.255), but a kurtosis of 4.23 exceeds the
0.01 level value of 3.60. These critical values come from table 5. In other words,
the probability of getting a sample from a normal distribution with kurtosis greater
than 3.60 is less than 1 percent. We may thus conclude with 99 percent confidence
that the distribution sampled is not normal.

Confidence intervals for autocorrelation estimates.—According to reference 7,
the standard deviation of a single estimate of the autocorrelation function is $\sqrt{n-\tau}$
where n is the number of values used in computing the autocorrelation value
corresponding to τ and τ is the lag value. Thus, n is the number of points
minus the lag value for each point of the autocorrelation function.

Suppose the following autocorrelation function were derived for 1000 points from
a series thought to be random:

τ (lag)	$R_X(\tau)$
0	1
1	0.33
2	0.01
3	0.04
4	-0.02
5	-0.08
6	0.06
7	0.02
8	0.09
9	0.02
10	-0.02

The standard deviation $\sqrt{n - \tau}$ would be as follows:

Lag	$R_X(\tau)$	$\sqrt{n - \tau}$
0	1	0.032
1	0 33	0.032
2	0.01	0.032
3	0.04	0.032
4	-0.02	0 032
5	-0.08	0.032
6	0 06	0.032
7	0.02	0.032
8	0.09	0.032
9	0.02	0.032
10	-0.02	0.032

The limits for a single autocorrelation estimate for 95 percent confidence are $R_X(\tau) \pm (0.032)(1.96)$, or $R_X(\tau) \pm 0.063$.

We see that three of the confidence intervals (those corresponding to lag values 1, 5, and 8) do not include zero. Chance would dictate that 5 percent of the 10 values, or at most one of the above intervals, should not include zero; hence we conclude that the series under consideration is not purely random, but rather contains some periodicity.

Confidence Intervals for Frequency Domain Statistical Parameters

Power spectrum.—The normalized standard error, which defines the random portion of the estimation error, is defined as follows:

$$\varepsilon_r = \frac{1}{\sqrt{B_e T}}$$

where B_e is the bandwidth of the estimate and T is the total length of the record. Since the number of degrees of freedom is $n = 2B_e T$, ε_r can be written as $\frac{1}{\sqrt{n/2}}$.

This equation indicates that for an unsmoothed estimate with only two degrees of freedom, the standard error of the estimate is as great as the quantity being estimated:

$$\varepsilon_r \frac{1}{\sqrt{2/2}} = 1$$

This would be unacceptable in most applications. The random error of the power-spectrum estimate is reduced by increasing the number of degrees of freedom by smoothing or by averaging adjacent estimates.

A curious fact about power-spectrum estimates is that, unlike most other statistical estimates, increasing the sample size (increasing the length of the time series) does not increase the number of degrees of freedom of the estimate. The equation on page 46 for power spectral density reveals that as the length of the time series increases, the number of fourier coefficients increases; that is, the resolution of the estimate increases, not the number of degrees of freedom. As implied in the beginning of this section, the SPA program is designed to increase the number of degrees of freedom of an estimate automatically by frequency averaging; hence, in the SPA program, increasing the length of the signal being analyzed will result in an increase in the number of degrees of freedom at the expense of a greater resolution bandwidth than would otherwise be expected.

Since the power spectrum shows the distribution of the variance of a time signal by frequency, the sampling distribution of a smoothed estimate is approximately χ^2 with $n = 2B_e T$ degrees of freedom. Hence, a $100(1 - \alpha)$ percent confidence interval for a power spectral density function, $G_X(f)$, based on an estimate $\hat{G}_X(f)$ with n degrees of freedom is given by the following equation:

$$\frac{n\hat{G}_X(f)}{\chi^2_{n,\alpha/2}} \leq G_X(f) \leq \frac{n\hat{G}_X(f)}{\chi^2_{n,(1-\alpha/2)}}$$

A similar expression holds also for the amplitude of the cross-spectrum estimate.

Coherence function estimate.—The coherence $\gamma_{XY}^2(f)$ of an input/output relationship is estimated as follows:

$$\gamma_{XY}^2(f) = \frac{\left|\hat{G}_{XY}(f)\right|^2}{\hat{G}_X(f)\hat{G}_Y(f)} .$$

where smoothed estimates of the cross spectrum $\hat{G}_{XY}(f)$ and the power spectra $\hat{G}_X(f)$ and $\hat{G}_Y(f)$ are used. Reference 2 suggests that for a number of degrees of freedom greater than 20 and for confidence intervals from 0.35 to 0.95, $\hat{\gamma}_{XY}^2(f)$ can be estimated by using the following transformation:

$$\omega(f) = \frac{1}{2} \ln\left[\frac{1 + \hat{\gamma}_{XY}(f)}{1 - \hat{\gamma}_{XY}(f)}\right]$$

Then $\omega(f) = \text{Tanh}^{-1}\left(\hat{\gamma}_{XY}(f)\right)$ has an approximately normal distribution with mean

$$\mu_{\omega(f)} = \frac{1}{n-2} + \text{Tanh}^{-1}\left(\hat{\gamma}_{XY}(f)\right)$$

and variance

$$\sigma_{\omega(f)}{}^2 = \frac{1}{n-2}$$

where n, once again, is the number of degrees of freedom of the estimate.

Hence the $100(1-\alpha)$ percent confidence interval for coherence, $\gamma_{XY}{}^2(f)$, can be estimated by using the following equation:

$$\text{Prob}\left[Z_{1-\alpha/2} \le \frac{\omega(f) - \mu_{\omega(f)}}{\sigma_{\omega(f)}} \le Z_{\alpha/2}\right]$$

If, for example, coherence at a given frequency, $\gamma_{XY}{}^2(f_0)$, is estimated as 0.52 for 32 degrees of freedom, then

$$\omega(f_0) = \frac{1}{2}\ln\left[\frac{1 + \sqrt{0.52}}{1 - \sqrt{0.52}}\right] = 0.91$$

$$\mu_{\omega(f_0)} = \frac{1}{30} + 0.91 = 0.94$$

$$\sigma_{\omega(f_0)}{}^2 = \frac{1}{30} = 0.03$$

which implies that

$$\sigma_{\omega(f_0)} = 0.18$$

Thus, the 95 percent confidence interval for $\omega(f_0)$ is

$$0.91 + 1.96(0.18) = (0.56, 1.26)$$

That is,

$$0.56 \le \omega(f_0) \le 1.26$$

Now $\omega(f_0)$ is the transformed coherence. We obtain the confidence interval for $\gamma_{XY}{}^2(f_0)$ by inverse transforming, as follows:

$$0.56 = \frac{1}{2}\ln\left[\frac{1 + \gamma_{XY}'(f)}{1 - \gamma_{XY}'(f)}\right]$$

and

$$1.26 = \frac{1}{2} \ln \left[\frac{1 + \gamma_{XY}''(f)}{1 - \gamma_{XY}''(f)} \right]$$

where

$$\gamma_{XY}'(f) < \gamma_{XY}(f) < \gamma_{XY}''(f)$$

with $100(1 - \alpha)$ percent confidence.

The inverse transform is thus

$$\gamma_{XY}(f) = \frac{e^{2\omega(f)} - 1}{e^{2\omega(f)} + 1}$$

so

$$0.51 \leq \gamma_{XY}(f_0) \leq 0.85$$

whence

$$0.26 \leq \gamma_{XY}^{2}(f_0) \leq 0.72$$

We may conclude, with 95 percent confidence, that coherence at f_0 is different from zero and that the degree of linear association between input and output at f_0 is somewhere between 26 percent and 72 percent.

Confidence intervals for phase angle estimate and gain.—The frequency response function $\left| \dot{H}_{XY}(f) \right|^{2}$ is given by the following equation:

$$\hat{H}_{XY}(f) = \hat{G}_{XY}^{2}(f) / \hat{G}_{X}(f)$$

We must then consider this complex quantity as a phase and gain, separately. Notice that, in keeping with the estimators for amplitude of power and cross spectrum, the gain factor follows a distribution that is a ratio of χ^{2}-distributions—that is, an F-distribution.

The F-statistic may be estimated as follows:

$$F(2, n - 2) = \frac{(n - 2)\hat{G}_{X}(f) \left| \hat{H}_{XY}(f) - H_{XY}(f) \right|^{2}}{2\hat{G}_{Y}(f) \left[1 - \gamma_{XY}^{2}(f) \right]}$$

Thus

$$\left| \hat{H}_{XY}(f) - H_{XY}(f) \right| \leq \hat{r}(f)$$

where

$$\hat{r}^2(f) = \frac{2}{n-2} F(2, n-2) \left[1 - \hat{\gamma}_{XY}^2(f)\right] \frac{\hat{G}_Y(f)}{\hat{G}_X(f)}$$

and

n	number of degrees of freedom, $2B_eT$
$F(2, n-2)$	100α percentage point of the F-distribution, with two degrees of freedom in the numerator and $n-2$ degrees of freedom in the denominator
$\hat{G}_X(f)$	power-spectrum estimate of input signal, X, at f
$\hat{G}_Y(f)$	power-spectrum estimate of output signal, Y, at f
$\hat{\gamma}_{XY}^2(f)$	coherence estimate between input, X, and output, Y, at f

The confidence interval for the gain (called amplitude ratio in SPA) is thus

$$\left|\hat{H}_{XY}(f)\right| - \hat{r}(f) \leq \left|H_{XY}(f)\right| \leq \left|\hat{H}_{XY}(f)\right| + \hat{r}(f)$$

The confidence interval for the phase angle $\varphi(f)$ is also given in terms of $\hat{r}(f)$, as follows:

$$\hat{\varphi}(f) - \Delta\hat{\varphi}(f) \leq \varphi(f) \leq \hat{\varphi}(f) + \Delta\hat{\varphi}(f)$$

where

$$\Delta\hat{\varphi}(f) = \sin^{-1}\left[\frac{\hat{r}(f)}{\left|\hat{H}_{XY}(f)\right|}\right]$$

It is clear from the above expressions that the random error in both the gain and the phase of the frequency response functions (measured, in part, by $\Delta\varphi(f)$) is heavily dependent upon the number of degrees of freedom of the estimate and the sample's coherence.

70

Example: Suppose that the following is true for an input signal X and an output signal Y at a frequency of 10 hertz: The power spectral density of the input is 0.1180; the power spectral density of the output is 0.1052; the cross spectrum has an amplitude of 0.0295; the phase angle is -6.228°; the coherence estimate is 0.9404; and the amplitude ratio is 0.9997. Further, each estimate is made with 64 degrees of freedom.

First, the normalized standard error, ε_r, is given by the equation

$$\varepsilon_r = \frac{1}{\sqrt{\frac{64}{2}}} = 0.18$$

The 95 percent confidence interval for the power spectral density of the input signal at 10 hertz is

$$\frac{64(0.1180)}{83\ 30} \leq G_X(f) \leq \frac{64(0.1180)}{40.48}$$

where χ^2-values are for 64 degrees of freedom at the 0.025 and 0.975 levels of significance, respectively. Therefore,

$$0.0907 \leq G_X(f) \leq 0.1866$$

Similarly, the 95 percent confidence intervals for the power spectral density of the output at 10 hertz is as follows

$$0.808 \leq G_Y(f) \leq 0.1663$$

and the amplitude of the cross spectrum is given by the expression

$$0.0227 \leq \left| G_{XY}(f) \right|^2 \leq 0.0466$$

The coherence estimate at $f = 10$ hertz is 0.9404. When transformed by $\omega(f) = \text{Tanh}^{-1}\hat{\gamma}_{XY}(f)$ (a transformation sometimes called Fisher's Z-transform (ref. 2)), the estimate becomes

$$\omega(10) = \frac{1}{2}\ln\left[\frac{1 + \sqrt{0.9404}}{1 - \sqrt{0.9404}}\right] = 2.09$$

Then, since $n = 64$ and $\mu_{\omega(10)} = \frac{1}{62} + 2.09 = 2.11$,

$$\sigma_{\omega(10)}{}^2 = \frac{1}{62} = 0.02$$

and

$$\sigma_{\omega(10)} = 0.13$$

Futhermore,

$$Z_{1-\alpha/2}\sigma_{\omega(10)} + \mu_{\omega(10)} \le \omega(10) \le \mu_{\omega(10)} + Z_{\alpha/2}\sigma_{\omega(10)}$$

or

$$1.86 \le \omega(10) \le 2.36$$

and

$$\hat{\gamma}_{XY}(10) = \mathrm{Tanh}[\omega(10)] = \frac{e^{2\omega(10)} - 1}{e^{2\omega(10)} + 1}$$

Therefore,

$$0.95 \le \gamma_{XY}(10) \le 0.98$$

or

$$0.91 \le \gamma_{XY}{}^2(10) \le 0.96$$

with 95 percent conficence.

The 95 percent confidence interval for phase angle is given by the expression

$$\hat{\phi}(10) - \Delta\hat{\phi}(10) < \phi(10) < \hat{\phi}(10) + \Delta\hat{\phi}(10)$$

where

$$\Delta\hat{\phi}(10) = \sin^{-1}\left[\frac{\hat{r}(10)}{|\hat{H}_{XY}(10)|}\right]$$

and

$$\hat{r}^2(10) = \frac{2}{n-2}\left[F_{2,62;\,0.95}\right]\left[1 - \hat{\gamma}_{XY}{}^2(10)\right]\hat{G}_Y(10)/\hat{G}_X(10)$$

$$= \frac{2}{62}(3.15)\left[1 - 0.9404\right]\left[\frac{0.1052}{0.1180}\right] = 0.0054$$

72

where

$\hat{\gamma}_{XY}^{2}$ coherence estimate

$\hat{G}_{Y}$ power spectrum of the output

$\hat{G}_{X}$ power spectrum of the input

$F_{2,62;0.95}$ F-score for two and 62 degrees of freedom at the 0.95 level of significance

Thus

$$r^{2}(10) = 0.0054$$

Hence

$$\Delta\hat{\varphi}(10) = \sin^{-1}\left[\frac{0.0054}{\left|H_{XY}(10)\right|}\right] = \sin^{-1}\left[\frac{0.0054}{0.9997}\right] = 0.31°$$

so that the 95 percent confidence interval for phase angle is

$$\hat{\varphi}(10) - \Delta\hat{\varphi}(10) \leq \varphi(10) \leq \hat{\varphi}(10) + \Delta\hat{\varphi}(10) - 6.54 < \varphi\ (10) < {}^{-}5.92$$

The value of F was taken from table 7 (from ref. 2). Finally the 95 percent confidence interval for amplitude ratio H(10) is given by the expression
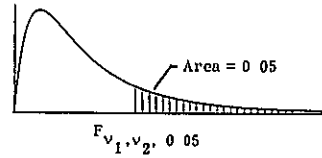
$$\left|\hat{H}_{XY}(10)\right| - r(10) \leq \left|H_{XY}(10)\right| \leq \left|H_{XY}(10)\right| + r(10)$$

or

$$0.9262 \leq \left|H_{XY}(10)\right| \leq 1.073$$

TABLE 7 —PERCENTAGE POINTS OF F-DISTRIBUTION

$$\left[\text{Values of } F_{\nu_1,\nu_2,\ 0\ 05} \text{ such that Prob}\left(F_{\nu_1,\nu_2} > F_{\nu_1,\nu_2,\ 0\ 05}\right) = 0.05\right]$$

Degrees of freedom for denominator, ν_2	Degrees of freedom for numerator, ν_1										∞
	1	2	3	4	5	6	7	8	9	10	
1	161	200	216	225	230	234	237	239	241	242	254
2	18 5	19 0	19 2	19 2	19 3	19 3	19 4	19 4	19 4	19.4	19 5
3	10 1	9 55	9.28	9 12	9 01	8 94	8 89	8 85	8 81	8 79	8.53
4	7.71	6 94	6 59	6 39	6 26	6 16	6 09	6 04	6 00	5 96	5 63
5	6 61	5 79	5 41	5 19	5 05	4 95	4 88	4 82	4 77	4 73	4 37
6	5 99	5.14	4 76	4 53	4 39	4 28	4 21	4 15	4 10	4 06	3 67
7	5 59	4.74	4 35	4.12	3 97	3 87	3 79	3 73	3 68	3 64	3 23
8	5 32	4 46	4 07	3 84	3 69	3 58	3 50	3 44	3.39	3.35	2 93
9	5 12	4 26	3 86	3 63	3 48	3 37	3 29	3 23	3 18	3 14	2 71
10	4.96	4.10	3 71	3.48	3 33	3 22	3 14	3 07	3 02	2 98	2 54
11	4 84	3.98	3 59	3 36	3 20	3 09	3 01	2 95	2 90	2 85	2 40
12	4 75	3 89	3 49	3 25	3 11	3 00	2.91	2 85	2 80	2 75	2 30
13	4 67	3.81	3 41	3 18	3 03	2.92	2 83	2.77	2.71	2.67	2.21
14	4 60	3 74	3 35	3 11	2 96	2 85	2 76	2 70	2 65	2 60	2 13
16	4 49	3.63	3 24	3 01	2 85	2 74	2 66	2 59	2 54	2 49	2.01
18	4 41	3.55	3 16	2 93	2 77	2 66	2 58	2.51	2 46	2 41	1 92
20	4 35	3 49	3.10	2 87	2 71	2 60	2 51	2 45	2 39	2.35	1.84
22	4 30	3.44	3 05	2 82	2 66	2.55	2 46	2.40	2 34	2.30	1 78
24	4 26	3.40	3 01	2.78	2 62	2 51	2 42	2 36	2 30	2 25	1 73
26	4 23	3 37	2 98	2.74	2 59	2 47	2 39	2 32	2 27	2 22	1 69
28	4 20	3 34	2 95	2.71	2 56	2 45	2 36	2 29	2 24	2 19	1 65
30	4 17	3 32	2 92	2.69	2 53	2 42	2 33	2 27	2 21	2 16	1 62
40	4 08	3 23	2 84	2.61	2 45	2 34	2 25	2 18	2.12	2 08	1 51
50	4 03	3 18	2 79	2.56	2 40	2 29	2 20	2 13	2 07	2 03	1 44
60	4 00	3 15	2 76	2 53	2 37	2 25	2 17	2 10	2 04	1 99	1 39
80	3 96	3.11	2 72	2 49	2 33	2 21	2 13	2 06	2 00	1 95	1 32
100	3 94	3.09	2 70	2.46	2 31	2 19	2 10	2 03	1 97	1 93	1 28
200	3 89	3 04	2 65	2.42	2 26	2 14	2 06	1 98	1 93	1 88	1 19
500	3 86	3.01	2.62	2 39	2 23	2 12	2 03	1 96	1 90	1 85	1 11
∞	3 84	3 00	2 60	2 37	2 21	2 10	2 01	1 94	1 88	1 83	1 00

TABLE 7 —Continued

Degrees of freedom for denominator, ν_2	Degrees of freedom for numerator, ν_1										∞
	11	12	13	14	16	18	20	22	24	26	
1	243	244	245	245	246	247	248	249	249	249	254
2	19.4	19 4	19 4	19 4	19 4	19 4	19 5	19 5	19 5	19 5	19 5
3	8.76	8 74	8 73	8 71	8 69	8 67	8 66	8 65	8 64	8 63	8 53
4	5 94	5.91	5 89	5 87	5 84	5 82	5 80	5 79	5 77	5 76	5 63
5	4.70	4 68	4 66	4 64	4.60	4 58	3 56	4 54	4 53	4 52	4 37
6	4 03	4 00	3 98	3 96	3 92	3 90	3 87	3 86	3 84	3 83	3 67
7	3 60	3 57	3 55	3 53	3.49	3 47	3 44	3 43	3 41	3 40	3 23
8	3.31	3 28	3 26	3 24	3.20	3 17	3 15	3 13	3.12	3 10	2 93
9	3 10	3 07	3 05	3 03	2 99	2 96	2 94	2 92	2 90	2 89	2 71
10	2 94	2 91	2.89	2 86	2 83	2.80	2 77	2 75	2.74	2.72	2 54
11	2 82	2.79	2 76	2 74	2 70	2 67	2 65	2 63	2 61	2 59	2 40
12	2.72	2.69	2 66	2 64	2 60	2 57	2 54	2.52	2.51	2 49	2 30
13	2.63	2 60	2 58	2 55	2.51	2 48	2 46	2.44	2 42	2 41	2 21
14	2.57	2 53	2.51	2 48	2.44	2 41	2 38	2 37	2.35	2 33	2 13
16	2 46	2.42	2 40	2 37	2 33	2 30	2 28	2 25	2.24	2 22	2 01
18	2 37	2 34	2 31	2 29	2.25	2 22	2 19	2 17	2 15	2 13	1 92
20	2.31	2.28	2.25	2 22	2.18	2 15	2 12	2 10	2 08	2 07	1 84
22	2.26	2.23	2 20	2 17	2 13	2.10	2 07	2 05	2 03	2 01	1 78
24	2 21	2.18	2 15	2 13	2 09	2 05	2 03	2 00	1.98	1 97	1 73
26	2.18	2 15	2.12	2 09	2 05	2.02	1 99	1 97	1 95	1 93	1 69
28	2.15	2.12	2.09	2 06	2 02	1 99	1 96	1 93	1 91	1 90	1 65
30	2 13	2 09	2 06	2 04	1 99	1 96	1 93	1 91	1 89	1 87	1 62
40	2.04	2 00	1 97	1 95	1 90	1 87	1 84	1 81	1 79	1 77	1 51
50	1.99	1 95	1 92	1 89	1 85	1 81	1 78	1 76	1 74	1 72	1 44
60	1.95	1 92	1 89	1 86	1 82	1 78	1 75	1 72	1 70	1 68	1 39
80	1.91	1 88	1 84	1 82	1.77	1 73	1 70	1 68	1 65	1 63	1 32
100	1 89	1 85	1 82	1 79	1 75	1 71	1 68	1 65	1 63	1 61	1 28
200	1 84	1 80	1 77	1 74	1 69	1 66	1 62	1 60	1 57	1 55	1 19
500	1.81	1 77	1 74	1 71	1 66	1 62	1 59	1 56	1 54	1 52	1 11
∞	1 79	1.75	1 72	1 69	1.64	1 60	1 57	1 54	1 52	1 50	1 00

TABLE 7 —Concluded

Degrees of freedom for denominator, ν_2	Degrees of freedom for numerator, ν_1									∞
	28	30	40	50	60	80	100	200	500	
1	250	250	251	252	252	252	253	254	254	254
2	19 5	19 5	19 5	19 5	19 5	19 5	19 5	19 5	19 5	19 5
3	8 62	8 62	8 59	8 59	8 57	8 56	8 55	8 54	8 53	8 53
4	5 75	5 75	5 72	5 70	5 69	5 67	5 66	5 65	5 64	5 63
5	4 50	4 50	4 46	4 44	4 43	4 41	4 41	4 39	4 37	4 37
6	3 82	3 81	3 77	3 75	3 74	3 72	3 71	3 69	3 68	3 67
7	3 39	3 38	3 34	3 32	3 30	3 29	3 27	3 25	3 24	3 23
8	3 09	3 08	3 04	3 02	3 01	2 99	2 97	2 95	2 94	2 93
9	2.87	2 86	2 83	2 80	2.79	2 77	2 76	2 73	2 72	2.71
10	2 71	2 70	2 66	2 64	2 62	2 60	2 59	2 56	2 55	2 54
11	2.58	2 57	2 53	2 51	2 49	2 47	2 46	2 43	2 42	2.40
12	2 48	2 47	2.43	2 40	2 38	2 36	2 35	2 32	2 31	2.30
13	2 39	2 38	2.34	2 31	2 30	2 27	2 26	2 23	2 22	2 21
14	2.32	2 31	2 27	2 24	2 22	2 20	2 19	2 16	2 14	2 13
16	2 21	2 19	2 15	2 12	2 11	2 08	2 07	2 04	2 02	2.01
18	2.12	2 11	2 06	2 04	2 02	1 99	1 98	1 95	1.93	1 92
20	2 05	2 04	1 99	1.97	1 95	1 92	1 91	1 88	1 86	1.84
22	2 00	1 98	1 94	1 91	1 89	1 86	1 85	1 82	1 80	1 78
24	1 95	1 94	1 89	1 86	1 84	1 82	1 80	1 77	1 75	1 73
26	1.91	1 90	1.84	1 82	1 80	1 78	1 76	1 73	1 71	1 69
28	1 88	1 87	1 82	1 79	1 77	1.74	1 73	1 69	1 67	1.65
30	1.85	1 84	1 79	1 76	1 74	1 71	1 70	1.66	1 64	1 62
40	1.76	1 74	1 69	1 66	1 64	1 61	1 59	1 55	1 53	1 51
50	1.70	1 69	1 63	1 60	1 58	1 54	1 52	1 48	1 46	1.44
60	1.66	1 65	1.59	1 56	1 53	1 50	1 48	1 44	1 41	1 39
80	1 62	1 60	1 54	1 51	1 48	1 45	1 43	1 38	1 35	1 32
100	1.59	1 57	1 52	1 48	1 45	1 41	1 39	1 34	1 31	1 28
200	1 53	1 52	1 46	1 41	1 39	1 35	1 32	1 26	1 22	1 19
500	1.50	1 48	1 42	1 38	1 34	1 30	1 28	1.21	1 16	1 11
∞	1.48	1 46	1 39	1 35	1 32	1 27	1 24	1 17	1 11	1 00

REFERENCES

1. Edwards, John W.; and Deets, Dwain A.: Development of a Remote Digital Augmentation System and Application to a Remotely Piloted Research Vehicle. NASA TN D-7941, 1975.

2. Bendat, Julius S.; and Piersol, Allan G.: Random Data: Analysis and Measurement Procedures. Wiley-Interscience, c.1971.

3. Cooley, James W.; and Tukey, John W.: An Algorithm for the Machine Calculation of Complex Fourier Series. Mathematics of Computation, vol. 19, Apr. 1965, pp. 297-301.

4. Rabiner, Lawrence R.; and Rader, Charles M., eds.: Digital Signal Processing. IEEE Press, c.1972.

5. Wiener, Norbert: The Fourier Integral and Certain of Its Applications. Dover Publications, Inc., 1958.

6. Lighthill, M. J.: Introduction to Fourier Analysis and Generalised Functions. Cambridge University Press, c.1958.

7. Jenkins, Gwilym M.; and Watts, Donald G.: Spectral Analysis and Its Applications. Holden-Day, Inc., c.1968.