C.1

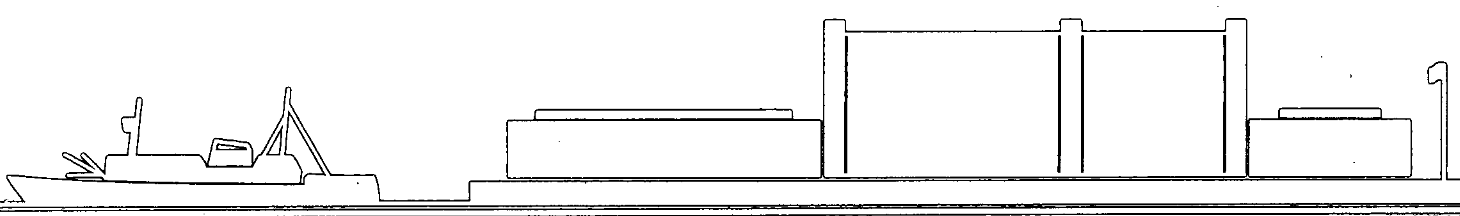
C.C.I.W.
MAY 16 1974
LIBRARY

# CANADA CENTRE FOR INLAND WATERS

PAPER #15. copy 1

# APPLICATION OF CONCEPTS OF THE ANALYSIS OF COMPLEX TIME SERIES TO THE VERIFICATION OF DYNAMIC MODELS OF LAKES

P. F. HAMBLIN , S. BRADLEY



# APPLICATION OF CONCEPTS OF THE ANALYSIS OF COMPLEX TIME SERIES TO THE VERIFICATION OF DYNAMIC MODELS OF LAKES

by

P. F. HAMBLIN

and

S. BRADLEY

CCIW PAPER No. 15 1973

Canada Centre for Inland Waters

Burlington, Ontario

December 1973

TABLE OF CONTENTS

## INTRODUCTION

Recent attempts to verify a variety of numerical hydrodynamical models using field data have indicated the need for a method of determining the phase relationships between pairs of vector time series and between vector and scalar time series. The input or forcing function applied to the numerical model consists of number vector time series such as pressure gradients and surface wind stresses while the output of models consists of scalars such as water level and temperature as well as vector series. Typically, vector output of models are horizontal transports and currents at various depths. In verification analysis it is desirable to compare either the output of models with field data or to determine the phase relationships between field data at various points for comparison with the predicted phase relationships. The approach used is to decompose the variance of the vector time series into a number of frequency bands and to determine the phase relationships in a band between variables using the concepts of spectral and cross-spectral analysis. (Jenkins and Watts, 1968). Moreover, all expressions are developed in terms of familiar spectral and cross-spectral quantities.

## I. THEORY OF COMPLEX TIME SERIES

### A. Analysis of a single vector time series

Consider a vector time series consisting of two oscillatory components, U and V, of frequency, $\sigma$, where U is the component on the east-west direction and V is the component in the north-south direction.

$$U = a_1 \cos\sigma t + b_1 \sin\sigma t$$

$$V = a_2 \cos\sigma t + b_2 \sin\sigma t$$

Let the complex time series, W, be defined:

$$W = U + iV$$

then $$W = 1/2\ [(a_1 + b_2) + i(a_2 - b_1)]e^{i\sigma t} + 1/2\ [(a_1 - b_2) + i(a_2 + b_1)]e^{-i\sigma t}$$

since from complex variable theory $e^{i\sigma t} = \cos\sigma t + i\sin\sigma t$.

It is evident that W may be considered as being composed of a 'positively' rotating frequency component

$$1/2\ [[a_1 + b_2] + i[a_2 - b_1]] = |A^+|e^{i\alpha+}$$

and a 'negatively' rotating frequency component

$$1/2\ [(a_1 - b_2) + i(a_2 + b_1)] = |A^-|e^{i\alpha-}$$

where $$|A^+| = \left(\frac{[a_1 + b_2]^2 + [a_2 - b_1]^2}{4}\right)^{1/2}$$

$$|A^-| = \left(\frac{[a_1 - b_2]^2 + [a_2 + b_1]^2}{4}\right)^{1/2}$$

$$\alpha^- = \tan^{-1} \frac{(a_2 + b_1)}{(a_1 - b_2)}$$

$$\alpha^+ = \tan^{-1} (\ (a_2 - b_1)/(a_1 + b_2)\ )$$

Now W may be written as

$$W = |A^+|e^{i(\sigma t+\alpha^+)} + |A^-|e^{i(\sigma t-\alpha^-)}$$

$$W = e^{\left[\frac{i(\alpha^+ + \alpha^-)}{2}\right]}\left[|A^+|e^{i(\sigma t+1/2(\alpha^+ - \alpha^-))} + |A^-|e^{-i(\sigma t + 1/2(\alpha^+ - \alpha^-))}\right]$$

Now if $W^1$ is W rotated through an angle of $(\alpha^+ + \alpha^-)/2$ then

$$W^1 = (|A^+| + |A^-|)\cos\left[\sigma t + 1/2(\alpha^+ - \alpha^-)\right]$$
$$+ i(|A^+| - |A^-|)\sin\left[\sigma t + 1/2(\alpha^+ - \alpha^-)\right]$$
$$= \text{Maj}\cos\left[\sigma t + 1/2(\alpha^+ - \alpha^-)\right]$$
$$+ i\,\text{Min}\sin\left[\sigma t + 1/2(\alpha^+ - \alpha^-)\right]$$

which is the equation of an ellipse

with major axis $|A^+| + |A^-|$

and minor axis $|A^+| - |A^-|$

Now the spectral equivalents of these quantities will be found.

Let $Pu = (a_1^2 + b_1^2)/2$ , where Pu is the power spectral density of the u component

$Pv = (a_2^2 + b_2^2)/2$ , where Pv is the v component power spectral density

$Puv = (a_1a_2 + b_1b_2)/2$ , where Puv is the cospectrum

and $Quv = (a_1b_2 - a_2b_1)/2$ , where Quv is the quadrature spectrum

Then by substitution

$$|A^+| = 1/2(2Pu + 2Pv + 4Quv)^{1/2}$$

$$= \frac{1}{\sqrt{2}} (Pu + Pv + 2Quv)^{1/2}$$

$$|A^-| = \frac{1}{2} (2Puv + 2Pv - 4Quv \qquad )^{1/2}$$

$$= \frac{1}{\sqrt{2}} (Pu + Pv - 2Quv)^{1/2}$$

The major axis is $|A^+| + |A^-|$

and minor axis is $|A^+| - |A^-|$

The angle of inclination of the semi-major axis to the x axis is

$$\theta = (\alpha^+ + \alpha^-)/2$$

$$\tan 2\theta = \tan(\alpha^+ + \alpha^-) = \frac{\tan \alpha^+ + \tan \alpha^-}{1 - \tan \alpha^+ \tan \alpha^-}$$

$$= \frac{(a_2 + b_1)(a_1 + b_2) + (a_1 - b_2)(a_2 - b_1)}{(a_1 - b_2)(a_1 + b_2) - (a_2 + b_1)(a_2 - b_1)}$$

$$= (2\, a_2 a_1 + 2\, b_1 b_2)/(a_1{}^2 + b_1{}^2 - (a_2{}^2 + b_2{}^2))$$

$$= 2\, Puv/(Pu - Pv)$$

$$\theta = 1/2 \arctan \left[ 2\, Puv/(Pu - Pv) \right]$$

In addition to the major and minor axis and the inclination of the major axis another useful quantity which may be defined is the rotary coefficient,

$$\frac{|A^-|^2 - |A^+|^2}{|A^+|^2 + |A^-|^2} = \frac{-2\, Quv}{Pu + Pv}$$

which represents circularly polarized motion in the clockwise sense for a value of +1, in the counterclockwise sense for -1 and rectilinear motion if it is zero.

B. Analysis of a pair of vector time series

Since the absolute phase angle, $1/2(\alpha^{+} - \alpha^{-})$ cannot be expressed in terms of elementary spectral quantities what remains to be done is to compute the phase differences between two vectors. Consider a vector W as a reference vector and vector W´ as some other vector whose phase with respect to W is unknown.

Let $W(t) = (u(t), v(t))$ where u is the east component of velocity, v is the north component of velocity of W(t).

and $W'(t) = (u'(t), v'(t))$

Define the following co-spectral quantities between u, v, u´ and v´

$$2\ Puu' = a_1 a_1' + b_1 b_1'$$

$$2\ Pvv' = a_2 a_2' + b_2 b_2'$$

$$2\ Puv' = a_1 a_2' + b_1 b_2'$$

$$2\ Pvu' = a_1' a_2 + b_1' b_2$$

which are generalizations of the previously defined quantity Puv, and the quadrature components from Quv are:

$$2\ Quu' = a_1 b_1' - a_1' b_1$$

$$2\ Qvv' = a_2 b_2' - a_2' b_2$$

$$2\ Quv' = a_1 b_2' - a_2' b_1$$

$$2\ Qvu' = a_2 b_1' - a_1' b_2$$

The positive and negative rotating components of W and $W^{1}$ are

$$W^{+} = 1/2\{(a_1 + b_2) + i(a_2 - b_1)\}$$

$$W^{-} = 1/2\{(a_1 - b_2) + i(a_2 + b_1)\}$$

and $$W^{+\prime} = 1/2\{(a_1' + b_2') + i(a_2' - b_1')\}$$

$$W^{-\prime} = 1/2\{(a_1' - b_2') + i(a_2' + b_1')\}$$

Alternatively, since

$$W = e^{i\theta}[|A^{+}|e^{i(\sigma t + \phi)} + |A^{-}|e^{-i(\sigma t + \phi)}]$$

and $$W' = e^{i\theta'}\ [|A^{+\prime}|e^{i(\sigma t + \phi')} + |A^{-\prime}|e^{-i(\sigma t + \phi')}],$$

the cross spectrum between the positively rotating components of W and W´ may be defined as $\langle W^{+} W^{+*\prime} \rangle$ where * denotes the complex conjugate.

$$\langle W^{+} W^{+*\prime} \rangle = |A^{+}||A^{+\prime}|e^{i(\sigma t + \phi) - i(\sigma t + \phi')}e^{i(\theta - \theta')}$$

$$= Pw^{+}w^{+\prime} - iQw^{+}w^{+\prime}$$

$$= |A^{+}|\ |A^{+\prime}|\, e^{i\phi^{++}}$$

Now the phase $\phi^{++}$ between the positively rotating components is

$$\phi^{++} = \arctan(Qw^{+}w^{+\prime}/Pw^{+}w^{+\prime})$$

$$= \theta - \theta' + \phi - \phi'$$

the phase difference $(\phi - \phi')$, between the two vectors, is then $\phi^{++} - (\theta - \theta')$ and the degree to which these components are correlated is given by the coherency squared $,C^{++}$

$$C^{++} = (P^{2}w^{+}w^{+\prime} + Q^{2}w^{+}w^{+\prime})/(Pw^{+}w^{+}\ Pw^{+\prime}w^{+\prime})$$

where $Pw^{+}w^{+} = 1/2(Pu + Pv + 2Quv)$

$$Pw^{+'}w^{+'} = 1/2(Pu' + Pv' + 2Qu'v')$$

and by using the previously defined spectral quantities.

$$Pw^{+}w^{+'} = 1/2(Puu' + Pvv' - Qvu' + Quv')$$

$$Qw^{+}w^{+'} = 1/2(Puv' - Pvu' - Qvv' - Quu')$$

Now if it turns out that the two components $|A^{+}|$ and $|A^{+'}|$ are zero or small then the vector process will be described principally by $|A^{-}|$ and $|A^{-'}|$.

In this case the phase difference $\phi^{--}$ between the two negatively rotating components is

$$\phi^{--} = \arctan(Qw^{-}w^{-'}/Pw^{-}w^{-'})$$

$$\doteq -(\phi - \phi') + \theta - \theta'$$

resulting in the phase difference between the vectors

$$\phi - \phi' = \theta - \theta' - \phi^{--}$$

As before the coherency squared between the negatively rotating components is

$$C^{--} = \frac{P^2w^{-}w^{-'} + Q^2w^{-}w^{-'}}{Pw^{-}w^{-} \quad Pw^{-'}w^{-'}}$$

where $Pw^{-}w^{-} = 1/2(Pu + Pv - 2Quv)$

$$Pw^{-'}w^{-'} = 1/2(Pu' + Pv' - 2Qu'v')$$

$$Qw^{-}w^{-'} = 1/2(Pvu' - Puv' - Qvv' - Quu')$$

$$Pw^{-}w^{-'} = 1/2(Puu' + Pvv' - Quv' + Qvu')$$

If a vector series is composed almost equally of positively and negatively rotating components, the phase difference may also be computed by averaging $\phi^{++}$ and $\phi^{--}$

$$\phi - \phi' = 1/2(\phi^{++} - \phi^{--})$$

or

$$\phi - \phi' = \pm\pi + 1/2 \arctan\left[\frac{Q_{w^+w^{+'}}/P_{w^+w^{+'}} - Q_{w^-w^{-'}}/P_{w^-w^{-'}}}{1 + \frac{Q_{w^+w^{+'}}}{P_{w^+w^{+'}}}\frac{Q_{w^-w^{-'}}}{P_{w^-w^{-'}}}}\right]$$

Note that the averaging process introduces an ambiguity of $\pm\pi$ into the computation of phase angle.

Another case which arises in practice occurs when $|A^-|$ and $|A^{+'}|$ are small or zero. In this case it is necessary to compute $<W^+ W^{-'}>$

$$<W^+ W^{-'}> = |A^+|\,|A^{-'}|\, e^{i(\sigma t + \phi) - i(\sigma t + \phi')} e^{i(\theta + \theta')}$$

$$= P_{w^+w^{-'}} - iQ_{w^+w^{-'}}$$

$$= |A^+|\,|A^{-'}|\, e^{i\phi^{+-}}$$

The phase difference between the two components is

$$\phi^{+-} = \arctan\,(Q_{w^+w^{-'}}/P_{w^+w^{-'}})$$

and the phase difference between the two series is

$$\phi - \phi' = -(\theta + \theta') + \phi^{+-}$$

as before

$$C^{+-} = \left(\frac{P^2_{w^+w^{-'}} + Q^2_{w^+w^{-'}}}{P_{w^+w^+} \quad P_{w^{-'}w^{-'}}}\right)$$

where $P_{w^+w^{-'}} \equiv 1/2(P_{uu'} - P_{vv'} - Q_{uv'} - Q_{vu'})$

$Qw^{+}w^{-\prime} = 1/2(Puv' + Pvu' + Quu' - Qvv')$

and $Pw^{+}w^{+}$ and $Pw^{-\prime}w^{-\prime}$ are defined above

As the final case, consider the cross correlation of $W^{-}$ and $W^{+\prime}$ $\langle W^{-}W^{+\prime}\rangle$.

$$\langle W^{-}W^{+\prime}\rangle = |A^{-}||A^{+\prime}|e^{-i(\sigma t+\phi)+i(\sigma t+\phi')}e^{i(\theta+\theta')}$$

$$= Pw^{-}w^{+\prime} - iQw^{-}w^{+\prime}$$

$$= |A^{-}||A^{+\prime}|\, e^{i\phi^{-+}}$$

The phase difference between $W^{-}$ and $w^{+\prime}$ is $\phi^{-+}$ = arctan $(Qw^{-}w^{+\prime}/Pw^{-}w^{+\prime})$

and the phase difference between the two series is $\phi - \phi'$.

$$\phi - \phi' = \theta + \theta' - \phi^{-+}$$

Finally $C^{-+} = \left(\dfrac{P^{2}w^{-}w^{+\prime} + Q^{2}w^{-}w^{+\prime}}{Pw^{-}w^{-}\quad Pw^{+}w^{+}}\right)$

where $Pw^{-}w^{+\prime} = 1/2(Puu' - Pvv' + Quv' + Qvu')$

and $Qw^{-}w^{+\prime} = 1/2(Puv' + Pvu' - Quu' + Qvv')$

As was computed for the case of correlation between either the positive or the negative components an average phase difference may be computed based on the above expression

$$\phi - \phi' = 1/2(\phi^{+-\prime} - \phi^{-+})$$

$$= 1/2 \arctan\left[\frac{Qw^{+}w^{-\prime}/Pw^{+}w^{-\prime} - Qw^{-}w^{+\prime}/Pw^{-}w^{+\prime}}{1 + \dfrac{Qw^{+}w^{-\prime}}{Pw^{+}w^{-\prime}}\,\dfrac{Qw^{-}w^{+\prime}}{Pw^{-}w^{+\prime}}}\right] \pm \pi$$

In summary, two possible strategies are evident for computing the phase difference between vectors; one is from formulae developed for the correlation of rotating components of the same sign and the other is based on correlation of opposite sign. In practice the first procedure is adopted if the signs of the rotary coefficients are equal while the second if the signs of the rotary coefficients in each of the two vector series are opposite. Once a strategy is adopted two coherences and six angles are computed; the angles between the two rotating components, the estimated phase difference between the two series based on the angle between positive components and the angles of orientation, the estimated phase difference between the two series based on the angle between the negative component and the angles of orientation, the estimated phase angles between the series without information on the angles of orientation and the complement of the last angle listed. Which of these angles to use depends on the particular circumstances.

If the two coherences listed are above the level of significance then the average value would be most appropriate since it does not rely upon the estimates of angles of orientation. Unfortunately, since there is an ambiguity of $\pm 180^{\circ}$ it is necessary to compare the averaged values with the individual values in order to select the appropriate angle. If only one of the coherences is above the level of statistical significance then it is appropriate to select the associated individual phase estimate and to ignore the averaged values.

The case of purely circularly polarized motion presents a special case. Since the angle of orientation of the major axis of the vector ellipse has no meaning in the instance of circularly polarized motion the phase difference is given by the phase angles between the individual rotary components. The averaged angles are of limited usefulness in this case since for circularly polarized motion only one of $C^{++}$ or $C^{--}$ can be significant.

C. Test Cases

Two examples are provided to illustrate the above concepts. Consider two vector time series composed of random noise signals upon which are imposed six sinusoidal fluctuations of differing frequencies, $f_1$, $f_2$, --- , $f_6$

$U$ = Real $W$

$V$ = Imag $W$

$U^1$= Real $W'$

$V^1$= Imag $W'$

where

$$W = \text{Random Noise} + (2\cos(f_1 t) + i\sin(f_1 t))e^{i87.3^\circ}$$
$$+ (1.4\cos(f_2 t - 151^\circ) - .7i\sin(f_2 t - 151))e^{-i65^\circ}$$
$$+ (3\cos(f_3 t - 172^\circ) + i\sin(f_3 t - 172^\circ))e^{-i70.5^\circ}$$
$$+ 4\cos(f_4 t + 120^\circ)$$
$$+ (4.2\cos(f_5 t) - i\sin(f_5 t))e^{-i65^\circ}$$
$$+ \cos(f_6 t - 120^\circ) - 2i\sin(f_6 t - 120^\circ))e^{-i65^\circ}$$

$$W' = \text{Random Noise} + (5.\cos(f_1 t + 130^\circ) - i\sin(f_1 t + 130^\circ))e^{-i70.5^\circ}$$
$$+ (3\cos(f_2 t) + i\sin(f_2 t))e^{-i70.5^\circ}$$
$$+ (3\cos(f_3 t) - i\sin(f_3 t))e^{+i87.3^\circ}$$
$$+ (9\cos(f_4 t) + i\sin(f_4 t))e^{i70.^\circ}$$
$$+ (3\cos(f_5 t-172^\circ) - i\sin(f_5 t-172^\circ))e^{-i70^\circ}$$
$$+ (9\cos(f_5 t) - i\sin(f_6 t))e^{i70^\circ}$$

The random noise component is added to simulate the background noise present in geophysical data series.

The output of the spectral computations by the programme presented in Appendix I for the given input vector series is listed in Table 1A.

Since nearly circularly polarized vectors are frequently encountered in the analysis of lake data an example of this type is included.

$$W = \text{random noise} + (\cos(f_1 t) + i\sin(f_1 t))e^{i87.3°}$$
$$+ (.7(\cos(f_2 t - 151°) - i\sin(f_2 t - 151°))e^{-i65.°}$$
$$+ (\cos(f_3 t - 172°) + i\sin(f_3 t - 172°))e^{-i70.5°}$$
$$+ 4.\cos(f_4 t + 120°)$$
$$+ (\cos(f_5 t\ \ ) - i\sin(f_5 t\ \ ))e^{-i65°}$$
$$2(\cos(f_6 t - 120°) - i\sin(f_6 t - 120°))e^{-i65°}$$
$$W^1 = \text{random noise} + (\cos(f_1 t + 130°) - i\sin(f_1 t + 130°))e^{-70.5°}$$
$$+ (\cos(f_2 t) + i\sin(f_2 t))e^{-i70.5°}$$
$$+ (\cos(f_3 t) - i\sin(f_3 t))e^{i87.5°}$$
$$+ (\cos(f_4 t) + i\sin(f_4 t))e^{i70°}$$
$$+ (\cos(f_5 t - 172°) - i\sin(f_5 t - 172°))e^{-i70.5°}$$
$$+ (\cos(f_6 t) - i\sin(f_6 t))e^{i70°}$$

It is evident that the phase information in the second example is obtained from either columns, $\phi^{+\pm}$ or $\phi^{-\pm}$ and not the other estimates of phase in the case of the circularly polarized vectors (Table 1B) whereas for the elliptically polarized example phase information is contained in last four columns (Table 1A).

The method of analysis is designed to handle one further special case which occurs when one vector component is circularly polarized and when one component is elliptically polarized. For the purpose of the calculation circularly polarized motion is considered to occur whenever the absolute value of the rotary coefficient is greater than 0.9 which corresponds to the case where the major axis is less than 50% larger than the minor axis of the ellipse. The angle of orientation is then set to zero and the calculation proceeds as before. The estimate of the phase difference is obtained from the appropriate columns, $(\phi - \phi')^{+\pm}$ or $(\phi - \phi')^{-\pm}$.

D. Analysis of a scalar time series and a vector time series

Let a scalar series be defined by

$$Z = a_1 \cos\sigma t + b_1 \sin\sigma t$$

$$= (|A^+| + |A^-|)(\cos(\sigma t + \phi)$$

$$= 2|A^+| \cos(\sigma t + \phi)$$

and a complex series $W'$

$$W' = \{(|A^{+'}| + |A^{-'}|)\cos(\sigma t + \phi')$$

$$+ i(|A^{+'}| - |A^{-'}|)\sin(\sigma t + \phi')\}e^{i\theta'}$$

Essentially all the expressions developed for the 2 complex time series carry over to this case with the modifications required by $a_2 = b_2 = \theta = Pvv^1 = Qvv^1 = Qvu^1 = Pvu^1 = Pvv = Puv = Quv = 0$. Not only are the above expressions for the phase angles greatly simplified but also only one strategy need be adopted since $|A^+|$ and $|A^-|$ are equal for a scalar process.

A computer programme which calculates the coherency and phase relations between a scalar and a vector is presented in Appendix II. Two examples of a scalar-vector correlation are provided by the frequency component, $f_4$, for the case of an ellipl.ically polarized vector in Table 1A and for the case of a circularly polarized vector in Table 1B.

## II Application to Model Verification

As an example of possible usefulness of the analysis consider a hypothetical model of nearshore circulation which predicts a certain phase relation between the motions associated with internal waves at an inshore and an offshore location. In order to test the predicted relation appropriate data are selected at each of the two locations and analysed by the above programme.

Hourly observations of current at the depth of 10m were selected at an inshore station, station 41, and at a location approximately 16 km, offshore station 6 for the 21 day period from 15 September to 6 October 1972. Both series were collected in the Oshawa region of Lake Ontario during the intensive field programme known as the International Field Year on the Great Lakes. The locations of the station and description of the programme are given by Blanton (1973).

Figure 1 is a plot of the logarithm of spectral densities of u and v components of nearshore current and the coherence between the components against frequency. A prominence is evident in all three quantities at a period of 16.6 h or a frequency of 0.06 cph. Associated with this peak is a rotary coefficient of 0.94 which indicates nearly circularly polarized motion in the clockwise sense. A similar plot for the offshore current series seen in Figure 2 also indicates th presence of a circularly polarized motion at the same frequency. The rotary coefficient in this case is 0.99.

In Figure 3b coherences between the positive and negative rotating components are plotted for 50 intervals of frequency. The highly significant coherence between the negatively rotating (clockwise) components at a period of 16.7 hr indicates that the motions are correlated between offshore and onshore locations. Furthermore, the phase angle between negatively rotating components shown in Figure 3a are in agreement with synchronous orbital motions of internal waves at the two locations.

A further test which is not presented in the figure between current data was conducted at station 8 during the same period as the previous test. Internal wave motions were highly correlated from the 15m to the 30m depth, but were exactly out of phase with one another.

To illustrate the use of the programme which correlates scalar and vector time series current meter data at station 6 and at a depth of 10 meters were correlated with the depth of the $6^{0}$ isotherm at that station. Depths were calculated by linear interpolation between closely spaced temperature sensors and the current meter spectra have been plotted in Figure 2.

The spectral densities of the isotherm depths are closely related to the current spectra in the lower frequency range. Figure 4b demonstrates the highly significant correlation between the negatively rotating current vector and the isotherm excursion at a period of 16.7 hr. The phase angle at that frequency depicted in Figure 4a indicates that maximum thermocline depths are associated with eastward currents at station 6.

A simple model of the flow which is constrained by continuity can explain the observed phase relation. On the northern half of the lake easterly flow in the upper layer is preceded by one half a cycle of onshore flow in the upper layer and by one half a cycle of offshore flow in the lower layer. Since fluid is increasing in the lower layer until the upper layer current is eastward, the upper layer must thicken and will reach its maximum thickness when the current is due east.

III Computer Programme Description

Due to severe memory restrictions on the 3300, the four-dimensional spectral analysis routine was written in the form of four programmes with intermediate data storage on tape. To simplify submission of jobs these four programmes were stored on disc files. Full listings of the programmes below are given in Appendix I.

The four programmes are:

| | |
|---|---|
| DATA 1 | - reads vector time series data for two station locations and buffers onto tape. |
| CALCATE | - reads from tape, performs Fourier transform on data, calculates four power spectra, and six cospectra and quadrature spectra for all possible pair combinations and writes this result on tape. |
| OUTPUT | - reads from tape, calculates coherence and phase, prints and plots spectral characteristics of each vector series by frequency interval, and writes results on tape. |
| SPECPLØT | - reads tape, calculates and plots coherences and phase angles between vector series against frequency. |

Similarly for the case of a scalar and vector time series four somewhat different programmes are stored on disc files

| | |
|---|---|
| SDATA1 | - read a scalar series from one station location, and a vector time series from a second station. Data are buffered onto tape. |

SALCATE - reads from tape, performs Fourier transform on each of the three series, calculates three power spectra cospectra and quadrature spectra and writes the results on tape.

SUTPUT - reads from tape; calculates coherence and phase, prints and plots spectral characteristics of each of the three series by frequency interval.

SSECPLOT - reads tape 1, calculates and plots coherences and phase angles between the two series against frequency.

Full listings of the above programmes are given in Appendix II.

## IV CONCLUSIONS

A number of statistical quantities describing complex time series have been derived in terms of the standard covariance spectral concepts. Two computer programmes based on the formulas derived herein are described and are applied to a number of test cases of known properties as well as actual pairs of current meter time series. Test results suggest wide horizontal coherency and synchronism of internal waves and that there is a phase reversal across the thermocline occurring in Lake Ontario. Phase relations between current and thermocline oscillations can be determined near the shoreline from simple continuity considerations. It is hoped that the concepts outlined and programmes developed will be of usefulness in the verification of dynamical models of lakes.

## ACKNOWLEDGEMENTS

The programming advice of H. Comba is gratefully acknowledged. J. A. Bull provided the isotherm data. Assistance in proof-reading was performed by D. George. M. Skafel and B. G. Krishnappen made a number of helpful suggestions.

## REFERENCES

BLANTON, J. O. 1973. Nearshore lake currents measured during upwelling and downwelling of the thermocline in Lake Ontario. Unpublished Manuscript.

JENKINS, G. M. and D. G. WATTS. 1968. Spectral analysis and its application. Holden-Day.

TABLE 1A

| equency | Rot W´/W** | $(\theta + \theta')^\circ$ | $(\theta - \theta')^\circ$ | $C^{+\pm}$ | $\phi^{+\pm}$ | $C^{-\pm}$ | $\phi^{-\pm}$ | $(\phi - \phi')^{\circ+\pm}$ | $(\phi - \phi)^{\circ+\pm}$ | $(\phi - \phi')$Ave, | $(\phi - \phi')$Ave$\pm 180^\circ$ |
|---|---|---|---|---|---|---|---|---|---|---|---|
| $f_1$ | -.49 | 17 | -200 | 1.0 | -113.1 | 1.0 | 147.9 | -131.1 | -129.9 | 49.5 | -130.5 |
| $f_2$ | .75 | 230 | 6 | 1.0 | 74.8 | 1.0 | 15.6 | -151.2 | -149.6 | 29.4 | -150.4 |
| $f_3$ | -1. | 17 | 200 | 1.0 | -154.9 | 1.0 | -171.3 | -171.9 | -171.7 | 8.2 | -171.8 |
| $f_4$ | - | 70 | - 70 | 1.0 | 50.1 | 1.0 | 170.2 | 120.1 | 119.8 | -60 | 120 |
| $f_5$ | 1.3 | 230 | 5 | 1.0 | 176.7 | 1.0 | -166.7 | 171.7 | 171.7 | -8.3 | 171.7 |
| $f_6$ | .275 | 5 | 130 | 1.0 | 104.9 | 1.0 | - 15.1 | -120.1 | -119.9 | 60 | -120 |

TABLE 1B

| | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| $f_1$ | -1 | -35 | 320 | 1 | -113 | .06 | -121.8 | -113 | -128.7 | 76.5 | -103.5 |
| $f_2$ | -1. | 32 | - 10 | .1 | 128 | 1.00 | - 15 | -150 | - 15 | 56.4 | -123.6 |
| $f_3$ | -1. | 37 | 270 | 1. | -155 | .26 | -156.5 | -155 | -166.5 | 0.6 | -179.4 |
| $f_4$ | - | - 4 | -360 | 1.0 | 50 | .22 | - 42.5 | - 50 | - 46.5 | 46.3 | -133.7 |
| $f_5$ | 1 | -11 | -310 | .23 | 12.4 | .100 | 166.7 | - 38.6 | 166.7 | 89.5 | - 90.5 |
| $f_6$ | 1 | 31 | -280 | .08 | 107.7 | 1.0 | 15 | 22.7 | 15 | 61.3 | -118.7 |

** Ratio of Rotary Coefficients between the two series.

## LIST OF FIGURES

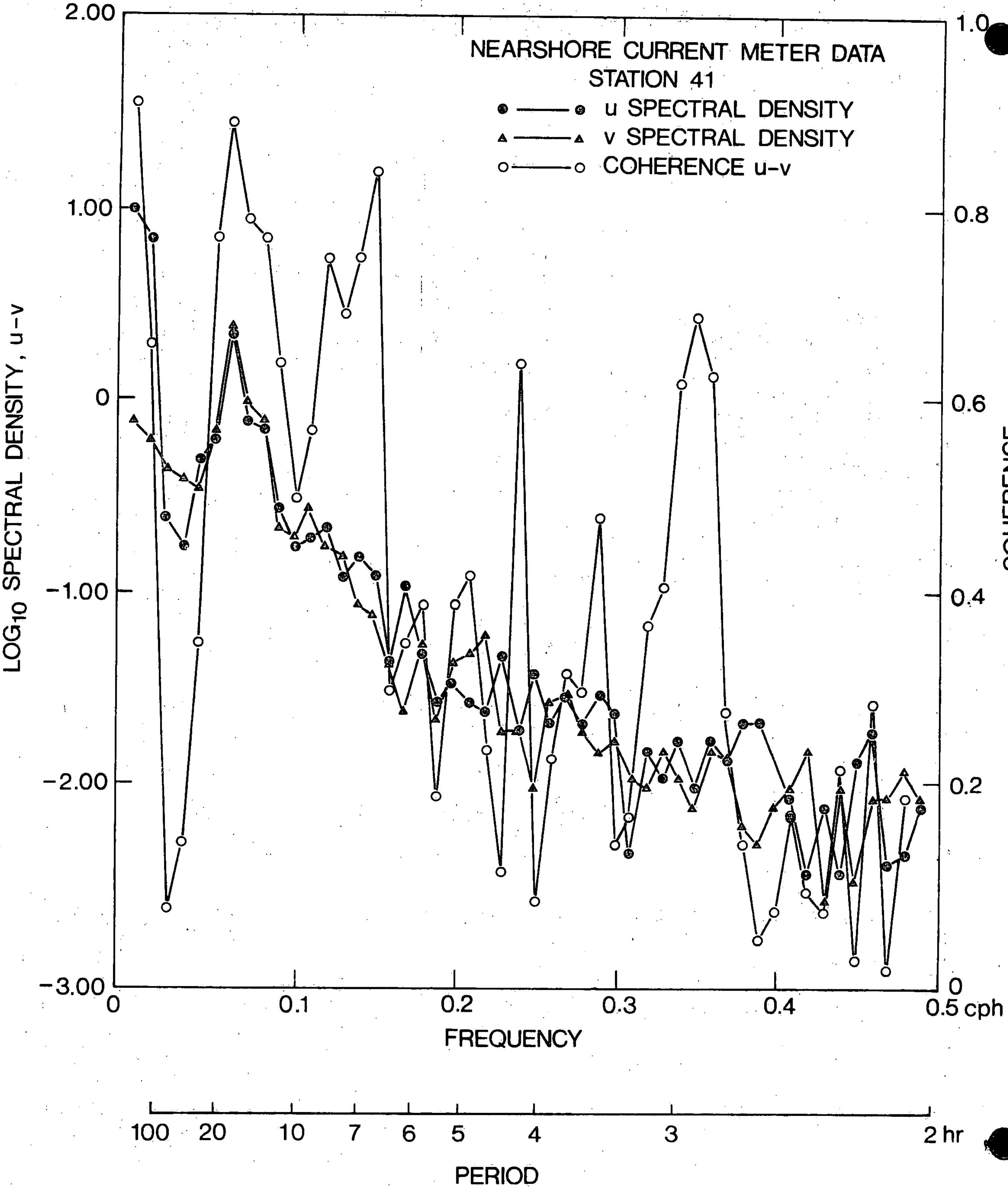

NEARSHORE CURRENT METER DATA
STATION 41
u SPECTRAL DENSITY
v SPECTRAL DENSITY
COHERENCE u-v
2.00
1.00
0
−1.00
−2.00
−3.00
LOG10 SPECTRAL DENSITY, u-v
1.0
0.8
0.6
0.4
0.2
0
COHERENCE
0
0.1
0.2
0.3
0.4
0.5 cph
FREQUENCY
100
20
10
7
6
5
4
3
2 hr
PERIOD

FIGURE.1

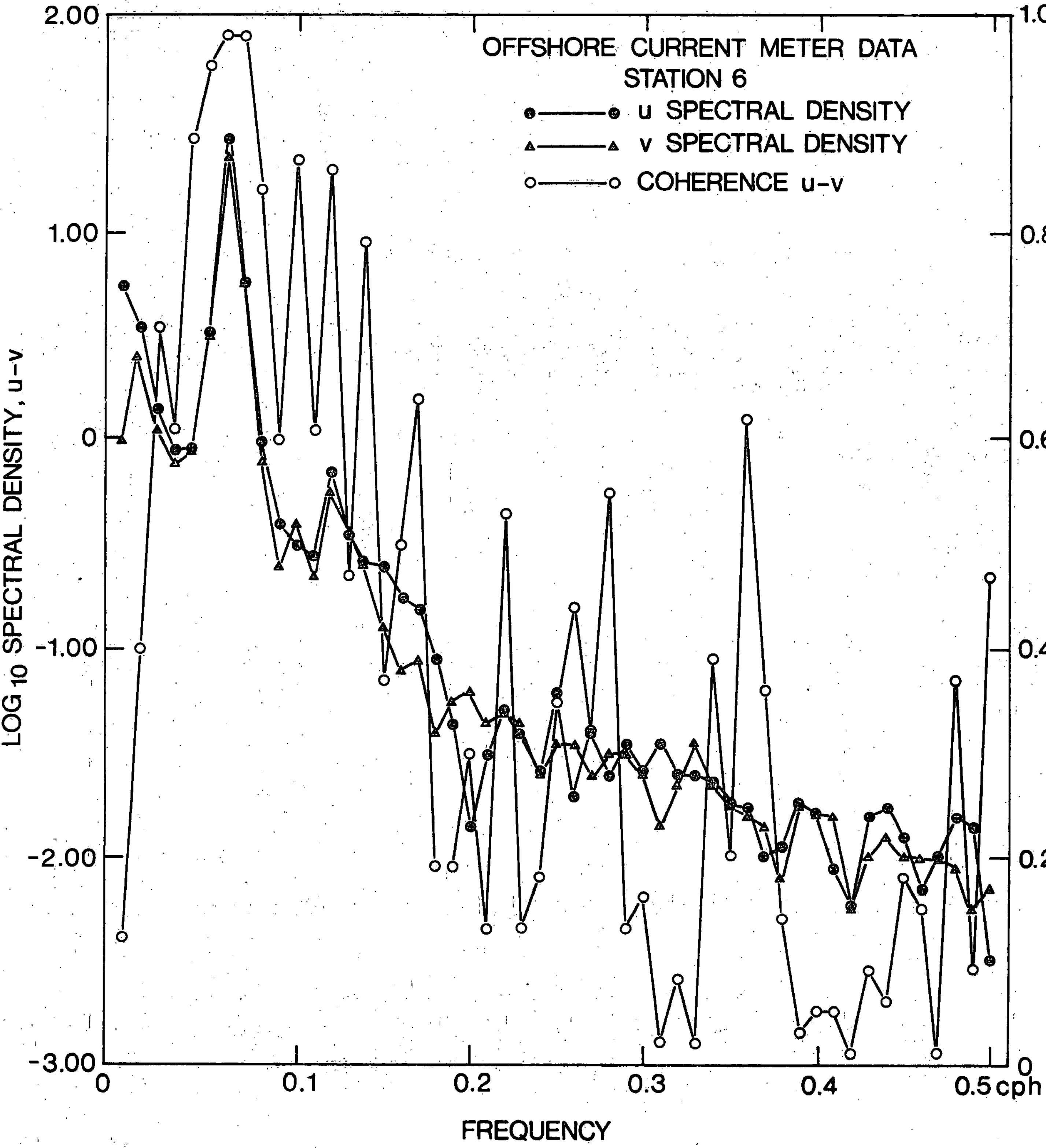
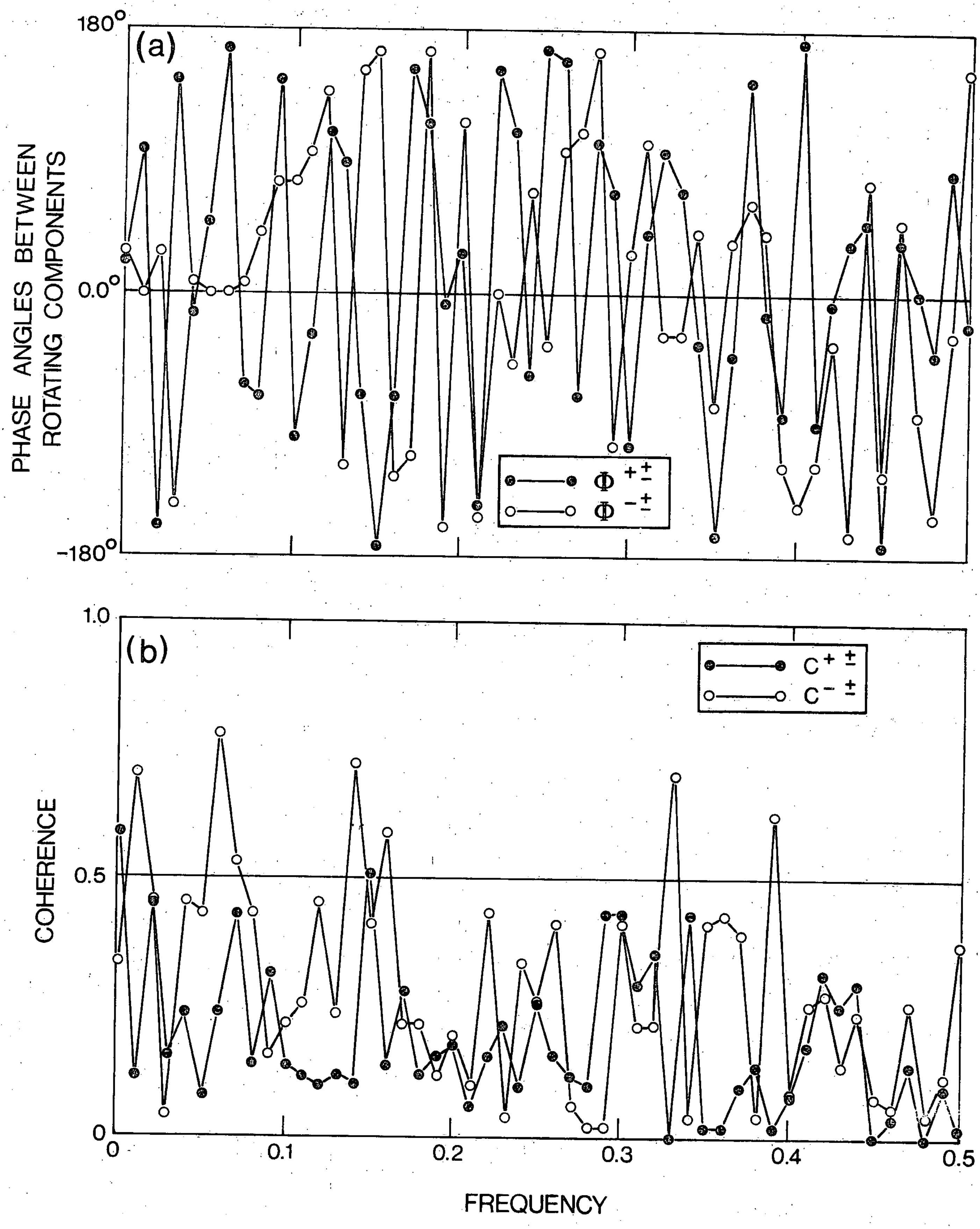
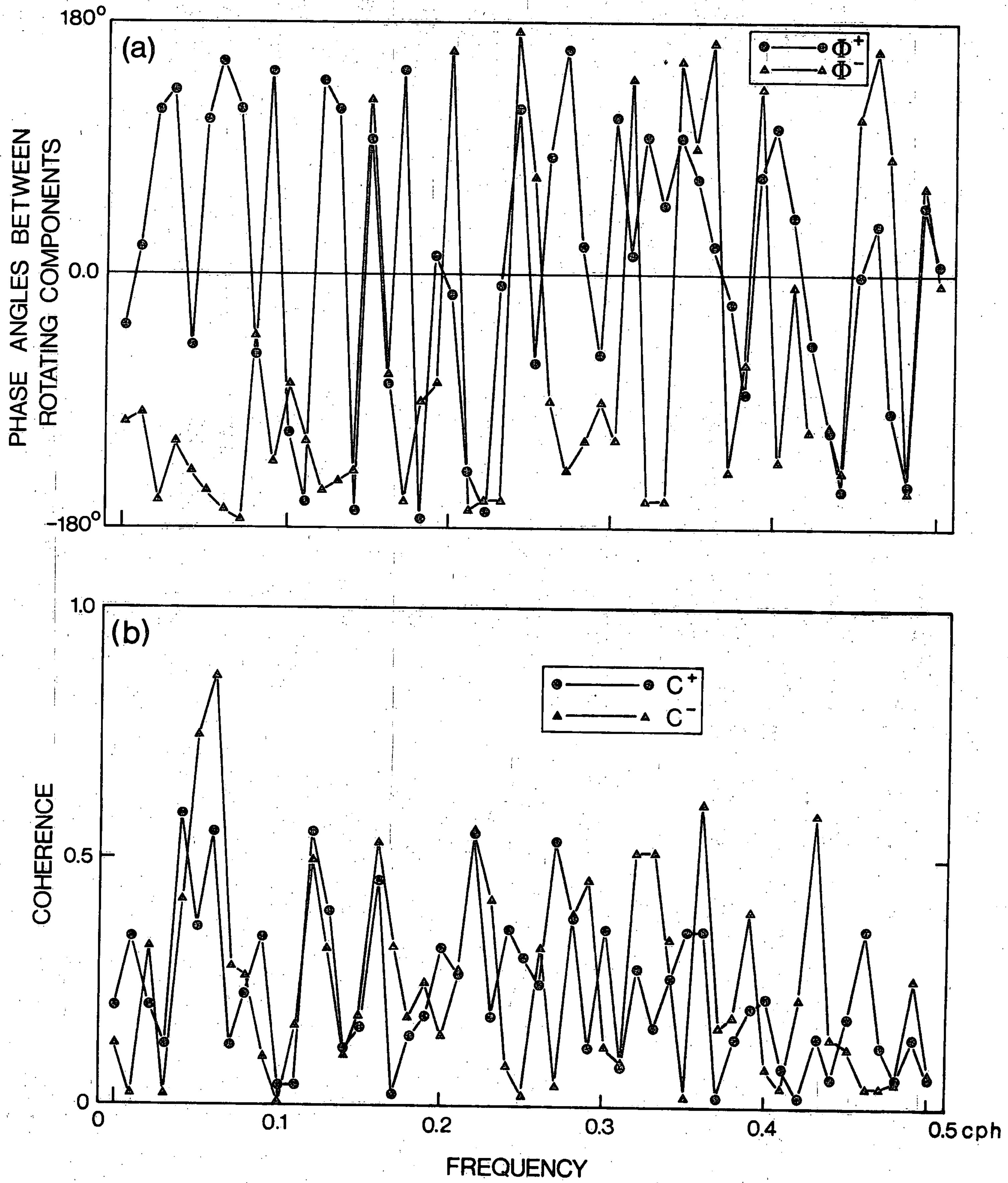

OFFSHORE CURRENT METER DATA
STATION 6
u SPECTRAL DENSITY
v SPECTRAL DENSITY
COHERENCE u-v
LOG 10 SPECTRAL DENSITY, u-v
COHERENCE
FREQUENCY
2.00
1.00
0
-1.00
-2.00
-3.00
1.0
0.8
0.6
0.4
0.2
0
0.1
0.2
0.3
0.4
0.5 cph

FIGURE. 2



180°
(a)
0.0°
-180°
PHASE ANGLES BETWEEN
ROTATING COMPONENTS
$\Phi^{+\pm}$
$\Phi^{-\pm}$
1.0
(b)
0.5
0
COHERENCE
$C^{+\pm}$
$C^{-\pm}$
0
0.1
0.2
0.3
0.4
0.5
FREQUENCY


FIGURE . 3



180°
(a)
Φ+
Φ−
PHASE ANGLES BETWEEN ROTATING COMPONENTS
0.0
−180°
1.0
(b)
C+
C−
COHERENCE
0.5
0
0.1
0.2
0.3
0.4
0.5 cph
FREQUENCY


FIGURE. 4

# APPENDIX I

Programme Listing for vector-vector correlations

DATE

**NAME** Calcate

**PURPOSE**

Reads two vector time series, removes trends, calculates Fourier coefficients, power spectra and cross spectra

**DESCRIPTION**

Calcate is the main driver routine for the spectral computations

**USAGE**

**INPUT**

X contains four scalar time series

**OUTPUT**

P four power spectra
C six cospectra
Q six quadrature spectra

**SUBROUTINES USED**

Spectr which in turn calls Fscran

**CALLED BY**

Data 1

**STORAGE REQUIREMENTS**

**SOURCE LANGUAGE**

**MACHINE**

**PROGRAMMER**

**REFERENCES**

```
      PROGRAM CALCATE
C
C
      COMMON XM1,XM2,LAG    ,FREPAR
      COMMON NAUG,NPTS
      COMMON X(1026,4),P(101,4)
      COMMON C(101,6),Q(101,6),CO(101,2),PH(101)
      REWIND 2
      BUFFER IN(2,1)(X(1),PH(101))
      IF(IFUNIT(2))1,2,1
 2    CONTINUE
      WRITE(6,753)NPTS
 753  FORMAT(1H1,I5)
      WRITE(6,701)XM1,XM2,LAG
 701  FORMAT(1H1/////,17H MOORING NUMBERS    ,A8,6H  AND  ,A8,///
     $5H LAG=,I4)
      CALL SPECTR(NPTS)
 1    REWIND 2
      STOP
      END
```

USASI FORTRAN DIAGNOSTIC RESULTS FOR CALCATE

NO ERRORS

```
      SUBROUTINE SPECTR(NUM)
      COMMON XM1,XM2,LAG    ,FREPAR
      COMMON NAUG,NPTS
      COMMON X(1026,4),P(101,4)
      COMMON C(101,6),Q(101,6),CO(101,2),PH(101)
      CALL ZEROV(P,1919)
      WRITE(6,6699)XM1,XM2
 6699 FORMAT(1H1,///5X,A8,5X,A8/////)
      NCH=4
      ISKIP=0
  800 CONTINUE
      CALL FSCRAN(X,NUM,LAG,P,CO,PH,ISKIP,NCH,C,Q)
      REWIND 2
      BUFFER OUT(2,1)(XM1,PH(101))
      IF(IFUNIT(2))901,902,901
  902 ENDFILE 2
      REWIND 2
  901 RETURN
      END
```

USASI FORTRAN DIAGNOSTIC RESULTS FOR SPECTR

NO ERRORS

UNREFERENCED STATEMENT LABELS

00800

DATE

**NAME** Fscran

**PURPOSE**

Filters out trends, calls routines for spectral analysis and corrects output for trend removal

**DESCRIPTION**

Calls Taper which applies a cosine taper to 4 series after mean is removed by Avvar and trend removed. Zefill augments series to multiples of 2 in length. Rlfort calculates the Fourier coefficients. Pspec calculates the four power spectra. Finally Xspec calculates co and quadrature spectra for all pairs.

**USAGE**

| INPUT | OUTPUT |
|---|---|
| X, 4 time series | P four power spectra<br>C six cospectra<br>Q six quadrature spectra |
| **SUBROUTINES USED** | **CALLED BY**<br>Spectra |
| **STORAGE REQUIREMENTS** | **SOURCE LANGUAGE** |
| **MACHINE** | **PROGRAMMER** |

**REFERENCES**

```
      SUBROUTINE FSCRAN(X,NPTS,NB,P,COH,PH,ISKIP,NCH,C,Q)
      COMMON XXX,XXY,LLL     ,FREPAR
      COMMON NAUG,KIT
      DIMENSION X(1026,4),P(101,4),C(101,6),Q(101,6),COH(101,2)
      DIMENSION PH(101)
      DIMENSION S(513),AVG(4),VAR(4),SUM(4)
      KR=5
      LP=6
      KIT=NPTS
      NBP1=NB+1
      IF(ISKIP.EQ.1)GO TO 450
      IND=0
      DO 135 ICH=1,NCH
      CALL AVVAR(X(1,ICH),NPTS,AVG(ICH),VAR(ICH))
      IF(ICH.LT.4)GO TO 135
      WRITE(LP,120)
  120 FORMAT( /,9X,13H   MEAN       , 3X,4HVAR ,19X,4HMEAN, 8X,4HVAR
     $2(19X,4HMEAN,8X,4HVAR   )/)
      WRITE(LP,130)(AVG(I),VAR(I),I=1,4)
  130 FORMAT(2X,7HX-COMP,2E11.3,5X  ,7HY-COMP,2E11.3,5X,8HXP-COMP,
     $2E11.3,5X,8HYP-COMP,2E11.3)
  135 CONTINUE
      DO 145 ICH=1,NCH
      DO 140 I=1,NPTS
      X(I,ICH)=X(I,ICH)-AVG(ICH)
  140 CONTINUE
      AVG(ICH)=0.
  145 CONTINUE
      FAC=.95
      DEL=0
      DAL=0
      DIL=0.
      DUL=0.
      DO 146 I=2,NPTS
      DEL2=FAC*DEL+.05*X(I,1)
      DEL3=FAC*DAL+.05*X(I,2)
      DEL4=FAC*DIL+.05*X(I,3)
      DEL5=FAC*DUL+.05*X(I,4)
      X(I-1,1)=X(I-1,1)-DEL
      X(I-1,2)=X(I-1,2)-DAL
      X(I-1,3)=X(I-1,3)-DIL
      X(I-1,4)=X(I-1,4)-DUL
      DAL=DEL3
      DIL=DEL4
      DUL=DEL5
  146 DEL=DEL2
      WRITE(6,25)((X(I,J),J=1,NCH),I=1,100)
   25 FORMAT(3(1X,4E10.3))
C     WRITE(LP,150)
  150 FORMAT(// 30H MEAN REMOVED FROM EACH SERIES   )
  160 CONTINUE
  165 DO 170 ICH=1,NCH
  170 CALL TAPER(X(1,ICH),NPTS)
C     WRITE(LP,180)
```

```
 180  FORMAT(// 44H COSINE TAPER APPLIED TO ENDS OF EACH SERIES
 190  CONTINUE
      DO 200 ICH=1,NCH
 200  CALL ZEFILL(X(1,ICH),NPTS,M,NAUG)
      IF(NAUG.EQ.NPTS) GO TO 220
      WRITE(LP,210)NPTS,NAUG
 210  FORMAT( / 36H EACH SERIES AUGMENTED BY ZEROS FROM I5,8H PTS T
     2       I5,4H PTS )
 220  CONTINUE
      CALL RLFORT(X(1,1),M+1,S,0,IFERR)
      DO 270 ICH=1,NCH
      CALL RLFORT(X(1,ICH),M+1,S,-3,IFERR)
      IF(IFERR.EQ.0) GO TO 270
      WRITE(LP,260) IFERR,ICH
 260  FORMAT(22H FORT ERROR IFERR,ICH=   2I4)
 115  RETURN
 270  CONTINUE
      NEB=NAUG/2
      NEBP1=NEB+1
      DO 290 ICH=1,NCH
      CALL PSPEC(X(1,ICH),NEB,P(1,ICH),NB)
      DO 290 I=1,NBP1
      P(I,ICH)=P(I,ICH)*FLOAT(NAUG)/FLOAT(NPTS)
 290  CONTINUE
 360  FORMAT(/ 4H SUM,13X,4E12.3)
      CALL XSPEC(X(1,1),X(1,2),NEB,C(1,1),Q(1,1),NB)
      CALL XSPEC(X(1,3),X(1,4),NEB,C(1,2),Q(1,2),NB)
      CALL XSPEC(X(1,1),X(1,3),NEB,C(1,3),Q(1,3),NB)
      CALL XSPEC(X(1,1),X(1,4),NEB,C(1,4),Q(1,4),NB)
      CALL XSPEC(X(1,2),X(1,3),NEB,C(1,5),Q(1,5),NB)
      CALL XSPEC(X(1,2),X(1,4),NEB,C(1,6),Q(1,6),NB)
      ZTL=FLOAT(NAUG)/FLOAT(NPTS)
      DO 2825 I=1,NBP1
      DO 2825 J=1,6
      C(I,J)=C(I,J)*ZTL
      Q(I,J)=Q(I,J)*ZTL
 2825 CONTINUE
      SUMC1=0.
      SUMC2=0.
      SUMC3=0.
      SUMC4=0.
      SUMC5=0.
      SUMC6=0.
      SUMQ1=0.
      SUMQ2=0.
      SUMQ3=0.
      SUMQ4=0.
      SUMQ5=0.
      SUMQ6=0.
      DO 460 I=1,NBP1
      SUMC1=SUMC1+C(I,1)
      SUMC2=SUMC2+C(I,2)
      SUMC3=SUMC3+C(I,3)
      SUMC4=SUMC4+C(I,4)
```

```
      SUMC5=SUMC5+C(I,5)
      SUMC6=SUMC6+C(I,6)
      SUMQ1=SUMQ1+Q(I,1)
      SUMQ2=SUMQ2+Q(I,2)
      SUMQ3=SUMQ3+Q(I,3)
      SUMQ4=SUMQ4+Q(I,4)
      SUMQ5=SUMQ5+Q(I,5)
      SUMQ6=SUMQ6+Q(I,6)
  460 CONTINUE
      SUMC=SUMC1
      SUMQ=SUMQ1
      WRITE(6,560) SUM(1),SUM(2),SUM(3),SUM(4),SUMC,SUMQ
      SUMC=SUMC2
      SUMQ=SUMQ2
      WRITE(6,560) SUM(1),SUM(2),SUM(3),SUM(4),SUMC,SUMQ
      SUMC=SUMC3
      SUMQ=SUMQ3
      WRITE(6,560) SUM(1),SUM(2),SUM(3),SUM(4),SUMC,SUMQ
      SUMC=SUMC4
      SUMQ=SUMQ4
      WRITE(6,560) SUM(1),SUM(2),SUM(3),SUM(4),SUMC,SUMQ
      SUMC=SUMC5
      SUMQ=SUMQ5
      WRITE(6,560) SUM(1),SUM(2),SUM(3),SUM(4),SUMC,SUMQ
      SUMC=SUMC6
      SUMQ=SUMQ6
      WRITE(6,560) SUM(1),SUM(2),SUM(3),SUM(4),SUMC,SUMQ
  560 FORMAT(/  4H SUM,13X,6E12.3)
  847 FAC2=1./(1.-.05/1.95)**2
  450 CONTINUE
      DO 203 I=1,NBP1
      AI=I-1
      AL=NB
                        P(I,1)=P(I,1)*FAC2
      P(I,2)=P(I,2)*FAC2
      P(I,3)=P(I,3)*FAC2
      P(I,4)=P(I,4)*FAC2
      DO 942 JA=1,6
      C(I,JA)=C(I,JA)*FAC2
      Q(I,JA)=Q(I,JA)*FAC2
  942 CONTINUE
      IF(I.GT. NB/ 6) GO TO 203
      B=SIN(3.1416*(AI   )/AL)
      A=COS(3.1416*(AI   )/AL)
      IF(I.EQ.1) B=SIN(3.1416/(4.*AL))
      IF(I.EQ.1) A=COS(3.1416/(4.*AL))
      Z1=(.95*(1.-A)*(1.-.95*A)+(.95*B)**2)**2+(.95*B*.05)**2
      Z2=(1.-.95*A)**2+(.95*B)**2
      P(I,1)=P(I,1)*Z2*Z2/Z1 /FAC2
      P(I,2)=P(I,2)*Z2*Z2/Z1 /FAC2
      P(I,3)=P(I,3)*Z2*Z2/Z1/FAC2
      P(I,4)=P(I,4)*Z2*Z2/Z1/FAC2
      ZP5=Z2*Z2/Z1/FAC2
      DO 203 JA=1,6
```

```
      C(I,JA)=C(I,JA)*ZP5
      Q(I,JA)=Q(I,JA)*ZP5
  203 CONTINUE
      DO 280 ICH=1,NCH
      SUM(ICH)=0.
      DO 280 I=1,NBP1
  280 SUM(ICH)=SUM(ICH)+P(I,ICH)
      WRITE(LP,560) (SUM(ICH),ICH=1,NCH)
      GO TO 115
      END
```

USASI FORTRAN DIAGNOSTIC RESULTS FOR F5CRAN

NO ERRORS

UNREFERENCED STATEMENT LABELS

00150 00160 00165 00180 00190

00360 00847

```
          SUBROUTINE AVVAR(X,N,AV,VAR)
          DIMENSION X(2)
          AV=0.
          DO 10 I=1,N
   10     AV=AV+X(I)
          AV=AV/FLOAT(N)
          VAR=0.
          DO 20 I=1,N
          XR=X(I)-AV
   20     VAR=VAR+XR*XR
          VAR=VAR/FLOAT(N)
          RETURN
          END
```

USASI FORTRAN DIAGNOSTIC RESULTS FOR AVVAR

NO ERRORS

```
USASI FORTRAN(1.1)/MSOS 4.2  INTEGER WORD SIZE = 1      PSR 287      DATE

            SUBROUTINE TAPER(X,N)
            DIMENSION X(2)
            PI=3.1415926
            DO 50 I=1,N
            ANG=(FLOAT(I)-.5)*PI*10./FLOAT(N)
            IF(ANG.GE.PI) GO TO 60
            FAC=.5-.5*COS(ANG)
            X(I)=X(I)*FAC
            IUP=N-I+1
      50    X(IUP)=X(IUP)*FAC
      60    RETURN
            END

      USASI FORTRAN DIAGNOSTIC RESULTS FOR TAPER

         NO ERRORS
```

USASI FORTRAN(1.1)/MSOS 4.2   INTEGER WORD SIZE = 1      PSR 287      DATE

```
      SUBROUTINE ZEFILL(X,NRCD,M,NAUG)
      DIMENSION X(2)
      DO 50 J=1,13
      M=J
      N=2**M
      NAUG=2*N
      IF(NAUG-NRCD)50,60,70
   50 CONTINUE
      STOP
   60 RETURN
   70 ISTART=NRCD+1
      DO 80 I=ISTART,NAUG
   80 X(I)=0.0
      RETURN
      END
```

USASI FORTRAN DIAGNOSTIC RESULTS FOR ZEFILL

NO ERRORS

```
      SUBROUTINE RLFORT (C,M,S,IFS,IFERR)
      DIMENSION C(1),S(1)
      N = 2**M
      IF (IABS(IFS).LT.2) CALL FORT (C,M,S,0,IFERR)
      MCPLX = M - 1
      IF (IFS) 10,45,30
   10 CALL FORT (C,MCPLX,S,-2,IFERR)
      CFACT = 1.
      C(N+1) = 0.5*(C(1) - C(2))
      C(N+2) = 0.
      C(1) = 0.5*(C(1) + C(2))
      C(2) = 0.
      GO TO 50
   30 CFACT = -1.
      C(2) = C(1) - C(N+1)
      C(1) = C(1) + C(N+1)
      GO TO 50
   40 CALL FORT (C,MCPLX,S,2,IFERR)
   45 RETURN
   50 K = N/2 - 1
      MSIN = N/4
      C(K+3) = -C(K+3)
      DO 60 I=3,K,2
      IS = (I-1)/2
      IC = MSIN - IS
      SI = S(IS)
      CI = S(IC)*CFACT
      A1 = C(I)
      B1 = C(I+1)
      L = N - I
      A2 = C(L+2)
      B2 = C(L+3)
      C(I) = 0.5*(A1+A2+(B1+B2)*CT-(A1-A2)*ST)
      C(I+1) = 0.5*(B1-B2-(B1+B2)*ST-(A1-A2)*CT)
      C(L+2) = 0.5*(A1+A2-(B1+B2)*CT+(A1-A2)*ST)
   60 C(L+3) = 0.5*(B2-B1-(B1+B2)*ST-(A1-A2)*CT)
      IF(IFS+3)90,70,90
70    C(1)=C(1)*2.
      C(N+1)=C(N+1)*2.
      DO 80 I=4,N,2
80    C(I)=-C(I)
90    CONTINUE
      IF (IFS) 45,45,40
      END
```

USASI FORTRAN DIAGNOSTIC RESULTS FOR RLFORT

NO ERRORS

```
      SUBROUTINE FORT(A,M,S,IFS,IFERR)
      DIMENSION A(1), S(1), K(14)
      EQUIVALENCE (K(13),K1),(K(12),K2),(K(11),K3),(K(10),K4)
      EQUIVALENCE (K( 9),K5),(K( 8),K6),(K(7),K7),(K( 6),K8)
      EQUIVALENCE (K( 5),K9 ),(K( 4),K10),(K( 3),K11),(K( 2),K12)
      EQUIVALENCE (K( 1),K13),( K(1),N2)
      IF(M)2,2,3
    3 IF(M-13) 5,5,2
    2 IFERR=1
    1 RETURN
    5 IFERR=0
      N=2**M
      IF( IABS(IFS) - 1 ) 200,200,10
   10 IF( N-NP )20,20,12
   12 IFERR=1
      GO TO 200
   20 K(1)=2*N
      DO 22 L=2,M
   22 K(L)=K(L-1)/2
      DO 24 L=M,12
   24 K(L+1)=2
      IJ=2
      DO 30 J1=2,K1,2
      DO 30 J2=J1,K2,K1
      DO 30 J3=J2,K3,K2
      DO 30 J4=J3,K4,K3
      DO 30 J5=J4,K5,K4
      DO 30 J6=J5,K6,K5
      DO 30 J7=J6,K7,K6
      DO 30 J8=J7,K8,K7
      DO 30 J9=J8,K9,K8
      DO 30 J10=J9,K10,K9
      DO 30 J11=J10,K11,K10
      DO 30 J12=J11,K12,K11
      DO 30 JI=J12,K13,K12
      IF(IJ-JI)28,30,30
   28 T=A(IJ-1 )
      A(IJ-1)=A(JI-1)
      A(JI-1)=T
      T=A(IJ)
      A(IJ)=A(JI)
      A(JI)=T
   30 IJ=IJ+2
      IF(IFS)32,2,36
   32 FN = N
      DO 34 I=1,N
      A(2*I-1) = A(2*I-1)/FN
   34 A(2*I)=-A(2*I)/FN
   36 DO 40 I=1,N,2
      T = A(2*I-1)
      A(2*I-1) =T + A(2*I+1)
      A(2*I+1)=T-A(2*I+1)
      T=A(2*I)
      A(2*I) = T + A(2*I+2)
```

```
   40 A(2*I+2)= T - A(2*I+2)
      IF(M-1) 2,1  ,50
C     SET FOR L=2
   50 LEXP1=2
C     LEXP1=2**(L-1)
      LEXP=8
C     LEXP=2**(L+1)
      NPL= 2**MT
C     NPL = NP* 2**-L
   60 DO 130 L=2,M
      DO 80 I=2,N2,LEXP
      I1=I + LEXP1
      I2=I1+ LEXP1
      I3 =I2+LEXP1
      T=A(I-1)
      A(I-1) = T +A(I2-1)
      A(I2-1) = T-A(I2-1)
      T =A(I)
      A(I) = T+A(I2)
      A(I2) = T-A(I2)
      T= -A(I3)
      T1 = A(I3-1)
      A(I3-1) = A(I1-1) - T
      A(I3   ) = A(I1 )    - TI
      A(I1-1) = A(I1-1) +T
   80 A(I1)    = A(I1    )    +TI
      IF(L-2) 120,120,90
   90 KLAST=N2-LEXP
      JJ=NPL
      DO 110 J=4,LEXP1,2
      NPJJ=NT-JJ
      UR=S(NPJJ)
      UI=S(JJ)
      ILAST=J+KLAST
      DO 100 I= J,ILAST,LEXP
      I1=I+LEXP1
      I2=I1+LEXP1
      I3=I2+LEXP1
      T=A(I2-1)*UR-A(I2)*UI
      T1=A(I2-1)*UI+A(I2)*UR
      A(I2-1)=A(I-1)-T
      A(I2  )=A(I    ) - TI
      A(I-1) =A(I-1)+T
      A(I)    =A(I)+TI
      T=-A(I3-1)*UI-A(I3)*UR
      T1=A(I3-1)*UR-A(I3)*UI
      A(I3-1)=A(I1-1)-T
      A(I3)    =A(I1   )-TI
      A(I1-1)=A(I1-1)+T
  100 A(I1)     =A(I1)    +TI
  110 JJ=JJ+NPL
  120 LEXP1=2*LEXP1
      LEXP = 2*LEXP
  130 NPL=NPL/2
```

```
      140 IF(IFS)145,2,1
      145 DO 150 I=1,N
      150 A(2*I) =-A(2*I)
      160 GO TO 1
      200 NP=N
          MP=M
          NT=N/4
          MT=M-2
          IF(MT) 260,260,205
      205 THETA=.7853981634
C         THETA=PI/2**(L+1)     FOR L=1
      210 JSTEP = NT
C         JSTEP = 2**( MT-L+1 ) FOR L=1
          JDIF = NT/2
C         JDIF = 2**(MT-L)  FOR L=1
          S(JDIF) = SIN(THETA)
          IF  (MT-2)260,220,220
      220 DO 250 L=2,MT
          THETA = THETA/2.
          JSTEP2 = JSTEP
          JSTEP = JDIF
          JDIF = JDIF/2
          S(JDIF)=SIN(THETA)
          JC1=NT-JDIF
          S(JC1)=COS(THETA)
          JLAST=NT-JSTEP2
          IF(JLAST-JSTEP)250,230,230
      230 DO 240 J=JSTEP,JLAST,JSTEP
          JC=NT-J
          JD=J+JDIF
      240 S(JD)=S(J)*S(JC1)+S(JDIF)*S(JC)
      250 CONTINUE
      260 IF(IFS)20,1,20
          END
```

USASI FORTRAN DIAGNOSTIC RESULTS FOR FORT

NO ERRORS

UNREFERENCED STATEMENT LABELS

00060    00140    00160    00210

USASI FORTRAN(1.1)/MSOS 4.2 INTEGER WORD SIZE = 1 PSR 287 DATE

```
      SUBROUTINE PSPEC(A,N,P,M)
      DIMENSION A(2),P(2)
      REAL L
      FN=N
      FM=M
      W=FN/FM
      MP1=M+1
      DO 100 INDL=1,MP1
      L=INDL-1
      XA=AMAX1(0.0,W*(L-0.5))
      XB=AMIN1(FN,W*(L+0.5))
      IXA=XA
      IXA=MAX0(0,IXA)
      IXAIND=IXA+1
      IXB=XB+1.0
      IXB=MIN0(IXB,N)
      IXBIND=IXB+1
      FINT=0.0
      DO 50 IXIND=IXAIND,IXBIND
      IX=IXIND-1
      R=A(2*IXIND-1)
      S=A(2*IXIND)
      AMP=R*R+S*S
      IF(IX.LE.IXA+1  .OR.  IX.GE.IXB-1) GO TO 40
      FINT=FINT+AMP
      GO TO 50
   40 XAA=AMAX1(XA,FLOAT(IX)-0.5)
      XBB=AMIN1(XB,FLOAT(IX)+0.5)
      DELX=AMAX1(0.0,XBB-XAA)
      FINT=FINT+AMP*DELX
   50 CONTINUE
      P(INDL)=FINT*0.5
  100 CONTINUE
      RETURN
      END
```

USASI FORTRAN DIAGNOSTIC RESULTS FOR PSPEC

NO ERRORS

USASI FORTRAN(1.1)/MSOS 4.2  INTEGER WORD SIZE = 1      PSR 287      DATE

```
      SUBROUTINE XSPEC(A,B,N,C,Q,M)
      REAL L
      DIMENSION A(2),B(2),C(2),Q(2)
      FN=N
      FM=M
      W=FN/FM
      MP1=M+1
      DO 100 INDL=1,MP1
      L=INDL-1
      XA=AMAX1(0.0,W*(L-0.5))
      XB=AMIN1(FN,W*(L+0.5))
      IXA=XA
      IXA=MAX0(0,IXA)
      IXAIND=IXA+1
      IXB=XB+1.0
      IXB=MIN0(IXB,N)
      IXBIND=IXB+1
      FINTC=0.0
      FINTQ=0.0
      DO 50 IXIND=IXAIND,IXBIND
      IX=IXIND-1
      RA=A(2*IXIND-1)
      SA=A(2*IXIND)
      RB=B(2*IXIND-1)
      SB=B(2*IXIND)
      AMPC=RA*RB+SA*SB
      AMPQ=RA*SB-RB*SA
      IF(IX.LE.IXA+1  .OR.  IX.GE.IXB-1) GO TO 40
      FINTC=FINTC+AMPC
      FINTQ=FINTQ+AMPQ
      GO TO 50
   40 XAA=AMAX1(XA,FLOAT(IX)-0.5)
      XBB=AMIN1(XB,FLOAT(IX)+0.5)
      DELX=AMAX1(0.0,XBB-XAA)
      FINTC=FINTC+AMPC*DELX
      FINTQ=FINTQ+AMPQ*DELX
   50 CONTINUE
      C(INDL)=FINTC*.5
      Q(INDL)=FINTQ*.5
  100 CONTINUE
      RETURN
      END
```

USASI FORTRAN DIAGNOSTIC RESULTS FOR XSPEC

NO ERRORS

TIME     001.98 MIN

DATE

**NAME** Output

**PURPOSE**

Calculates a number of statistical quantities for individual vector series and plots coherence and power.

**DESCRIPTION**

Power, co and quadrature spectra are read in and from these the coherence and phase are calculated from Polar for each series, and from Nfacts the angle of orientation of the vector ellipse for each vector the major and minor axes and the rotary coefficient, RC. These values are printed by frequency interval and the X & Y power spectral densities and the coherence between X & Y components are plotted on the line printer.
If RC is less than 0.9 the sum and difference of the angles of orientation are calculated.

**USAGE**

| INPUT | OUTPUT |
|---|---|
| Power spectra P<br>Cospectra C<br>Quadrature spectra Q | Rotary coefficients ROT<br>Positive rotary spectra PPP<br>Negative rotary spectra PMM<br>Difference between angles of orientation is Daning<br>Sum of angles of orientation is Sumang |
| **SUBROUTINES USED** | **CALLED BY** |
| Move (Library)<br>Polar, NFACTS<br>Kange, KFIX, SGN | Data 1 |
| **STORAGE REQUIREMENTS** | **SOURCE LANGUAGE** |
| **MACHINE** | **PROGRAMMER** |

**REFERENCES**

KANGE converts an angle to range 0-360
KFIX converts angle in radians to integer degrees
SIGN - determines sign of variable

```
      PROGRAM OUTPUT
C
      COMMON XM1,XM2,LAG      ,FREPAR
      COMMON NAUG,NPTS
      COMMON X(1046,4),P(101,4),C(101,6),Q(101,6),PMM(101,2),PPP(101,2)
      COMMON HANING(101),ROT(101,2),SUMANG(101)
      COMMON FRQ(101)     ,CO(101,6),PH(101,6)
      INTEGER PLOTLN(100)
      EQUIVALENCE(NB,LAG),(L1,NBP1)
      DIMENSION SUM(4)
      LP=6
      NCH=4
      IKK=0
      REWIND 2
      BUFFER IN(2,1)(XM1,Q(101,6))
      IF(IFUNIT(2))1,801,1
  801 CONTINUE
      WRITE(6,701)XM1,XM2,LAG
  701 FORMAT(1H1/////,17H MOORING NUMBERS     ,A8,6H  AND  ,A8,///
     $5H LAG=,I4)
      CALL POLAR(C(1,1),Q(1,1),P(1,1),P(1,2),LAG,CO(1,1),PH(1,1))
      REWIND 2
      WRITE(6,603)
      WRITE(6,604)
  800 CONTINUE
      IKK=IKK+1
      WRITE(6,601)
      L1=LAG+1
      DO 2 I=1,L1
      AL=LAG
      AI=I-1
      FREQ=AI/AL*0.5*FREPAR
      IF(AI)3,3,4
    3 ALFREQ=0.
      PERIOD=0.
      GO TO 5
    4 PERIOD=1./FREQ
      ALFREQ=ALOG10(FREQ)
    5 IF(P(I,1))6,6,7
    6 ALPX=0.
      GO TO 8
    7 SX=P(I,1)
      ALPX=ALOG10(SX)
    8 IF(P(I,2))9,9,10
    9 ALPY=0.
      GO TO 11
   10 SX=P(I,2)
      ALPY=ALOG10(SX)
      J=IKK
   12 CO(I,J)=CO(I,J)**2
      CALL NFACTS(P(I,1),P(I,2),C(I,1),Q(I,1),AMAJ,AMIN,AP,AM,THETA,RC,P
     1P,PM)
      IF(IKK.LT.3)PPP(I,IKK)=PP
      IF(IKK.LT.3)PMM(I,IKK)=PM
```

```
      KANGLE=KANGE(KFIX(THETA))
      ARC=ABS(RC)
      ANGLE=KANGLE
      IF(ARC.GT.0.9) ANGLE=0.0
      IF(IKK.EQ.2) SUMANG(I)=DANING(I)+ANGLE
      IF(IKK.EQ.2) DANING(I)=DANING(I)-ANGLE
      IF(IKK.EQ.1) DANING(I)= ANGLE
   11 WRITE(6,602)
     $     AI,FREQ,ALFREQ,PERIOD,P(I,1),ALPX,P(I,2),ALPY,C(I,1),Q(I,1),
     $CO(I,IKK),PH(I,IKK),AMAJ,AMIN,RC,KANGLE
      FRQ(I)=PERIOD
      ROT(I,IKK)=RC
      P(I,1)=ALPX
      P(I,2)=ALPY
      CO(I,IKK)=6.*CO(I,IKK)-3.
    2 CONTINUE
      WRITE(6,803)
  803 FORMAT(1H1)
      XMIN=-3
      XMAX=3
C-    A PLOT IS WANTED
      SCALFAC=(XMAX-XMIN)/100.
      WRITE(61,23)
   23 FORMAT(1H1)
      WRITE(61,88)SCALFAC
   88 FORMAT(40X,15HSCALE FACTOR = ,F8.2)
      WRITE(61,82)XMIN,XMAX
   82 FORMAT(11X,F6.2,97X,F5.2/)
      WRITE(61,79)
   79 FORMAT(14X,102(1H+))
      AL = LAG
      L1=LAG +1
      DO 15  I=1,L1
      AI=I-1
      FREQ=AI/AL*.5*FREPAR
      IF(AI)90,90,91
   90 PERIOD=0.
      GO TO 92
   91 PERIOD=1./FREQ
   92 CALL BLANKV(PLOTLN(1),100)
      JK=(P(I,1)-XMIN)/SCALFAC+.5001
      IF(JK.LE.100.AND.JK.GE.1) PLOTLN(JK)=1HX
      IF(JK.GT.100)PLOTLN(100)=1H+
      IF(JK.LT.1)PLOTLN(1)=1H-
      JK=(P(I,2)-XMIN)/SCALFAC+.5001
      IF(JK.LE.100.AND.JK.GE.1) PLOTLN(JK)=1HY
      IF(JK.GT.100)PLOTLN(100)=1H+
      IF(JK.LT.1)PLOTLN(1)=1H-
      CA=CO(I,IKK)
      JK=(CA    -XMIN)/SCALFAC+.5001
      IF(JK.LE.100.AND.JK.GE.1) PLOTLN(JK)=1HC
      IF(JK.GT.100)PLOTLN(100)=1H+
      IF(JK.LT.1) PLOTLN(1) = 1H-
      WRITE(61,1000) PERIOD,PLOTLN
```

```
      1000  FORMAT(2X,F7.3,5X,1H*,100A1,1H*)
      15    CONTINUE
            GO TO(20,39),IKK
       20   CALL MOVE(P(1,3),P(1,1),101)
            CALL MOVE(P(1,4),P(1,2),101)
            CALL MOVE(C(1,2),C(1,1),101)
            CALL MOVE(Q(1,2),Q(1,1),101)
            CALL POLAR(C(1,2),Q(1,2),P(1,3),P(1,4),LAG,CO(1,2),PH(1,2))
            WRITE(6,605)
            WRITE(6,606)
            GO TO 800
       39   CONTINUE
            DO 2482 I=1,LI
            TA=SUMANG(I)
            IF(TA.GT.180.) SUMANG(I)=TA-360.
      2482  IF(TA.LT.-180.)SUMANG(I)=TA+360.
            REWIND 2
            BUFFER IN(2,1)(XM1,Q(101,6))
            IF(IFUNIT(2))1,33,1
       33   REWIND 2
            BUFFER OUT(2,1)(XM1,NPTS)
            BUFFER OUT(2,1)(P(101,1),Q(101,6))
            IF(IFUNIT(2))1,34,1
        34  CONTINUE
            BUFFER OUT(2,1)  (PAM(1,1),SUMANG(101))
            J=IFUNIT(2)
            BUFFER OUT(2,1)(FRQ(1)                ,PH(101,6))
            IF(IFUNIT(2))1,32,1
       32   ENDFILE 2
            REWIND 2
       1    STOP
      601   FORMAT(1H1,1X,134HLAG    FREQ  LNFREQ PERIOD  POWER(U)   LNPOWU  PO
           $WER(V)  LNPOWV COSPECTRUM QUADSPEC  COHERNC  PHASE     MAJ        M
           $IN          ROT.COEF. T     )
      602   FORMAT(1H ,1X,F4.0,1X,3(F6.2,1X),E 10.2,1X,F6.2,1X,E10.2,1X,F6.2,1
           $X,2(E9.2,1X),1X,F7.3,1X,F7.2,3(E9.2),1X,I3)
      603   FORMAT(1H1 ///////////,25X,3H U  )
      605   FORMAT(1H1 ///////////,25X,3H UP )
      604   FORMAT(1H+,30X,3H V    )
      606   FORMAT(1H+,30X,3H VP   )
      607   FORMAT(1H1 /////////// 25X,10H U       UP                  )
      608   FORMAT(1H1 /////////// 25X,10H V       VP                  )
            END
```

USASI FORTRAN DIAGNOSTIC RESULTS FOR OUTPUT

NO ERRORS

```
      SUBROUTINE  POLAR(C,Q,X,Y,L,COHERE,PHASE)
      DIMENSION C(2),Q(2),X(2),Y(2),COHERE(2),PHASE(2)
      LP1=L+1
      PI=3.1415926536
      EPS=1.E-15
      XYMAX=0.
      DO 20 K=1,LP1
      XY=X(K)*Y(K)
      IF(XY.GT.XYMAX) XYMAX=XY
   20 CONTINUE
      DO 40 K=1,LP1
      XY=X(K)*Y(K)
      IF(XY.LE.XYMAX*EPS) GO TO 30
      COHERK=SQRT((C(K)*C(K)+Q(K)*Q(K))/XY)
      IF((Q(K)**2+C(K)**2).EQ.0.)GO TO 41
      PHASE(K)=ATAN2(Q(K),C(K))*180./PI
      GO TO 42
   41 PHASE(K)=0.
   42 CONTINUE
      COHERE(K)=COHERK
      GO TO 40
   30 COHERE(K)=0.
      PHASE(K)=0.
   40 CONTINUE
      RETURN
      END
```

USASI FORTRAN DIAGNOSTIC RESULTS FOR POLAR

NO ERRORS

USASI FORTRAN(1.1)/MSOS 4.2   INTEGER WORD SIZE = 1   PSR 287   DATE   11/1

```
          FUNCTION KANGE(N)
    5     IF(N.LT.360)GO TO 11
          GO TO 2
   11     IF(N.GE.0)GO TO 1
          GO TO 3
    1     KANGE=N
          RETURN
    2     N=N-360
          GO TO 5
    3     N=N+360
          GO TO 5
          END
```

USASI FORTRAN DIAGNOSTIC RESULTS FOR KANGE

NO ERRORS

USASI FORTRAN(1.1)/MSOS 4.2   INTEGER WORD SIZE = 1   PSR 287   DATE 11/1

```
      FUNCTION KFIX(X)
      CONV=ATAN(1.)/45.
      X=X/CONV
      KFIX=IFIX(SGN(X)*(ABS(X)+.5))
      RETURN
      END
```

USASI FORTRAN DIAGNOSTIC RESULTS FOR KFIX

NO ERRORS

USASI FORTRAN(1.1)/MSOS 4.2 INTEGER WORD SIZE = 1 PSR 287 DATE 11/1

```
      FUNCTION SGN(X)
      IF(X.EQ.0.)GO TO 2
      SGN=X/ABS(X)
    1 RETURN
    2 SGN=0.
      GO TO 1
      END
```

USASI FORTRAN DIAGNOSTIC RESULTS FOR SGN

NO ERRORS

USASI FORTRAN(1.1)/MSOS 4.2 INTEGER WORD SIZE = 1 PSR 287 DATE 11/

```
      SUBROUTINE NFACTS(PU,PV,PUV,QUV,MAJ,MIN,AP,AM,THETA,RC,PP,PM)
      REAL MAJ,MIN
      RC=-2.*QUV/(PU+PV)
      PP=.25 *  (2.*PU+2.*PV+4.*QUV)
      PM=.25*   (2.*PU+2.*PV-4.*QUV)
      AP=SQRT(PP)
      AM=SQRT(PM)
      THETA=.5*ATAN2(2.*PUV,PU-PV)
      MAJ=ABS(AP+AM)
      MIN=ABS(AP-AM)
      RETURN
      END
```

USASI FORTRAN DIAGNOSTIC RESULTS FOR NFACTS

NO ERRORS

E 000.98 MIN
SE,47

DATE

**NAME** Specplt

**PURPOSE**

Determine correlation and phase between two vector series.

**DESCRIPTION**

Reads Co and quadrature spectra for six pairs. Positive and negative rotating spectra for each vector, rotary coefficients, difference and sum of angles of orientation, calculates in COSPEC positive and negative coherence and phase angles.

**USAGE**

Plots on the line printer are made for positive and negative coherence and associated phase angles. A final plot is made the phase angles between vectors based on phase angle between positive correlations negative correlations, the averaged phase angle and its complement

| INPUT | OUTPUT |
|---|---|
| See above | Printed output of correlation parameters by frequency band. |
| **SUBROUTINES USED** | **CALLED BY** |
| COspect | Datal |
| **STORAGE REQUIREMENTS** | **SOURCE LANGUAGE** |
| **MACHINE** | **PROGRAMMER** |

**REFERENCES**

```
      PROGRAM SPECPLT
      COMMON XM1,XM2,LAG    ,FREFAR
      COMMON NAUG,NPTS
      COMMON          P(101,4),C(101,6),Q(101,6),PMM(101,2),PPP(101,2)
      COMMON DANING(101),ROT(101,2) ,SUMANG(101)
      COMMON FRQ(101)
      INTEGER PLOTLN(102)
      REWIND 2
      BUFFER IN(2,1) (XM1,NPTS)
      BUFFER IN(2,1) (P(101,1),Q(101,6))
      BUFFER IN(2,1) (PMM(1,1),SUMANG(101))
      IF(IFUNIT(2))1,2,1
    1 WRITE(6,600)
  600 FORMAT(1H1,6HERROR    )
    3 STOP
    2 CONTINUE
      WRITE(6,701)XM1,XM2,LAG
  701 FORMAT(1H1/////,17H MOORING NUMBERS    ,A8,6H  AND  ,A8,///
     $5H LAG=,I4)
      L1=LAG+1
      CALL ZEROV(FRQ,101)
      DO 5 I=1,L1
      IF(I.EQ.1)GO TO 123
      A=FLOAT(I-1)/FLOAT(LAG)*.5*FREFAR
      FRQ(I)=1./A
  123 CONTINUE
      CALL CUSPEC(C(I,3),C(I,6),Q(I,5),Q(I,4),C(I,4),C(I,5),Q(I,6),
     $Q(I,3),TPP,TMM,DT,PPP(I,1),PMM(I,1),PPP(I,2),PMM(I,2),DANING(I),
     1P(I,1),P(I,2),P(I,3),P(I,4),ROT(I,1),ROT(I,2),SUMANG(I))
      K=I-1
      WRITE(6,700)K,FRQ(I)
  700 FORMAT(1H+,I3,2X,F6.2)
    5 CONTINUE
 5501 FORMAT(1H1)
      WRITE(6,5501)
      WRITE(6,118)
  118 FORMAT(1X ,4HBAND,1X,6HPERIOD,2X,3H0.0,21X,3H0.5,21X,3H1.0,1X,4H-1
     180,20X,3H0.0,20X,3H180)
      DO 15 I=1,L1
      CALL BLANKV(PLOTLN(1),102)
      JK=P(I,1)*50.+.5001
      IF(JK.LE.50.AND.JK.GE.1)PLOTLN(JK)=1H+
      JK=P(I,3)*50.+.5001
      IF(JK.LE.50.AND.JK.GE.1)PLOTLN(JK)=1H-
      PLOTLN(51)=1H*
      JK= P(I,2)*25./180.+77.5001
      PLOTLN(77)=1H.
      IF(JK.LE.102.AND.JK.GE.51)PLOTLN(JK)=1H+
      JK=P(I,4)*25./180.+77.5
      IF(JK.LE.102.AND.JK.GE.51)PLOTLN(JK)=1H-
      WRITE(6,1000)I,FRQ(I),PLOTLN
 1000 FORMAT(1X,I4,2X,F7.1,1X,1H*,102A1,1H*)
   15 CONTINUE
      WRITE(6,118)
```

```
      WRITE(6,550)
      WRITE(6,119)
  119 FORMAT(1X,4HBAND,1X,5HPER100,2X,4H-180,18X,4H-90.,23X,3H0.0,21X,3H90.,21X
     190.,21X,4H180.)
      DO 16 I=1,LL
      CALL BLANKV(PLOTLN(1),102)
      JK=PPP(I,1)*50./180.+50.501
      PLOTLN(76)=1H.
      PLOTLN(26)=1H.
      PLOTLN(51)=1H.
      IF(JK.LE.101.AND.JK.GE.1)PLOTLN(JK)=1H+
      JK=PPM(I,1)*50./180.+50.501
      IF(JK.LE.101.AND.JK.GE.1)PLOTLN(JK)=1H-
      JK=PPP(I,2)*50./180.+50.501
      IF(JK.LE.101.AND.JK.GE.1)PLOTLN(JK)=1HC
      JK=PPM(I,2)*50./180.+50.501
      IF(JK.LE.101.AND.JK.GE.1)PLOTLN(JK)=1HA
      WRITE(6,1000)I,PER(I),PLOTLN
   16 CONTINUE
      WRITE(6,119)
      REWIND 2
      GO TO 3
      END
```

FORTRAN DIAGNOSTIC RESULTS FOR SPECPLT

0 ERRORS

DATE

**NAME** Cospec

**PURPOSE**

Selects a strategy and computes correlations between vectors based on rotary coefficients.

**DESCRIPTION**

In input argument Pvv is cospectrum of the 2 X components
Qvv quadrature of the 2 X components
and so on.
R1, R2 are the rotary coefficient, SA sum of angles of orientation, D1 difference between angles.

**USAGE**

P3, P5 negative rotary spectra
P4, P6 positive rotary spectra
C2 positive coherency, TP positive phase angle
C1 negative coherency TM negative phase angle
P6 phase angle between vectors based on TP

| INPUT | OUTPUT |
|---|---|
| P5 phase angle between vectors based on TM, Ang 2, complementary averaged angle, Ang averaged angle | |

| SUBROUTINES USED | CALLED BY |
|---|---|
| | Specplt |

| STORAGE REQUIREMENTS | SOURCE LANGUAGE |
|---|---|
| | |

| MACHINE | PROGRAMMER |
|---|---|
| | |

**REFERENCES**

```
      SUBROUTINE COSPEC(PUU,PVV,QU1V,QUV1,PUV1,PU1V,QVV,QUU,TPP,TMM,DT,P
     16,P5,P4,P3,D1,C2,TP,C1,TM,R1,R2,SA)
      S1=1
      C=180./3.14159265
      IF(R1/R2.LT.0.)GO TO 27
      IF(R1.GT.0..AND.R2.GT.0.)S1=-1.
      PXPXP=.5*(PUU+PVV-QU1V+QUV1)
      QXPXP=.5*(PUV1-PU1V-QVV-QUU)
      QXMXM=0.5*(PUV1-PU1V+QVV+QUU)
      PXMXM=.5*(PUU+PVV-QUV1+QU1V)
      TPP=-ATAN2(QXPXP,PXPXP)
      TMM= ATAN2(-QXMXM,PXMXM)*(-1.)
      Q2=QXPXP
      Q1=QXMXM
      P1=PXMXM
      P2=PXPXP
C
      C1=(P1*P1+Q1*Q1)/(P3*P5)
      C2=(P2*P2+Q2*Q2)/(P4*P6)
C
      ANG=-.5*ATAN2(Q2*P1-Q1*P2,P1*P2+Q2*Q1)*C
      TM=C*TMM
      TP=C*TPP
   25 TL=(TP-D1)
      TO=(D1+TM)
      GO TO 15
   27 PXPXP=.5*(PUU-PVV-QUV1-QU1V)
      QXPXP=.5*(PUV1+PU1V+QUU-QVV)
      QXMXM=.5*(PUV1+PU1V-QUU+QVV)
      PXMXM=.5*(PUU-PVV+QUV1+QU1V)
      C2=(PXPXP**2+QXPXP**2)/(P6*P3)
      C1=(PXMXM**2+QXMXM**2)/(P4*P5)
      IF(R1.GT.0..AND.R2.GT.0.)S1=-1
      ANG=.5*ATAN2(QXPXP*PXMXM-QXMXM*PXPXP,PXPXP*PXMXM+QXPXP*QXMXM)*C
      TMM=ATAN2(QXMXM,PXMXM)  *(-1.)
      TPP= ATAN2(QXPXP,PXPXP)
      TM=C*TMM
      TP=C*TPP
      TL=TP-SA
      TO=SA+TM
   15 CONTINUE
      IF(ANG.GT.0.)ANG2=ANG-180.
      IF(ANG.LT.0.) ANG2=ANG+180.
      P3=ANG
      P4=ANG2
      P6=TL
      P5=TO
      IF(TL.GT.180.)P6=TL-360.
      IF(TL.LT.-180.)P4=TL+360.
      IF(TO.GT.180.)P5=TO-360.
      IF(TO.LT.-180.)P5=TO+360.
      WRITE(6,600)C2,TP,C1,TM,P6,P5,ANG2,ANG
  600 FORMAT(1H ,12X
     2         ,4HC++=,F5.2,2X,8HTHETA++=,F6.1,2X,4HC--=,F5.2,2X,8HTHET
```

```
5        1A--=,F6.1,2X,5HANG+=,F6.1,5HANG-=,F6.1,2X,10HCOMPL.ANG=,F6.1,1X,
6        2+HANG=,F6.1)
7         RETURN
8         END
```

US SI FORTRAN DIAGNOSTIC RESULTS FOR COSPEC

NO ERRORS

ERENCED STATEMENT LABELS

00.71 MIN

DATE

NAME DATA1

PURPOSE

Reads vector time series and calls total assembled programmes

DESCRIPTION

Reads data, X, as two, two component vectors, checks for missing data, interpolates and writes data onto tape via Speak. Subsequently Calcate, Output and Specplt are called and the program is completed.

USAGE

| INPUT | OUTPUT |
| --- | --- |
| X raw data form two vector time series. | Interpolated data series |

| SUBROUTINES USED | CALLED BY |
| --- | --- |
| Speak | |

| | |
| --- | --- |
| STORAGE REQUIREMENTS | SOURCE LANGUAGE |
| MACHINE | PROGRAMMER |

REFERENCES

SI FORTRAN(1.1)/MSOS 4.2  INTEGER WORD SIZE = 1     PSR 287     DATE     10/29/73

```
          PROGRAM DATA1
    C
          REAL MOOR
          COMMON XM1,XM2,LAG ,FREPAR
          COMMON NAUG,NPTS
          COMMON X(1026,2,2)
          CALL ZEROV(X,4104)
          CONV=ATAN(1.)/45.
          READ(5,81) NPTS,LAG,FREPAR
     81   FORMAT(2(5X,I5),F10.0)
          DO 3 M=1,2
          I=-5
      1   I=I+6
          I3=I+5
          READ(5,500)MOOR,K,(X(J,1,M),X(J,2,M),J=I,I3)
          DO 115 J=I,I3
          SP=X(J,1,M)
          DIR=CONV*X(J,2,M)
          X(J,1,M)=SP*SIN(DIR)
    115   X(J,2,M)=SP*COS(DIR)
          IF(I.GT.512) GO TO 4
          IF(K.EQ.0)GO TO 2
          GO TO 1
      4   WRITE(6,601)
      2   I=I-5
          WRITE(6,600)I,MOOR
          XM2=MOOR
          IF(M.EQ.2)GO TO 3
          XM1=MOOR
          NUM=I
      3   CONTINUE
          IF(NUM.GT.I)NUM=I
    500   FORMAT(5X,F1.0,14X,I1,1X,6(3X,F3.1,F3.0))
    600   FORMAT(1H-,/////5X,I5,3X,24HDATA POINTS AT MOORING      ,A8)
    601   FORMAT(1H-,25HTOO MANY DATA POINTS                        )
    C     FILTER FOR UNDEFINED VALUES BY LINEAR INTERPOLATION
          KAPPA=0
          DO 9000 M=1,2
          DO 9000 N=1,2
          DO 9000J=1,NUM
          IF(X(J,N,M).LE.9998.5)GO TO 9000
          KAPPA=KAPPA+1
          IF((J.LT.NUM).AND.(X(J+1,N,M).LE.9998.5))GO TO 9001
          X(J,N,M)=X(J-1,N,M)
          GO TO 9000
   9001   X(J,N,M)=  (X(J-1,N,M)+X(J+1,N,M))/2.
   9000   CONTINUE
          WRITE(6,6666)KAPPA
   6666   FORMAT(1H //I4,30H  DATA POINTS INTERPOLATED              ,/////)
          NPTS=NUM
          CALL SPEAK(NUM)
          WRITE(6,603)
    603   FORMAT(1H1,8HFINISHED  )
          WRITE(6,509) NPTS,LAG,FREPAR
```

```
SI FORTRAN(1.1)/MSOS 4.2  INTEGER WORD SIZE = 1     PSR 287      DATE    10/29,

      509 FORMAT(1X,2I5,F10.3)
          STOP
          END

USASI FORTRAN DIAGNOSTIC RESULTS FOR DATA1

     NO ERRORS
```

ASI FORTRAN(1.1)/MSOS 4.2  INTEGER WORD SIZE = 1     PSR 287     DATE     10/29/73

```
          SUBROUTINE SPEAK(NUM)
          COMMON XM1,XM2,LAG    ,FREPAR
          COMMON NAUG,NPTS
          COMMON X(1026,4),P(101,4)
          COMMON C(101,6),Q(101,6),CO(101,2),PH(101)
          NPTS=NUM
          NAUG=0
          CALL ZEROV(P,1919)
          REWIND 2
          BUFFER OUT(2,1)(XM1,PH(101))
          IF(IFUNIT(2))1,2,1
    2     CONTINUE
          ENDFILE 2
          REWIND 2
          BUFFER IN(2,1)(XM1,PH(101))
          KANG=IFUNIT(2)
    1     REWIND 2
          RETURN
          END
```

USASI FORTRAN DIAGNOSTIC RESULTS FOR SPEAK

NO ERRORS

00.51 MIN

DATA POINTS AT MOORING  00000000

DATA POINTS AT MOORING  00000000

POINTS INTERPOLATED

# APPENDIX II

## Programme Listing for scalar-vector correlations

| DATE |
|---|



**NAME** Salcate

**PURPOSE**

Reads a scalar and a vector time series, removes trends, calculates Fourier coefficients, power spectra, and crosspectra. Salcate is main driver routine for the spectral computations.

**DESCRIPTION**

**USAGE**

Note that the other routines, sutput ssecplt are essentially outlined by the previous programmes in Data 1

| INPUT | OUTPUT |
|---|---|
| X vector contains a scalar and vector time series | |

| SUBROUTINES USED | CALLED BY |
|---|---|
| | S Data 1 |
| STORAGE REQUIREMENTS | SOURCE LANGUAGE |
| MACHINE | PROGRAMMER |

**REFERENCES**

RTRAN(1.1)/MSOS 4.2 INTEGER WORD SIZE = 1 PSR 2a7 DATE 11/20/73

```
      PROGRAM SALCATE
C     SCALAR - VECTOR TIME SERIES CORRELATIONS
C
      COMMON XM1,XM2,LAG   ,FREPAR
      COMMON NAUG,NPTS
      COMMON X(1026,3),P(101,3)
      COMMON C(101,3),Q(101,3),CO(101),PH(101)
      REWIND 2
      BUFFER IN(2,1)(XM1,PH(101))
      IF(IFUNIT(2))1,2,1
 2    CONTINUE
      WRITE(6,753)NPTS
 753  FORMAT(1H1,I5)
      WRITE(6,701)XM1,XM2,LAG
 701  FORMAT(1H1/////,17H MOORING NUMBERS    ,A8,6H  AND  ,A8,///
     $5H LAG=,I4)
      CALL SPECTR(NPTS)
 1    REWIND 2
      STOP
      END
```

FORTRAN DIAGNOSTIC RESULTS FOR SALCATE

NO ERRORS

```
)RTRAN(1.1)/MSOS 4.2  INTEGER WORD SIZE = 1   PSR 2n7      DATE   11/20/73

      SUBROUTINE SPECTR(NUM)
      COMMON XM1,XM2,LAG   ,FREPAR
      COMMON NAUG,NPTS
      COMMON X(1026,3),P(101,3)
      COMMON C(101,3),Q(101,3),CO(101),PH(101)
      WRITE(6,6699)XM1,XM2
 6699 FORMAT(1H1,///5X,A8,5X,A8/////)
      NCH=3
      ISKIP=0
  800 CONTINUE
      CALL FSCRAN(X,NUM,LAG,P,CO,PH,ISKIP,NCH,C,Q)
      REWIND 2
      BUFFER OUT(2,1)(XM1,PH(101))
      IF(IFUNIT(2))901,902,901
  902 ENDFILE 2
      REWIND 2
  901 RETURN
      END

 FORTRAN DIAGNOSTIC RESULTS FOR SPECTR

 0 ERRORS

CED STATEMENT LABELS
```

```
      SUBROUTINE FSCRAN(X,NPTS,NB,P,COH,PH,ISKIP,NCH,C,Q)
      COMMON XXX,XXY,LIL    ,FREPAR
      COMMON NAUG,KIT
      DIMENSION X(1026,3),P(101,3),C(101,3),Q(101,3),COH(101)
      DIMENSION PH(101)
      DIMENSION S(513),AVG(3),VAR(3),SUM(3)
      KR=5
      LP=6
      KIT=NPTS
      NBP1=NB+1
      IF(ISKIP.EQ.1)GO TO 450
      IND=0
      DO 135 ICH=1,NCH
      CALL AVVAR(X(1,ICH),NPTS,AVG(ICH),VAR(ICH))
      IF(ICH.LT.3)GO TO 135
      WRITE(LP,120)
 120  FORMAT( /,9X,13H    MEAN         , 3X,4HVAR ,19X,4HMEAN, 8X,4HVAR ,
     *2(19X,4HMEAN,8X,4HVAR   )/)
      WRITE(LP,130)(AVG(I),VAR(I),I=1,4)
 130  FORMAT(2X,7HX-COMP,2E11.3,5X   ,7HY-COMP,2E11.3,5X,8HXP-COMP,
     *2E11.3,5X,8HYP-COMP,2E11.3)
 135  CONTINUE
      DO 145 ICH=1,NCH
      DO 140 I=1,NPTS
      X(I,ICH)=X(I,ICH)-AVG(ICH)
 140  CONTINUE
      AVG(ICH)=0.
 145  CONTINUE
      FAC=.95
      DEL=0
      DAL=0
      DIL=0.
      DO 146 I=2,NPTS
      DEL2=FAC*DEL+.05*X(I,1)
      DEL3=FAC*DAL+.05*X(I,2)
      DEL4=FAC*DIL+.05*X(I,3)
      X(I-1,1)=X(I-1,1)-DEL
      X(I-1,2)=X(I-1,2)-DAL
      X(I-1,3)=X(I-1,3)-DIL
      DAL=DEL3
      DIL=DEL4
 146  DEL=DEL2
      WRITE(6,25)((X(I,J),J=1,NCH),I=1,100)
 25   FORMAT(3(1X,4E10.3))
C     WRITE(LP,150)
150   FORMAT(// 30H MEAN REMOVED FROM EACH SERIES  )
160   CONTINUE
165   DO 170 ICH=1,NCH
170   CALL TAPER(X(1,ICH),NPTS)
C     WRITE(LP,180)
180   FORMAT(// 44H COSINE TAPER APPLIED TO ENDS OF EACH SERIES )
190   CONTINUE
      DO 200 ICH=1,NCH
200   CALL ZEFILL(X(1,ICH),NPTS,M,NAUG)
```

```
      IF(NAUG.EQ.NPTS) GO TO 220
      WRITE(LP,210)NPTS,NAUG
210   FORMAT( / 36H EACH SERIES AUGMENTED BY ZEROS FROM I5,8H PTS TO
     2         I5,4H PTS )
220   CONTINUE
      CALL RLFORT(X(1,1),M+1,S,0,IFERR)
      DO 270 ICH=1,NCH
      CALL RLFORT(X(1,ICH),M+1,S,-3,IFERR)
      IF(IFERR.EQ.0) GO TO 270
      WRITE(LP,260) IFERR,ICH
260   FORMAT(22H FORT ERROR IFERR,ICH=    2I4)
 115  RETURN
270   CONTINUE
      NEB=NAUG/2
      NEBP1=NEB+1
      DO 290 ICH=1,NCH
      CALL PSPEC(X(1,ICH),NEB,P(1,ICH),NB)
      DO 290 I=1,NBP1
      P(I,ICH)=P(I,ICH)*FLOAT(NAUG)/FLOAT(NPTS)
290   CONTINUE
 360  FORMAT(/ 4H SUM,13X,4E12.3)
      CALL XSPEC(X(1,1),X(1,2),NEB,C(1,1),Q(1,1),NB)
      CALL XSPEC(X(1,1),X(1,3),NEB,C(1,3),Q(1,3),NB)
      CALL XSPEC(X(1,2),X(1,3),NEB,C(1,2),Q(1,2),NB)
      ZTL=FLOAT(NAUG)/FLOAT(NPTS)
      DO 2825 I=1,NBP1
      DO 2825 J=1,3
      C(I,J)=C(I,J)*ZTL
      Q(I,J)=Q(I,J)*ZTL
 2825 CONTINUE
      SUMC1=0.
      SUMC2=0.
      SUMC3=0.
      SUMQ1=0.
      SUMQ2=0.
      SUMQ3=0.
      DO 460 I=1,NBP1
      SUMC1=SUMC1+C(I,1)
      SUMC2=SUMC2+C(I,2)
      SUMC3=SUMC3+C(I,3)
      SUMQ1=SUMQ1+Q(I,1)
      SUMQ2=SUMQ2+Q(I,2)
      SUMQ3=SUMQ3+Q(I,3)
 460  CONTINUE
      SUMC=SUMC1
      SUMQ=SUMQ1
      WRITE(6,560)SUM(1),SUM(2),SUM(3),SUMC,SUMQ
      SUMC=SUMC2
      SUMQ=SUMQ2
      WRITE(6,560)SUM(1),SUM(2),SUM(3),SUMC,SUMQ
      SUMC=SUMC3
      SUMQ=SUMQ3
      WRITE(6,560)SUM(1),SUM(2),SUM(3),SUMC,SUMQ
 560  FORMAT(/  4H SUM.13X,6E12.3)
```

ORTRAN(1.1)/MSOS 4.2  INTEGER WORD SIZE = 1     PSR 247     DATE  11/20/73

```
    7 FAC2=1./(1.-.05/1.95)**2
  450 CONTINUE
      DO 203 I=1,NBP1
      AI=I-1
      AL=NB
                     P(I,1)=P(I,1)*FAC2
      P(I,2)=P(I,2)*FAC2
      P(I,3)=P(I,3)*FAC2
      DO 942 JA=1,3
      C(I,JA)=C(I,JA)*FAC2
      Q(I,JA)=Q(I,JA)*FAC2
  942 CONTINUE
      IF(I.GT. NB/ 6) GO TO 203
      B=SIN(3.1416*(AI  )/AL)
      A=COS(3.1416*(AI  )/AL)
      IF(I.EQ.1) B=SIN(3.1416/(4.*AL))
      IF(I.EQ.1) A=COS(3.1416/(4.*AL))
      Z1=(.95*(1.-A)*(1.-.95*A)+(.95*B)**2)**2+(.95*B*.05)**2
      Z2=(1.-.95*A)**2+(.95*B)**2
      P(I,1)=P(I,1)*Z2*Z2/Z1 /FAC2
      P(I,2)=P(I,2)*Z2*Z2/Z1 /FAC2
      P(I,3)=P(I,3)*Z2*Z2/Z1/FAC2
      ZP5=Z2*Z2/Z1/FAC2
      DO 203 JA=1,3
      C(I,JA)=C(I,JA)*ZP5
      Q(I,JA)=Q(I,JA)*ZP5
  203 CONTINUE
      DO 280 ICH=1,NCH
      SUM(ICH)=0.
      DO 280 I=1,NBP1
  280 SUM(ICH)=SUM(ICH)+P(I,ICH)
      WRITE(LP,560) (SUM(ICH),ICH=1,NCH)
      GO TO 115
      END
```

1 FORTRAN DIAGNOSTIC RESULTS FOR FSCRAN

NO ERRORS

CED STATEMENT LABELS

160    00165    00180    00190

847

ORTRAN(1.1)/MSOS 4.2 INTEGER WORD SIZE = 1 PSR 2.7 DATE 11/20/73

```
      SUBROUTINE AVVAR(X,N,AV,VAR)
      DIMENSION X(2)
      AV=0.
      DO 10 I=1,N
10    AV=AV+X(I)
      AV=AV/FLOAT(N)
      VAR=0.
      DO 20 I=1,N
      XR=X(I)-AV
20    VAR=VAR+XR*XR
      VAR=VAR/FLOAT(N)
      RETURN
      END
```

I FORTRAN DIAGNOSTIC RESULTS FOR AVVAR

NO ERRORS

ORTRAN(1.1)/MSOS 4.2 INTEGER WORD SIZE = 1 PSR 297 DATE 11/20/73

```
      SUBROUTINE TAPER(X,N)
      DIMENSION X(2)
      PI=3.1415926
      DO 50 I=1,N
      ANG=(FLOAT(I)-.5)*PI*10./FLOAT(N)
      IF(ANG.GE.PI) GO TO 60
      FAC=.5-.5*COS(ANG)
      X(I)=X(I)*FAC
      IUP=N-I+1
50    X(IUP)=X(IUP)*FAC
60    RETURN
      END
```

1 FORTRAN DIAGNOSTIC RESULTS FOR TAPER

NO ERRORS

ORTRAN(1.1)/MSOS 4.2 INTEGER WORD SIZE = 1 PS 207 DATE 11/20/73

```
      SUBROUTINE ZEFILL(X,NRCD,M,NAUG)
      DIMENSION X(2)
      DO 50 J=1,13
      M=J
      N=2**M
      NAUG=2*N
      IF(NAUG-NRCD)50,60,70
50    CONTINUE
      STOP
60    RETURN
70    ISTART=NRCD+1
      DO 80 I=ISTART,NAUG
80    X(I)=0.0
      RETURN
      END
```

I FORTRAN DIAGNOSTIC RESULTS FOR ZEFILL

NO ERRORS

```
      SUBROUTINE RLFORT (C,M,S,IFS,IFERR)
      DIMENSION C(1),S(1)
      N = 2**M
      IF (IABS(IFS).LT.2) CALL FORT (C,M,S,0,IFERR)
      MCPLX = M - 1
      IF (IFS) 10,45,30
   10 CALL FORT (C,MCPLX,S,-2,IFERR)
      CFACT = 1.
      C(N+1) = 0.5*(C(1) - C(2))
      C(N+2) = 0.
      C(1) = 0.5*(C(1) + C(2))
      C(2) = 0.
      GO TO 50
   30 CFACT = -1.
      C(2) = C(1) - C(N+1)
      C(1) = C(1) + C(N+1)
      GO TO 50
   40 CALL FORT (C,MCPLX,S,2,IFERR)
   45 RETURN
   50 K = N/2 - 1
      MSIN = N/4
      C(K+3) = -C(K+3)
      DO 60 I=3,K,2
      IS = (I-1)/2
      IC = MSIN - IS
      ST = S(IS)
      CT = S(IC)*CFACT
      A1 = C(I)
      B1 = C(I+1)
      L = N - I
      A2 = C(L+2)
      B2 = C(L+3)
      C(I) = 0.5*(A1+A2+(B1+B2)*CT-(A1-A2)*ST)
      C(I+1) = 0.5*(B1-B2-(B1+B2)*ST-(A1-A2)*CT)
      C(L+2) = 0.5*(A1+A2-(B1+B2)*CT+(A1-A2)*ST)
   60 C(L+3) = 0.5*(B2-B1-(B1+B2)*ST-(A1-A2)*CT)
      IF(IFS+3)90,70,90
70    C(1)=C(1)*2.
      C(N+1)=C(N+1)*2.
      DO 80 I=4,N,2
80    C(I)=-C(I)
90    CONTINUE
      IF (IFS) 45,45,40
      END
```

FORTRAN DIAGNOSTIC RESULTS FOR RLFORT

NO ERRORS

FORTRAN (1.1)/MSOS 4.2  INTEGER WORD SIZE = 1     PSR 287     DATE   11/20/73

```
      SUBROUTINE FORT(A,M,S,IFS,IFERR)                               FORT 004
      DIMENSION A(1), S(1), K(14)                                    FORT 051
      EQUIVALENCE (K(13),K1),(K(12),K2),(K(11),K3),(K(10),K4)        FORT 052
      EQUIVALENCE (K( 9),K5),(K( 8),K6),(K(7),K7),(K( 6),K8)         FORT 053
      EQUIVALENCE (K( 5),K9 ),(K( 4),K10),(K( 3),K11),(K( 2),K12)    FORT 054
      EQUIVALENCE (K( 1),K13),( K(1),N2)                             FORT 055
      IF(M)2,2,3                                                     FORT 056
    3 IF(M-13) 5,5,2                                                 FORT 057
    2 IFERR=1                                                        FORT 058
    1 RETURN                                                         FORT 059
    5 IFERR=0                                                        FORT 060
      N=2**M                                                         FORT 061
      IF( IABS(IFS) - 1 ) 200,200,10                                 FORT 062
   10 IF( N-NP )20,20,12                                             FORT 065
   12 IFERR=1                                                        FORT 066
      GO TO 200                                                      FORT 067
   20 K(1)=2*N                                                       FORT 069
      DO 22 L=2,M                                                    FORT 070
   22 K(L)=K(L-1)/2                                                  FORT 071
      DO 24 L=M,12                                                   FORT 072
   24 K(L+1)=2                                                       FORT 073
      IJ=2                                                           FORT 076
      DO 30 J1=2,K1,2                                                FORT 077
      DO 30 J2=J1,K2,K1                                              FORT 078
      DO 30 J3=J2,K3,K2                                              FORT 079
      DO 30 J4=J3,K4,K3                                              FORT 080
      DO 30 J5=J4,K5,K4                                              FORT 081
      DO 30 J6=J5,K6,K5                                              FORT 082
      DO 30 J7=J6,K7,K6                                              FORT 083
      DO 30 J8=J7,K8,K7                                              FORT 084
      DO 30 J9=J8,K9,K8                                              FORT 085
      DO 30 J10=J9,K10,K9                                            FORT 086
      DO 30 J11=J10,K11,K10                                          FORT 087
      DO 30 J12=J11,K12,K11                                          FORT 088
      DO 30 JI=J12,K13,K12                                           FORT 089
      IF(IJ-JI)28,30,30                                              FORT 090
   28 T=A(IJ-1 )                                                     FORT 091
      A(IJ-1)=A(JI-1)                                                FORT 092
      A(JI-1)=T                                                      FORT 093
      T=A(IJ)                                                        FORT 094
      A(IJ)=A(JI)                                                    FORT 095
      A(JI)=T                                                        FORT 096
   30 IJ=IJ+2                                                        FORT 097
      IF(IFS)32,2,36                                                 FORT 098
   32 FN = N                                                         FORT 100
      DO 34 I=1,N                                                    FORT 101
      A(2*I-1) = A(2*I-1)/FN                                         FORT 102
   34 A(2*I)=-A(2*I)/FN                                              FORT 103
   36 DO 40 I=1,N,2                                                  FORT 105
      T = A(2*I-1)                                                   FORT 106
      A(2*I-1) =T + A(2*I+1)                                         FORT 107
      A(2*I+1)=T-A(2*I+1)                                            FORT 108
      T=A(2*I)                                                       FORT 109
      A(2*I) = T + A(2*I+2)                                          FORT 110
```

```
   40 A(2*I+2)= T - A(2*I+2)                                    FORT 111
      IF(M-1) 2,1   ,50                                         FORT 112
C     SET FOR L=2                                               FORT 113
   50 LEXP1=2                                                   FORT 114
C     LEXP1=2**(L-1)                                            FORT 115
      LEXP=8                                                    FORT 116
C     LEXP=2**(L+1)                                             FORT 117
      NPL= 2**MT                                                FORT 118
C     NPL = NP* 2**-L                                           FORT 119
   60 DO 130 L=2,M                                              FORT 120
      DO 80 I=2,N2,LEXP                                         FORT 122
      I1=I + LEXP1                                              FORT 123
      I2=I1+ LEXP1                                              FORT 124
      I3 =I2+LEXP1                                              FORT 125
      T=A(I-1)                                                  FORT 126
      A(I-1) = T +A(I2-1)                                       FORT 127
      A(I2-1) = T-A(I2-1)                                       FORT 128
      T =A(I)                                                   FORT 129
      A(I) = T+A(I2)                                            FORT 130
      A(I2) = T-A(I2)                                           FORT 131
      T= -A(I3)                                                 FORT 132
      TI = A(I3-1)                                              FORT 133
      A(I3-1) = A(I1-1) - T                                     FORT 134
      A(I3    ) = A(I1 )     - TI                               FORT 135
      A(I1-1) = A(I1-1) +T                                      FORT 136
   80 A(I1)    = A(I1    )   +TI                                FORT 137
      IF(L-2) 120,120,90                                        FORT 138
   90 KLAST=N2-LEXP                                             FORT 139
      JJ=NPL                                                    FORT 140
      DO 110 J=4,LEXP1,2                                        FORT 141
      NPJJ=NT-JJ                                                FORT 142
      UR=S(NPJJ)                                                FORT 143
      UI=S(JJ)                                                  FORT 144
      ILAST=J+KLAST                                             FORT 145
      DO 100 I= J,ILAST,LEXP                                    FORT 146
      I1=I+LEXP1                                                FORT 147
      I2=I1+LEXP1                                               FORT 148
      I3=I2+LEXP1                                               FORT 149
      T=A(I2-1)*UR-A(I2)*UI                                     FORT 150
      TI=A(I2-1)*UI+A(I2)*UR                                    FORT 151
      A(I2-1)=A(I-1)-T                                          FORT 152
      A(I2   )=A(I    ) - TI                                    FORT 153
      A(I-1) =A(I-1)+T                                          FORT 154
      A(I)    =A(I)+TI                                          FORT 155
      T=-A(I3-1)*UI-A(I3)*UR                                    FORT 156
      TI=A(I3-1)*UR-A(I3)*UI                                    FORT 157
      A(I3-1)=A(I1-1)-                                          FORT 158
      A(I3)    =A(I1   )-TI                                     FORT 159
      A(I1-1)=A(I1-1)+T                                         FORT 160
  100 A(I1)    =A(I1)    +TI                                    FORT 161
  110 JJ=JJ+NPL                                                 FORT 163
  120 LEXP1=2*LEXP1                                             FORT 165
      LEXP = 2*LEXP                                             FORT 166
  130 NPL=NPL/2                                                 FORT 167
```

TRAN(1.1)/MSOS 4.2 INTEGER WORD SIZE = 1 PSR 287 DATE 11/20/73

```
  140 IF(IFS)145,2,1                                          FORT 16
  145 DO 150 I=1,N                                            FORT 171
  150 A(2*I) =-A(2*I)                                         FORT 172
  160 GO TO 1                                                 FORT 173
  200 NP=N                                                    FORT 176
      MP=M                                                    FORT 177
      NT=N/4                                                  FORT 178
      MT=M-2                                                  FORT 179
      IF(MT) 260,260,205                                      FORT 180
  205 THETA=.7853981634                                       FORT 181
C     THETA=PI/2**(L+1)     FOR L=1                           FORT 182
  210 JSTEP = NT                                              FORT 183
C     JSTEP = 2**( MT-L+1 ) FOR L=1                           FORT 184
      JDIF = NT/2                                             FORT 185
C     JDIF = 2**(MT-L)   FOR L=1                              FORT 186
      S(JDIF) = SIN(THETA)                                    FORT 187
      IF (MT-2)260,220,220                                    FORT 188
  220 DO 250 L=2,MT                                           FORT 189
      THETA = THETA/2.                                        FORT 190
      JSTEP2 = JSTEP                                          FORT 191
      JSTEP = JDIF                                            FORT 192
      JDIF = JDIF/2                                           FORT 193
      S(JDIF)=SIN(THETA)                                      FORT 194
      JC1=NT-JDIF                                             FORT 195
      S(JC1)=COS(THETA)                                       FORT 196
      JLAST=NT-JSTEP2                                         FORT 197
      IF(JLAST-JSTEP)250,230,230                              FORT 198
  230 DO 240 J=JSTEP,JLAST,JSTEP                              FORT 199
      JC=NT-J                                                 FORT 200
      JD=J+JDIF                                               FORT 201
  240 S(JD)=S(J)*S(JC1)+S(JDIF)*S(JC)                         FORT 202
  250 CONTINUE                                                FORT 203
  260 IF(IFS)20,1,20                                          FORT 204
      END                                                     FORT 205
```

FORTRAN DIAGNOSTIC RESULTS FOR FORT

NO ERRORS

FED STATEMENT LABELS

```
40     00160     00210
```

TRAN(1.1)/MSOS 4.2 INTEGER WORD SIZE = 1 PSR 287 DATE 11/20/73

```
      SUBROUTINE PSPEC(A,N,P,M)
      DIMENSION A(2),P(2)
      REAL L
      FN=N
      FM=M
      W=FN/FM
      MP1=M+1
      DO 100 INDL=1,MP1
      L=INDL-1
      XA=AMAX1(0.0,W*(L-0.5))
      XB=AMIN1(FN,W*(L+0.5))
      IXA=XA
      IXA=MAX0(0,IXA)
      IXAIND=IXA+1
      IXB=XB+1.0
      IXB=MIN0(IXB,N)
      IXBIND=IXB+1
      FINT=0.0
      DO 50 IXIND=IXAIND,IXBIND
      IX=IXIND-1
      R=A(2*IXIND-1)
      S=A(2*IXIND)
      AMP=R*R+S*S
      IF(IX.LE.IXA+1  .OR.  IX.GE.IXB-1) GO TO 40
      FINT=FINT+AMP
      GO TO 50
40    XAA=AMAX1(XA,FLOAT(IX)-0.5)
      XBB=AMIN1(XB,FLOAT(IX)+0.5)
      DELX=AMAX1(0.0,XBB-XAA)
      FINT=FINT+AMP*DELX
50    CONTINUE
      P(INDL)=FINT*0.5
100   CONTINUE
      RETURN
      END
```

FORTRAN DIAGNOSTIC RESULTS FOR PSPEC

NO ERRORS

?TRAN(1.1)/MSOS 4.2 INTEGER WORD SIZE = 1 PSR 2?7 DATE 11/20/73

```
      SUBROUTINE XSPEC(A,B,N,C,Q,M)
      REAL L
      DIMENSION A(2),B(2),C(2),Q(2)
      FN=N
      FM=M
      W=FN/FM
      MP1=M+1
      DO 100 INDL=1,MP1
      L=INDL-1
      XA=AMAX1(0.0,W*(L-0.5))
      XB=AMIN1(FN,W*(L+0.5))
      IXA=XA
      IXA=MAX0(0,IXA)
      IXAIND=IXA+1
      IXB=XB+1.0
      IXB=MIN0(IXB,N)
      IXBIND=IXB+1
      FINTC=0.0
      FINTQ=0.0
      DO 50 IXIND=IXAIND,IXBIND
      IX=IXIND-1
      RA=A(2*IXIND-1)
      SA=A(2*IXIND)
      RB=B(2*IXIND-1)
      SB=B(2*IXIND)
      AMPC=RA*RB+SA*SB
      AMPQ=RA*SB-RB*SA
      IF(IX.LE.IXA+1 .OR. IX.GE.IXB-1) GO TO 40
      FINTC=FINTC+AMPC
      FINTQ=FINTQ+AMPQ
      GO TO 50
   40 XAA=AMAX1(XA,FLOAT(IX)-0.5)
      XBB=AMIN1(XB,FLOAT(IX)+0.5)
      DELX=AMAX1(0.0,XBB-XAA)
      FINTC=FINTC+AMPC*DELX
      FINTQ=FINTQ+AMPQ*DELX
   50 CONTINUE
      C(INDL)=FINTC*.5
      Q(INDL)=FINTQ*.5
  100 CONTINUE
      RETURN
      END
```

1 FORTRAN DIAGNOSTIC RESULTS FOR XSPEC

NO ERRORS

?IN

```
      PROGRAM SUTPUT
C     SCALAR - VECTOR TIME SERIES CORRELATIONS
      COMMON XM1,XM2,LAG     ,FREPAR
      COMMON NAUG,NPTS
      COMMON X(1026,3),P(101,3),C(101,3),Q(101,3),PMM(101),PPP(101)
      COMMON DANING(101),SUMANG(101)
      COMMON FRQ(101),CO(101),PH(101)
      INTEGER PLOTLN(100)
      EQUIVALENCE(NB,LAG),(L1,NBP1)
      DIMENSION SUM(4)
      NCH=3
      LP=6
      REWIND 2
      BUFFER IN(2,1)(XM1,Q(101,3))
      IF(IFUNIT(2))1,801,1
801   CONTINUE
      WRITE(6,701)XM1,XM2,LAG
701   FORMAT(1H1/////,17H MOORING NUMBERS     ,A8,6H  AND  ,A8,///
     *5H LAG=,I4)
      CALL POLAR(C(1,2),Q(1,2),P(1,2),P(1,3),LAG,CO(1),PH(1))
      REWIND 2
      WRITE(6,603)
      WRITE(6,601)
      L1=LAG+1
      DO 2 I=1,L1
      AL=LAG
      AI=I-1
      FREQ=AI/AL*0.5*FREPAR
      IF(AI)3,3,4
3     ALFREQ=0.
      PERIOD=0.
      GO TO 5
4     PERIOD=1./FREQ
      ALFREQ=ALOG10(FREQ)
    5 IF(P(I,1))16,16,17
   16 ALPZ=0
      GO TO 18
   17 SW=P(I,1)
      ALPZ=ALOG10(SW)
   18 IF(P(I,2))6,6,7
6     ALPX=0.
      GO TO 8
7     SW=P(I,2)
      ALPX=ALOG10(SW)
    8 IF(P(I,3))9,9,10
9     ALPY=0.
      GO TO 11
   10 SX=P(I,3)
      ALPY=ALOG10(SX)
12    CO(I  )=CO(I  )**2
      CALL NFACTS(P(I,2),P(I,3),C(I,2),Q(I,2),AMAJ,AMIN,AP,AM,THETA,RC,P
     1P,PM)
      PPP(I)=PP
      PMM(I)=PM
```

```
      KANGLE=KANGE(KFIX(THETA))
      DANING(I)=-FLOAT(KANGLE)
      ARC=ABS(RC)
      IF(ARC.GT.0.9) DANING(I)=0.
 11   WRITE(6,602)
     $AI,FREQ,ALFREQ,PERIOD,P(I,1),P(I,2),P(I,3),C(I,2),Q(I,2),CO(I),
     $PH(I),AMAJ,AMIN,RC,KANGLE
      FRQ(I)=PERIOD
      P(I,2)=ALPX
      P(I,3)=ALPY
      SUMANG(I)=ALPZ
      CO(I     )=6.*CO(I      )-3.
 2    CONTINUE
      WRITE(6,803)
 803  FORMAT(1H1)
      XMIN=-3
      XMAX=3
C-    A PLOT IS WANTED
      SCALFAC=(XMAX-XMIN)/100.
      WRITE(61,23)
 23   FORMAT(1H1)
      WRITE(61,88) SCALFAC
 88   FORMAT(40X,15HSCALE FACTOR = ,F8.2)
      WRITE(61,82)XMIN,XMAX
 82   FORMAT(11X,F6.2,97X,F5.2/)
      WRITE(61,79)
 79   FORMAT(14X,102(1H+))
      AL = LAG
      L1=LAG +1
      DO 15  I=1,L1
      AI=I-1
      FREQ=AI/AL*.5*FREPAR
      IF(AI)90,90,91
 90   PERIOD=0.
      GO TO 92
 91   PERIOD=1./FREQ
 92   CALL BLANKV(PLOTLN(1),100)
      PZZ=SUMANG(I)
      JK=(PZZ    -XMIN)/SCALFAC+.5001
      IF(JK.LE.100.AND.JK.GE.1)PLOTLN(JK)=1HZ
      IF(JK.GT.100)PLOTLN(100)=1H+
      IF(JK.LT.1)PLOTLN(1)=1H-
      JK=(P(I,2)-XMIN)/SCALFAC+.5001
      IF(JK.LE.100.AND.JK.GE.1) PLOTLN(JK)=1HX
      IF(JK.GT.100)PLOTLN(100)=1H+
      IF(JK.LT.1)PLOTLN(1)=1H-
      JK=(P(I,3)-XMIN)/SCALFAC+.5001
      IF(JK.LE.100.AND.JK.GE.1) PLOTLN(JK)=1HY
      IF(JK.GT.100)PLOTLN(100)=1H+
      IF(JK.LT.1)PLOTLN(1)=1H-
      CA=CO(I     )
      JK=(CA    -XMIN)/SCALFAC+.5001
      IF(JK.LE.100.AND.JK.GE.1) PLOTLN(JK)=1HC
      IF(JK.GT.100)PLOTLN(100)=1H+
```

```
      IF(JK.LT.1) PLOTLN(1) = 1H-
      WRITE(61,1000) PERIOD,PLOTLN
1000  FORMAT(2X,F7.3,5X,1H*,100A1,1H*)
15    CONTINUE
      WRITE(6,605)
      WRITE(6,606)
39    CONTINUE
      REWIND 2
      BUFFER IN(2,1)(XM1,Q(101,3))
      IF(IFUNIT(2))1,33,1
33    REWIND 2
      BUFFER OUT(2,1)(XM1,NPTS)
      BUFFER OUT(2,1)(P(1   ,1),Q(101,3))
      IF(IFUNIT(2))1,34,1
34    CONTINUE
      BUFFER OUT(2,1)  (PMM(1),SUMANG(101))
      J=IFUNIT(2)
      BUFFER OUT(2,1)  (FRQ(1),PH(101))
      IF(IFUNIT(2))1,32,1
32    ENDFILE 2
      REWIND 2
1     STOP
601   FORMAT(1H1,  134HLAG  FREQ  LNFREQ PERIOD POWER(Z)   POWER(U)             M
     $POWER(V)        COSPECTRUM  QUADSPEC COHERNC PHASE   MAJ
     $IN      ROT COEF   T        )
602   FORMAT(1H ,1X,F4.0,1X,3(F6.2,1X), E10.2,1X,E10.2,1X,E10.2, 1
     $X,2(E9.2,1X),1X,F7.3,1X,F7.2,3(E9.2),1X,I3)
603   FORMAT(1H1 //////////,25X,3H U   )
605   FORMAT(1H1 //////////,25X,3H UP )
604   FORMAT(1H+,30X,3H V     )
606   FORMAT(1H+,30X,3H VP   )
607   FORMAT(1H1 ////////// 25X,10H U      UP                )
608   FORMAT(1H1 ////////// 25X,10H V      VP                )
      END
```

1 FORTRAN DIAGNOSTIC RESULTS FOR SUTPUT

NO ERRORS

CED STATEMENT LABELS

039   00604   00607   00608

RTRAN(1.1)/MSOS 4.2 INTEGER WORD SIZE = 1 PSR 287 DATE 11/27/73

```
      SUBROUTINE  POLAR(C,Q,X,Y,L,COHERE,PHASE)
      DIMENSION C(2),Q(2),X(2),Y(2),COHERE(2),PHASE(2)
      LP1=L+1
      PI=3.1415926536
      EPS=1.E-15
      XYMAX=0.
      DO 20 K=1,LP1
      XY=X(K)*Y(K)
      IF(XY.GT.XYMAX) XYMAX=XY
 20   CONTINUE
      DO 40 K=1,LP1
      XY=X(K)*Y(K)
      IF(XY.LE.XYMAX*EPS) GO TO 30
      COHERK=SQRT((C(K)*C(K)+Q(K)*Q(K))/XY)
      IF((Q(K)**2+C(K)**2).EQ.0.)GO TO 41
      PHASE(K)=ATAN2(Q(K),C(K))*180./PI
      GO TO 42
 41   PHASE(K)=0.
 42   CONTINUE
      COHERE(K)=COHERK
      GO TO 40
 30   COHERE(K)=0.
      PHASE(K)=0.
 40   CONTINUE
      RETURN
      END
```

( FORTRAN DIAGNOSTIC RESULTS FOR POLAR

NO ERRORS

```
ORTRAN(1.1)/MSOS 4.2  INTEGER WORD SIZE = 1     PSR 287      DATE    11/27/73

      FUNCTION KANGE(N)
5     IF(N.LT.360)GO TO 11
      GO TO 2
11    IF(N.GE.0)GO TO 1
      GO TO 3
1     KANGE=N
      RETURN
2     N=N-360
      GO TO 5
3     N=N+360
      GO TO 5
      END

1 FORTRAN DIAGNOSTIC RESULTS FOR KANGE

NO ERRORS
```

FTRAN(1.1)/MSOS 4.2 INTEGER WORD SIZE = 1 PSR 287 DATE 11/27/73

```
      FUNCTION KFIX(X)
      CONV=ATAN(1.)/45.
      X=X/CONV
      KFIX=IFIX(SGN(X)*(ABS(X)+.5))
      RETURN
      END
```

1 FORTRAN DIAGNOSTIC RESULTS FOR KFIX

NO ERRORS

ORTRAN(1.1)/MSOS 4.2  INTEGER WORD SIZE = 1    PSR 287    DATE   11/27/73

```
      FUNCTION SGN(X)
      IF(X.EQ.0.)GO TO 2
      SGN=X/ABS(X)
1     RETURN
2     SGN=0.
      GO TO 1
      END
```

I FORTRAN DIAGNOSTIC RESULTS FOR SGN

NO ERRORS

FTRAN(1.1)/MSOS 4.2 INTEGER WORD SIZE = 1 PSR 287 DATE 11/27/73

```
      SUBROUTINE NFACTS(PU,PV,PUV,QUV,MAJ,MIN,AP,AM,THETA,RC,PP,PM)
      REAL MAJ,MIN
      RC=-2.*QUV/(PU+PV)
      PP=.25 *  (2.*PU+2.*PV+4.*QUV)
      PM=.25*   (2.*PU+2.*PV-4.*QUV)
      AP=SQRT(PP)
      AM=SQRT(PM)
      THETA=.5*ATAN2(2.*PUV,PU-PV)
      MAJ=ABS(AP+AM)
      MIN=ABS(AP-AM)
      RETURN
      END
```

FORTRAN DIAGNOSTIC RESULTS FOR NFACTS

NO ERRORS

MIN

ERR

```
FTRAN(1.1)/MSOS 4.2  INTEGER WORD SIZE = 1      PSR 287       DATE    11/26/73

      PROGRAM SSECPLT
C     SCALAR - VECTOR TIME SERIES CORRELATIONS
      COMMON XM1,XM2,LAG    ,FREPAR
      COMMON NAUG,NPTS
      COMMON P(101,3),C(101,3),Q(101,3),PMM(101),PPP(101)
      COMMON DANING(101),SUMANG(101)
      COMMON FRQ(101)
      INTEGER PLOTLN(102)
      REWIND 2
      BUFFER IN(2,1)(XM1,NPTS)
      BUFFER IN(2,1)(P(1   ,1),Q(101,3) )
      BUFFER IN(2,1)(PMM(1),SUMANG(101))
      IF(IFUNIT(2))1,2,1
1     WRITE(6,600)
600   FORMAT(1H1,6HERROR     )
3     STOP
2     CONTINUE
      WRITE(6,701)XM1,XM2,LAG
701   FORMAT(1H1/////,17H MOORING NUMBERS     ,A8,6H  AND  ,A8,///
     *5H LAG=,I4)
      L1=LAG+1
      CALL ZEROV(FRQ,101)
      DO 5 I=1,L1
      IF(I.EQ.1)GO TO 123
      A=FLOAT(I-1)/FLOAT(LAG)*.5*FREPAR
      FRQ(I)=1./A
123   CONTINUE
      CALL COSPEC(C(I,1),Q(I,3),C(I,3),Q(I,1),TPP,TMM,DT,PPP(I),PMM(I),
     *DANING(I),P(I,1),P(I,2),P(I,3),C(I,2),SUMANG(I))
700   FORMAT(1H+,I3,2X,F6.2)
      K=I-1
      WRITE(6,700)K,FRQ(I)
5     CONTINUE
5501  FORMAT(1H1)
      WRITE(6,5501)
      WRITE(6,118)
 118  FORMAT(1X ,4HBAND,1X,6HPERIOD,2X,3H0.0,21X,3H0.5,21X,3H1.0,1X,4H-180,20X,3
     1H0,20X,3H0.0,20X,3H180)
      DO 15 I=1,L1
      CALL BLANKV(PLOTLN(1),102)
      JK=P(I,1)*50.+.5001
      IF(JK.LE.50.AND.JK.GE.1)PLOTLN(JK)=1H+
      JK=P(I,3)*50.+.5001
      IF(JK.LE.50.AND.JK.GE.1)PLOTLN(JK)=1H-
      PLOTLN(51)=1H*
      JK= P(I,2)*25./180.+77.5001
      PLOTLN(77)=1H.
      IF(JK.LE.102.AND.JK.GE.51)PLOTLN(JK)=1H+
      JK=C(I,2)*25./180.+77.5
      IF(JK.LE.102.AND.JK.GE.51)PLOTLN(JK)=1H-
      WRITE(6,1000)I,FRQ(I),PLOTLN
1000  FORMAT(1X,I4,2X,F7.1,1X,1H*,102A1,1H*)
  15  CONTINUE
      WRITE(6,118)
```

```
      WRITE(6,5501)
      WRITE(6,119)
  119 FORMAT(1X,4HBAND,1X,6HPERIOD,2X,4H-180.18X,4H-90.,23X,3H0.0,21X,3H909,21X,
     19..,21X,4H180.)
      DO 16 I=1,L1
      CALL BLANKV(PLOTLN(1),102)
      JK=PPP(I   )*50./180.+50.5 1
      PLOTLN(76)=1H.
      PLOTLN(26)=1H.
      PLOTLN(51)=1H.
      IF(JK.LE.101.AND.JK.GE.1)PLOTLN(JK)=1H+
      JK=PMM(I   )*50./180.+50.5:1
      IF(JK.LE.101.AND.JK.GE.1)PLOTLN(JK)=1H-
      JK=    50./180.         *C(I,1)     +50.501
      IF(JK.LE.101.AND.JK.GE.1)PLOTLN(JK)=1HC
      JK=   50./180.          *Q(I,3)          +50.501
      IF(JK.LE.101.AND.JK.GE.1)PLOTLN(JK)=1HA
      WRITE(6,1000)I,FR Q(I),PLOTLN
   16 CONTINUE
      WRITE(6,119)
      REWIND 2
      GO TO 3
      END
```

( FORTRAN DIAGNOSTIC RESULTS FOR SSECPLT

NO ERRORS

ORTRAN(1.1)/MSOS 4.2 INTEGER WORD SIZE = 1 PSR 287 DATE 11/26/73

```
      SUBROUTINE COSPEC(PUU,QUV1,PUV1,QUU,TPP,TMM,DT,P6,P5,D1,C2,TP,C1,
     1TM,SA)
      PZ=C2*.5
      C=180./3.14159265
      PXPXP=.5*(PUU+QUV1)
      QXPXP=.5*(PUV1-QUU)
      QXMXM=.5*(PUV1+QUU)
      PXMXM=.5*(PUU-QUV1)
      TPP=-ATAN2(QXPXP,PXPXP)
      TMM= ATAN2(-QXMXM,PXMXM)*(-1.)
      Q2=QXPXP
      Q1=QXMXM
      P1=PXMXM
      P2=PXPXP
C
      C1=(P1*P1+Q1*Q1)/(PZ*P5)
      C2=(P2*P2+Q2*Q2)/(PZ*P6)
      ANG=-.5*ATAN2(Q2*P1-Q1*P2,P1*P2+Q2*Q1)*C
      TM=C*TMM
      TP=C*TPP
   25 TL=(TP-D1)
      TO=(D1+TM)
      IF(ANG.GT.0.)ANG2=ANG-180.
      IF(ANG.LT.0.) ANG2=ANG+180.
      PUU=ANG
      QUV1=ANG2
      P6=TL
      P5=TO
      IF(TL.GT.180.)P6=TL-360.
      IF(TL.LT.-180.)P6=TL+360.
      IF(TO.GT.180.)P5=TO-360.
      IF(TO.LT.-180.)P5=TO+360.
      WRITE(6,600)C2,TP,C1,TM,P6,P5,ANG2,ANG
  600 FORMAT(1H ,11X
     2          ,4HC++=,F5.2,2X,8HTHETA++=,F6.1,2X,4HC--=,F5.2,2X,8HTHET
     1A--=,F6.1,2X,5HANG+=.F6.1,5HANG-=,F6.1,2X,10HCOMPL.ANG=,F6.1,1X,
     24HANG=,F6.1)
      RETURN
      END
```

FORTRAN DIAGNOSTIC RESULTS FOR COSPEC

NO ERRORS

CED STATEMENT LABELS

MIN

URTRAN(1.1)/MSOS 4.2 INTEGER WORD SIZE = 1 PSR 287 DATE 11/26/73

```
      PROGRAM SDATA1
C     SCALAR - VECTOR TIME SERIES CORRELATIONS
      REAL MOOR
      COMMON XM1,XM2,LAG ,FREPAR
      COMMON NAUG,NPTS
      COMMON X(1026,3)
      COMPLEX I,U,V,W1,W2  ,W3,W4
      PI=3.1415
      CONV=ATAN(1.)/45.
      READ(5,81) NPTS,LAG,FREPAR
 81   FORMAT(2(5X,I5),F10.0)
      I=CMPLX(0.,1.)
      W3=CEXP(I*CONV*87.3)
      W4=CEXP( I*CONV*70.)
      PH1=-120.*CONV
      PH2=    CONV*151.
      PH3=    CONV*172.
      W2=CEXP(I*CONV*(-65.))
      W1=CEXP(I*CONV*70.5*(-1.)    )
      PH4=CONV*130.
      DO 1 J=1,1024
      T=FLOAT(4*J)*PI*.1
      T4=FLOAT(7*J)*PI*.1
      T3=FLOAT(6*J)*PI*.1
      T1=FLOAT(3*J)*PI*.1
      T6=FLOAT(8*J)*PI*.1
      T2=FLOAT(5*J)*PI*.1
      U=1.0*COS(T)+I*SIN(T)
      U=U*W1
      U=U+(1.*COS(T2)-I*SIN(T2))*W3
      U=U+(1.*COS(T3)+I*SIN(T3))*W4
      U=U+(1.*COS(T6)-I*SIN(T6))*W4
      U=U+(5.*COS(T1+PH4)-I*SIN(T1+PH4))*W1
      U=U+(1.*COS(T4-PH3)-I*SIN(T4-PH3))*W1
      U=U+.1*RANDNO(0)*CMPLX(0.,1.)
      U=U+.1*RANDNO(0)*CMPLX(1.,0.)
      V=1.4*COS(T-PH2)
      V=V+2.*COS(T1)
      V=V+2.*COS(T6+PH1)
      V=V+4.2*COS(T4)
      V=V+3.*COS(T2-PH3)
      V=V+.1*RANDNO(0)*CMPLX(0.,1.)
      V=V+.1*RANDNO(0)*CMPLX(1.,0.)
      X(J,2)=REAL(U)
      X(J,3)=AIMAG(U)
      X(J,1)=REAL(V)
 1    CONTINUE
      CALL SPEAK(1024)
      STOP
      END
```

```
      SUBROUTINE SPEAK(NUM)
      COMMON XM1,XM2,LAG    ,FREPAR
      COMMON NAUG,NPTS
      COMMON X(1026.3),P(101,3)
      COMMON C(101,3),Q(101,3),CO(101),PH(101)
      NPTS=NUM
      NAUG=0
      CALL ZEROV(P,1212)
      REWIND 2
      BUFFER OUT(2,1)(XM1,PH(101))
      IF(IFUNIT(2))1,2,1
    2 CONTINUE
      ENDFILE 2
      REWIND 2
      BUFFER IN(2,1)(XM1,PH(101))
      KANG=IFUNIT(2)
    1 REWIND 2
      RETURN
      END
```

I FORTRAN DIAGNOSTIC RESULTS FOR SPEAK

NO ERRORS

MIN

18165