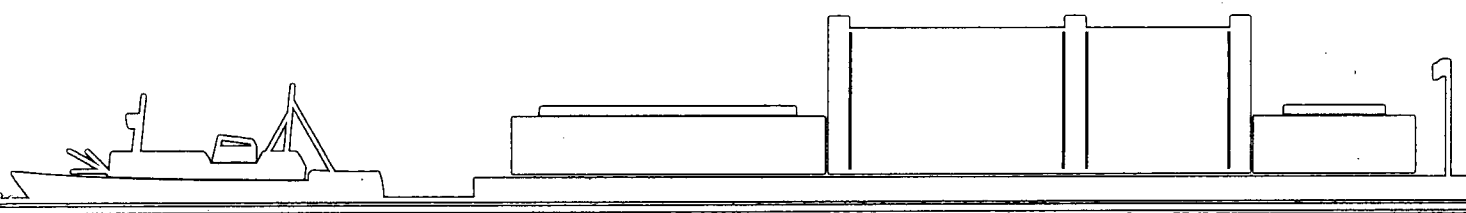


# CANADA CENTRE FOR INLAND WATERS

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

	Page
Introduction	1
I Theory of Complex Time Series	1
A. Analysis of a single vector time series	1
B. Analysis of a pair of vector time series	5
C. Test Cases	11
D. Analysis of scalar time series and a vector time series	13
II Application to Model Verification	14
III Computer Programme Description	16
IV Conclusions	17
Acknowledgements	18
References	19
Table 1A	20
Table 1B	20
List of Figures	21
Appendix I	26
Appendix II	63

## 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

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

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

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

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

Now W may be written as

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

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

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

$$\begin{aligned} W^1 &= (|A^+| + |A^-|) \cos [\sigma t + 1/2(\alpha^+ - \alpha^-)] \\ &\quad + i(|A^+| - |A^-|) \sin [\sigma t + 1/2(\alpha^+ - \alpha^-)] \\ &= \text{Maj} \cos [\sigma t + 1/2(\alpha^+ - \alpha^-)] \\ &\quad + i \text{Min} \sin [\sigma t + 1/2(\alpha^+ - \alpha^-)] \end{aligned}$$

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

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

$P_{uv} = (a_1 a_2 + b_1 b_2)/2$ , where  $P_{uv}$  is the cospectrum

and  $Q_{uv} = (a_1 b_2 - a_2 b_1)/2$ , where  $Q_{uv}$  is the quadrature spectrum

Then by substitution

$$\begin{aligned} |A^+| &= 1/2(2Pu + 2Pv + 4Quv)^{1/2} \\ &= \frac{1}{\sqrt{2}} (Pu + Pv + 2Quv)^{1/2} \end{aligned}$$

$$\begin{aligned} |A^-| &= \frac{1}{2} (2Puv + 2Pv - 4Quv)^{1/2} \\ &= \frac{1}{\sqrt{2}} (Pu + Pv - 2Quv)^{1/2} \end{aligned}$$

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

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

$$\begin{aligned} \tan 2\theta &= \tan(\alpha^+ + \alpha^-) = \frac{\tan \alpha^+ + \tan \alpha^-}{1 - \tan \alpha^+ \tan \alpha^-} \\ &= \frac{(a_2 + b_1)(a_1 + b_2) + (a_1 - b_2)(a_2 - b_1)}{(a_1 - b_2)(a_1 + b_2) - (a_2 + b_1)(a_2 - b_1)} \\ &= (2a_2a_1 + 2b_1b_2)/(a_1^2 + b_1^2 - (a_2^2 + b_2^2)) \\ &= 2Puv/(Pu - Pv) \end{aligned}$$

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

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,

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

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  
and  $W'(t) = (u'(t), v'(t))$  velocity,  $v$  is the north component  
of velocity of  $W(t)$ .

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

$$2 P_{uu'} = a_1 a_1' + b_1 b_1'$$

$$2 P_{vv'} = a_2 a_2' + b_2 b_2'$$

$$2 P_{uv'} = a_1 a_2' + b_1 b_2'$$

$$2 P_{vu'} = a_1' a_2 + b_1' b_2$$

which are generalizations of the previously defined quantity  $P_{uv}$ , and the quadrature components from  $Q_{uv}$  are:

$$2 Q_{uu'} = a_1 b_1' - a_1' b_1$$

$$2 Q_{vv'} = a_2 b_2' - a_2' b_2$$

$$2 Q_{uv'} = a_1 b_2' - a_2' b_1$$

$$2 Q_{vu'} = a_2 b_1' - a_1' b_2$$



The positive and negative rotating components of  $W$  and  $W'$  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  $W'^+ = 1/2\{(a_1' + b_2') + i(a_2' - b_1')\}$

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

Alternatively, since

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

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

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

$$\begin{aligned} \langle W^+ W'^{+*} \rangle &= |A^+| |A'^+| e^{i(\sigma t + \phi) - i(\sigma t + \phi')} e^{i(\theta - \theta')} \\ &= P W^+ W'^+ - i Q W^+ W'^+ \\ &= |A^+| |A'^+| e^{i\phi^{++}} \end{aligned}$$

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

$$\begin{aligned} \phi^{++} &= \arctan(Q W^+ W'^+ / P W^+ W'^+) \\ &= \theta - \theta' + \phi - \phi' \end{aligned}$$

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,  $C^{++}$

$$C^{++} = (P^2 W^+ W'^+ + Q^2 W^+ W'^+) / (P W^+ W'^+ P W^+ W'^+)$$

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

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

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

resulting in the phase difference between the vectors

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

As before the coherency squared between the negatively rotating components is

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

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

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

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  $\langle W^+ W^- \rangle$

$$\begin{aligned} \langle W^+ W^- \rangle &= |A^+| |A^-| e^{i(\sigma t + \phi)} - i(\sigma t + \phi) e^{i(\theta + \theta')} \\ &= P^+ w^- - iQ^+ w^- \\ &= |A^+| |A^-| e^{i\phi^{+-}} \end{aligned}$$

The phase difference between the two components is

$$\phi^{+-} = \arctan (Q^+ w^- / P^+ w^-)$$

and the phase difference between the two series is

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

as before

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

where  $P^+ w^+ \equiv 1/2(Pu\dot{u} - Pvv - Qu\dot{v} - Qv\dot{u})$

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

and  $Pw^+w^+$  and  $Pw^-w^-$  are defined above

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

$$\begin{aligned} \langle W^-W^+ \rangle &= |A^-||A^+| e^{-i(\sigma t + \phi)} + i(\sigma t + \phi') e^{i(\theta + \theta')} \\ &= Pw^-w^+ - iQw^-w^+ \\ &= |A^-||A^+| e^{i\phi^{-+}} \end{aligned}$$

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

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

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

$$\text{Finally } C^{-+} = \left( \frac{P^2w^-w^+ + Q^2w^-w^+}{Pw^-w^- Pw^+w^+} \right)$$

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

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

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

$$\begin{aligned} \phi - \phi' &= 1/2(\phi^{+-} - \phi^{-+}) \\ &= 1/2 \arctan \left[ \frac{Qw^+w^-/Pw^+w^- - Qw^-w^+/Pw^-w^+}{1 + \frac{Qw^+w^-}{Pw^+w^-} \frac{Qw^-w^+}{Pw^-w^+}} \right] \pm \pi \end{aligned}$$

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,  $f_1, f_2, \dots, f_6$

$$U = \text{Real } W$$

$$V = \text{Imag } W$$

$$U^1 = \text{Real } W^1$$

$$V^1 = \text{Imag } W^1$$

where

$$\begin{aligned} W = & \text{Random Noise} + (2 \cos(f_1 t) + i \sin(f_1 t))e^{i87.3^\circ} \\ & + (1.4 \cos(f_2 t - 151^\circ) - .7i \sin(f_2 t - 151^\circ))e^{-i65^\circ} \\ & + (3 \cos(f_3 t - 172^\circ) + i \sin(f_3 t - 172^\circ))e^{-i70.5^\circ} \\ & + 4 \cos(f_4 t + 120^\circ) \\ & + (4.2 \cos(f_5 t) - i \sin(f_5 t))e^{-i65^\circ} \\ & + (\cos(f_6 t - 120^\circ) - 2i \sin(f_6 t - 120^\circ))e^{-i65^\circ} \\ W^1 = & \text{Random Noise} + (5. \cos(f_1 t + 130^\circ) - i \sin(f_1 t + 130^\circ))e^{-i70.5^\circ} \\ & + (3 \cos(f_2 t) + i \sin(f_2 t))e^{-i70.5^\circ} \\ & + (3 \cos(f_3 t) - i \sin(f_3 t))e^{i87.3^\circ} \\ & + (9 \cos(f_4 t) + i \sin(f_4 t))e^{i70.^\circ} \\ & + (3 \cos(f_5 t - 172^\circ) - i \sin(f_5 t - 172^\circ))e^{-i70^\circ} \\ & + (9 \cos(f_6 t) - i \sin(f_6 t))e^{i70^\circ} \end{aligned}$$

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.

$$\begin{aligned}
 W = & \text{random noise} + (\cos (f_1 t) + i \sin (f_1 t)) e^{i 87.3^\circ} \\
 & + (7(\cos (f_2 t - 151^\circ) - i \sin (f_2 t - 151^\circ)) e^{-i 65^\circ} \\
 & + (\cos (f_3 t - 172^\circ) + i \sin (f_3 t - 172^\circ)) e^{-i 70.5^\circ} \\
 & + 4 \cdot \cos (f_4 t + 120^\circ) \\
 & + (\cos (f_5 t) - i \sin (f_5 t)) e^{-i 65^\circ} \\
 & 2(\cos (f_6 t - 120^\circ) - i \sin (f_6 t - 120^\circ)) e^{-i 65^\circ}
 \end{aligned}$$

$$\begin{aligned}
 W^1 = & \text{random noise} + (\cos (f_1 t + 130^\circ) - i \sin (f_1 t + 130^\circ)) e^{-i 70.5^\circ} \\
 & + (\cos (f_2 t) + i \sin (f_2 t)) e^{-i 70.5^\circ} \\
 & + (\cos (f_3 t) - i \sin (f_3 t)) e^{i 87.5^\circ} \\
 & + (\cos (f_4 t) + i \sin (f_4 t)) e^{i 70^\circ} \\
 & + (\cos (f_5 t - 172^\circ) - i \sin (f_5 t - 172^\circ)) e^{-i 70.5^\circ} \\
 & + (\cos (f_6 t) - i \sin (f_6 t)) e^{i 70^\circ}
 \end{aligned}$$

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,  $(\phi - \hat{\phi})^{+\pm}$  or  $(\phi - \hat{\phi})^{-\pm}$ .

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

Let a scalar series be defined by

$$\begin{aligned} Z &= a_1 \cos \sigma t + b_1 \sin \sigma t \\ &= (|A^+| + |A^-|) (\cos(\sigma t + \phi)) \\ &= 2|A^+| \cos(\sigma t + \phi) \end{aligned}$$

and a complex series  $W$

$$\begin{aligned} W &= \{(|A^+| + |A^-|) \cos(\sigma t + \hat{\phi}) \\ &\quad + i(|A^+| - |A^-|) \sin(\sigma t + \hat{\phi})\} e^{i\theta} \end{aligned}$$

Essentially all the expressions developed for the 2 complex time series carry over to this case with the modifications required by  $a_2 = b_2 = \theta = Pvv^1 = Qvv^1 = Qvu^1 = Pvv = Puv = Quv = 0$ . 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 elliptically 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 the 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  $6^{\circ}$  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 $W/W^{**}$	$(\theta + \theta')$	$(\theta - \theta')$	$C^{+\pm}$	$\phi^{+\pm}$	$C^{-\pm}$	$\phi^{-\pm}$	$(\phi - \phi')^{+\pm}$	$(\phi - \phi')^{0+\pm}$	$(\phi - \phi')Ave,$	$(\phi - \phi')Ave \pm 180^0$
f <sub>1</sub>	-.49	17	-200	1.0	-113.1	1.0	147.9	-131.1	-129.9	49.5	-130.5
f <sub>2</sub>	.75	230	6	1.0	74.8	1.0	15.6	-151.2	-149.6	29.4	-150.4
f <sub>3</sub>	-1.	17	200	1.0	-154.9	1.0	-171.3	-171.9	-171.7	8.2	-171.8
f <sub>4</sub>	-	70	-70	1.0	50.1	1.0	170.2	120.1	119.8	-60	120
f <sub>5</sub>	1.3	230	5	1.0	176.7	1.0	-166.7	171.7	171.7	-8.3	171.7
f <sub>6</sub>	.275	5	130	1.0	104.9	1.0	-15.1	-120.1	-119.9	60	-120

TABLE 1B

f <sub>1</sub>	-1	-35	320	1	-113	.06	-121.8	-113	-128.7	76.5	-103.5
f <sub>2</sub>	-1.	32	-10	.1	128	1.00	-15	-150	-15	56.4	-123.6
f <sub>3</sub>	-1.	37	270	1.	-155	.26	-156.5	-155	-166.5	0.6	-179.4
f <sub>4</sub>	-	-4	-360	1.0	50	.22	-42.5	-50	-46.5	46.3	-133.7
f <sub>5</sub>	1	-11	-310	.23	12.4	.100	166.7	-38.6	166.7	89.5	-90.5
f <sub>6</sub>	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

- Figure 1 Spectral densities of the east and north components of current for the nearshore station, cgs units.
- Figure 2 Spectral densities of the east and north components of current for the offshore station, cgs units.
- Figure 3 (a) Phase angles between the two current series associated with the two oppositely rotating components.  
(b) Coherences between the two current series associated with the two oppositely rotating components.
- Figure 4 (a) Phase angles between the thermocline displacement and upper layer currents for each of the two oppositely rotating components.  
(b) Coherences between the thermocline displacement and current at station 6 for each of the two rotating components.



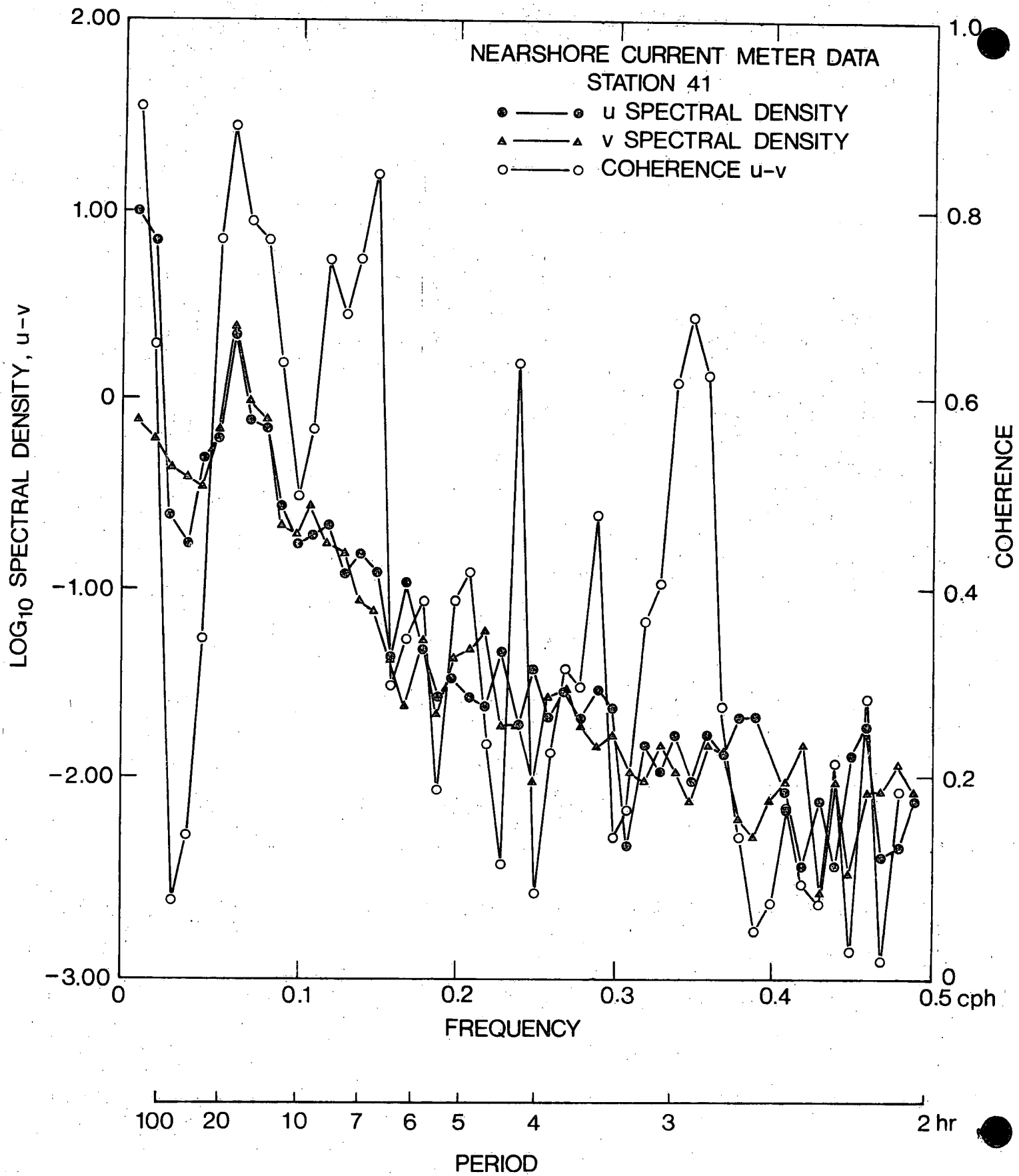


FIGURE.1

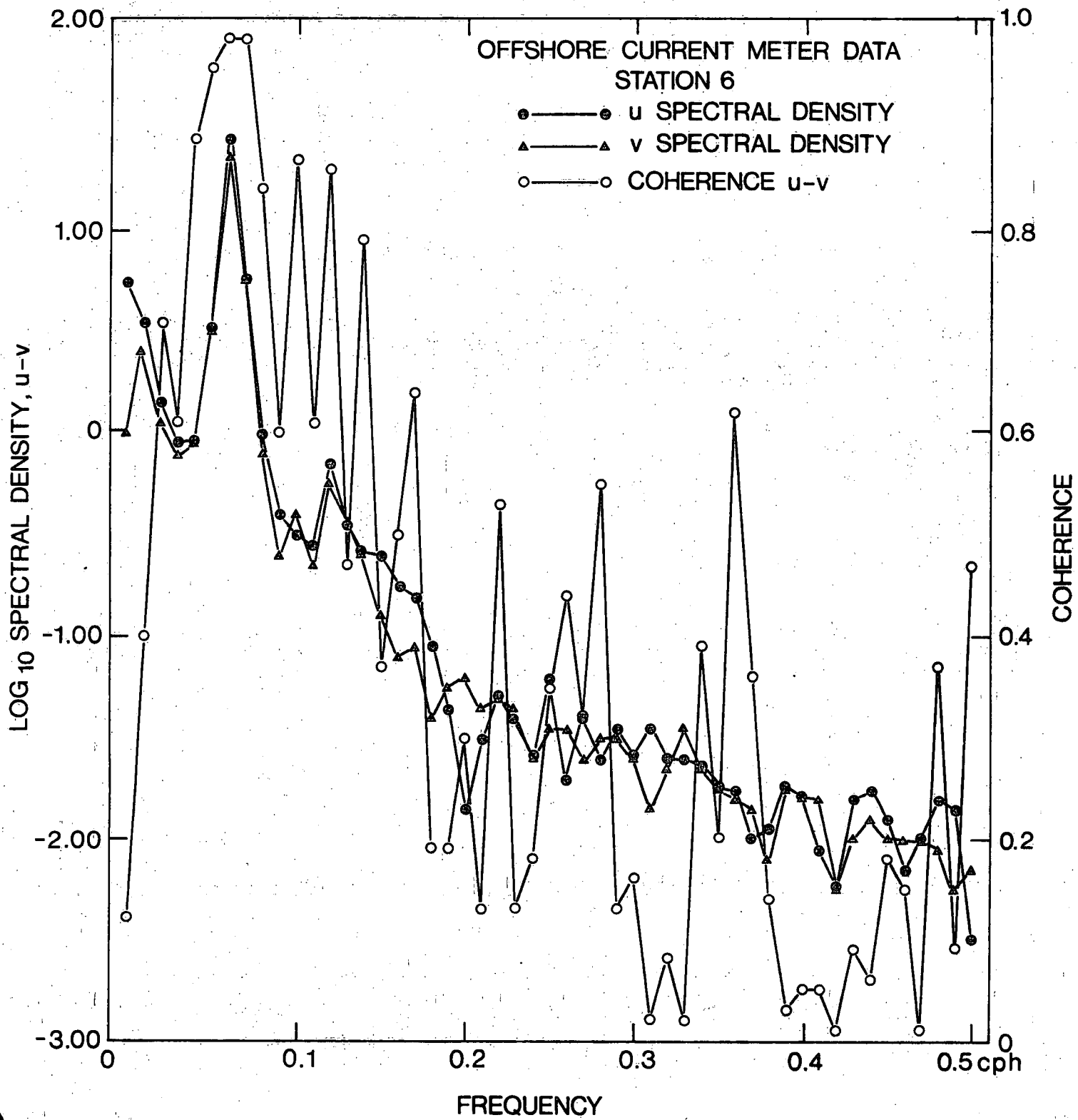


FIGURE. 2

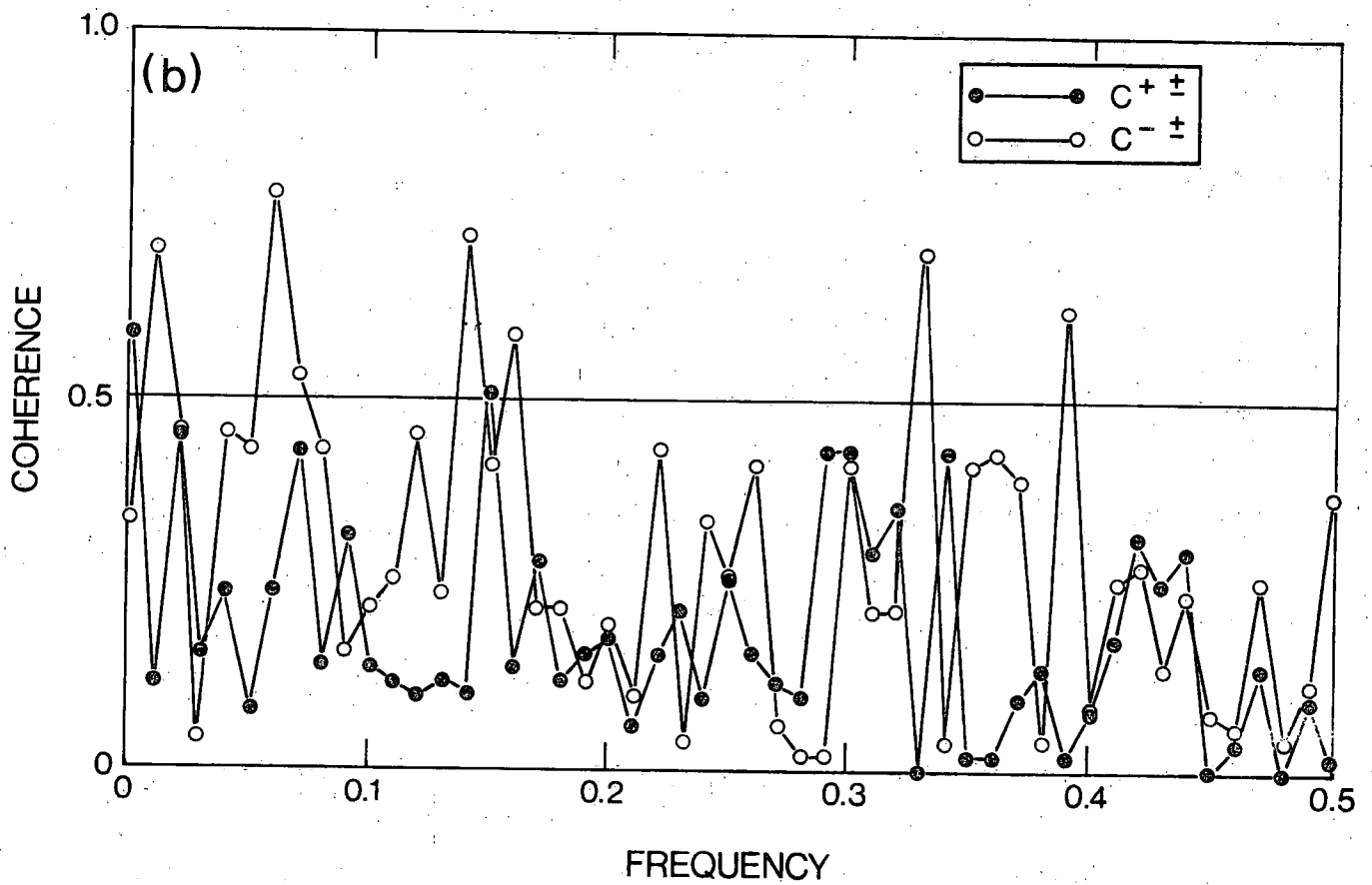
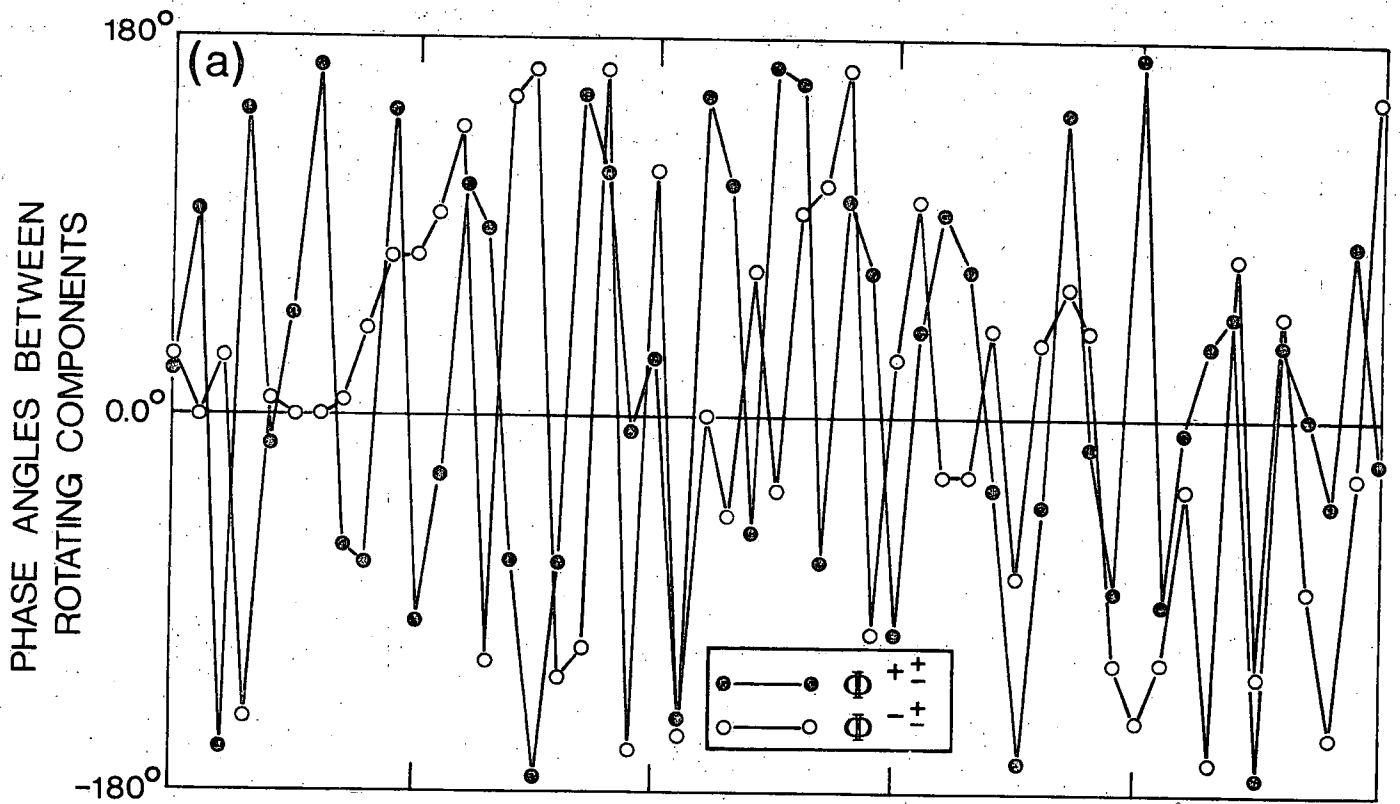
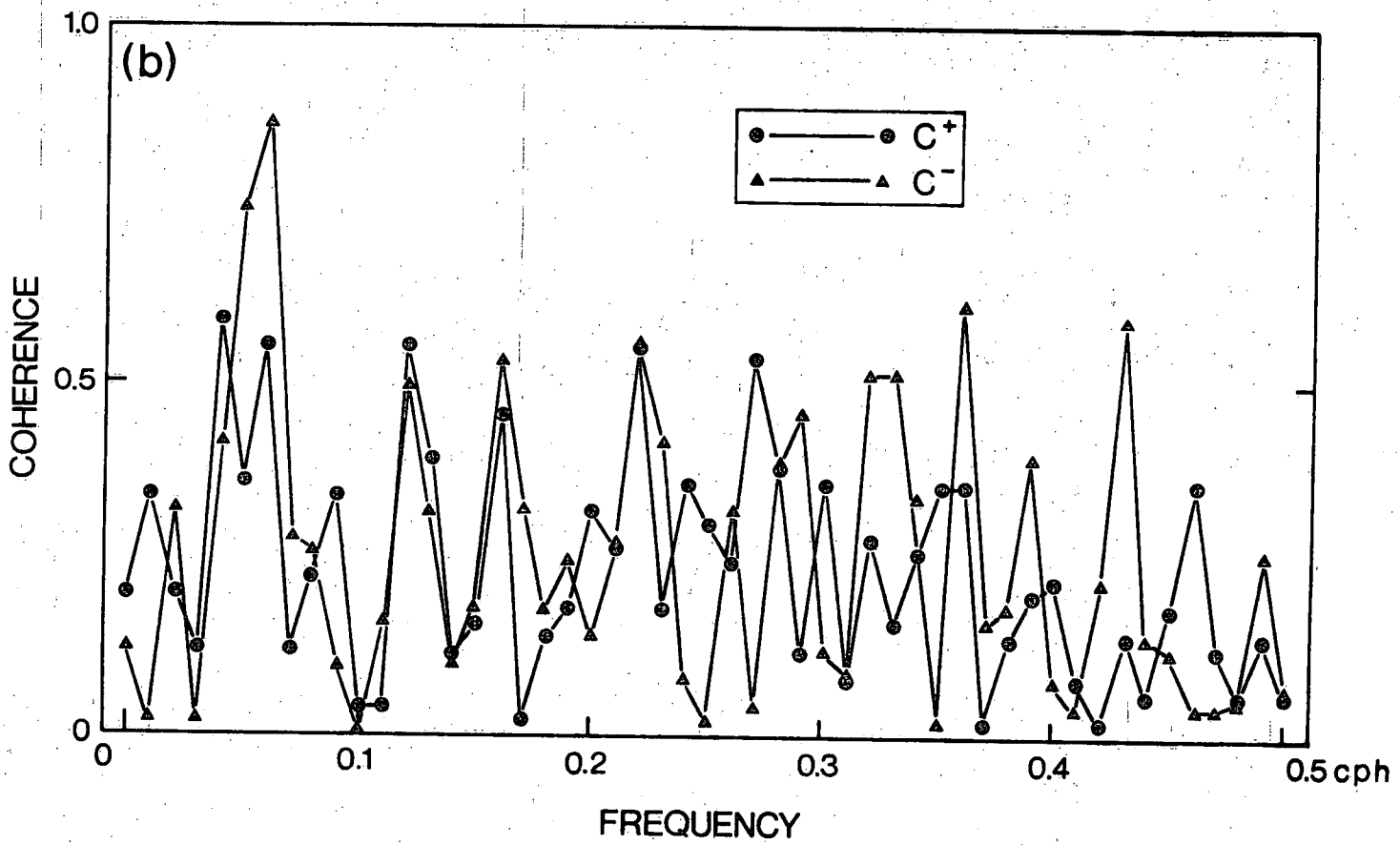
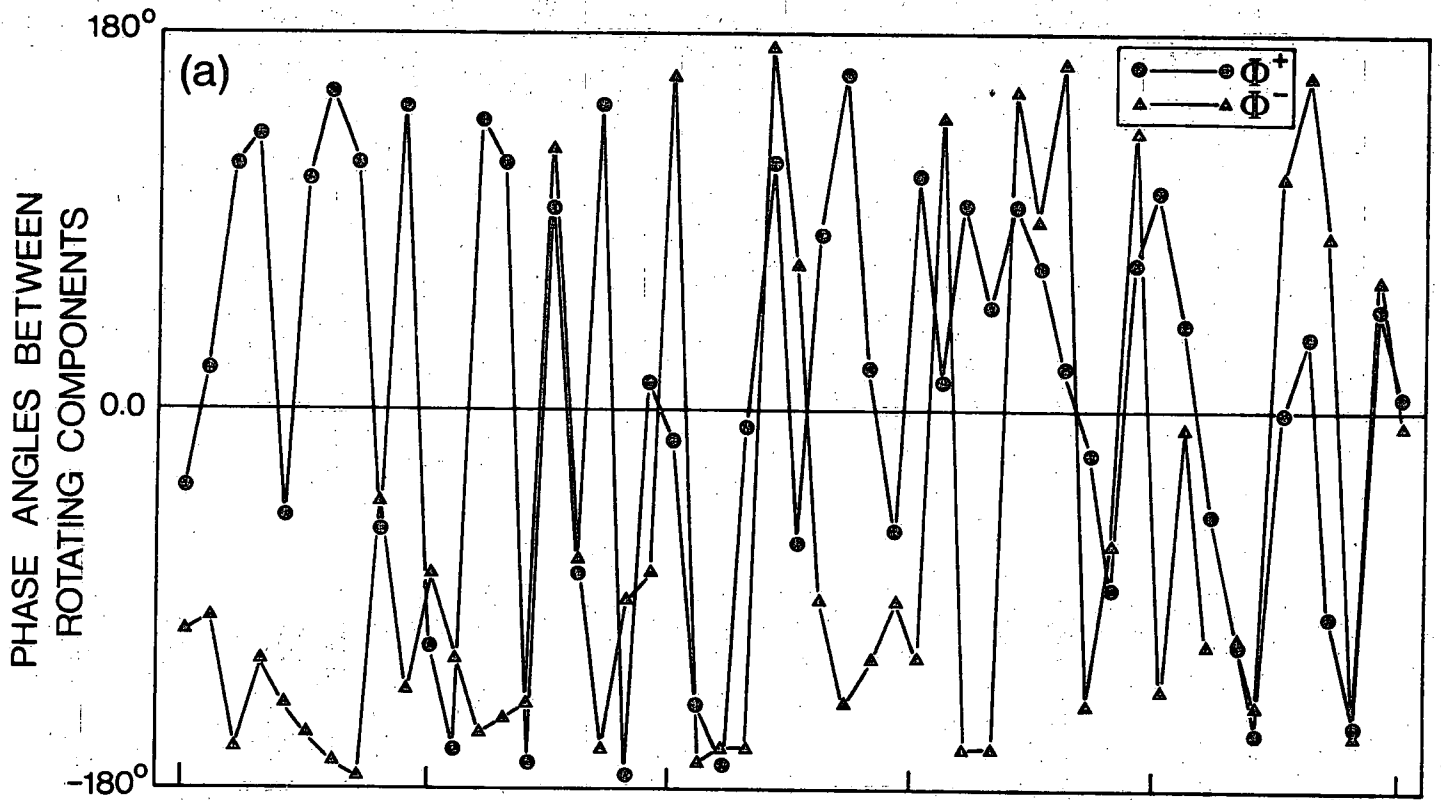


FIGURE .3



FREQUENCY

FIGURE .4

APPENDIX I

Programme Listing for vector-vector correlations

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 Fseran

CALLED BY

Data 1

STORAGE REQUIREMENTS

SOURCE LANGUAGE

MACHINE

PROGRAMMER

REFERENCES

```
LN 0001      PROGRAM CALCATE
LN 0002      C
LN 0003      C
LN 0004      COMMON XM1,XM2,LAG      ,FREFAR
LN 0005      COMMON NAUG,NPTS
LN 0006      COMMON X(1026,4),P(101,4)
LN 0007      COMMON C(101,6),S(101,6),SO(101,2),PH(101)
LN 0008      REWIND 2
LN 0009      BUFFER IN(2,1)(X(1,PH(101)))
LN 0010      IF (IFUNIT(2))1,2,1
LN 0011      2      CONTINUE
LN 0012      WRITE(6,753)NPTS
LN 0013      753  FORMAT(1H1,I5)
LN 0014      WRITE(6,701)XM1,XM2,LAG
LN 0015      701  FORMAT(1H1////////,17H MOORING NUMBERS      ,A8,6H      AND      ,A8,///
LN 0016      $5H LAG=,I4)
LN 0017      CALL SPECTR(NPTS)
LN 0018      1      REWIND 2
LN 0019      STOP
LN 0020      END
```

USASI FORTRAN DIAGNOSTIC RESULTS FOR CALCATE

NO ERRORS

```
LN 0001      SUBROUTINE SPECTR(NUM)
LN 0002      COMMON XM1,XM2,LAG ,FREPAR
LN 0003      COMMON NAUG,NPTS
LN 0004      COMMON X(1026,4),P(101,4)
LN 0005      COMMON C(101,6),Q(101,6),CU(101,2),PH(101)
LN 0006      CALL ZEROV(P,1919)
LN 0007      WRITE(6,6699)XM1,XM2
LN 0008      6699 FORMAT(1H1,///5X,A8,5X,AB/////))
LN 0009      NCH=4
LN 0010      ISKIP=0
LN 0011      800 CONTINUE
LN 0012      CALL FSCRAN(X,NUM,LAG,P,CU,PH,ISKIP,NCH,C,Q)
LN 0013      REWIND 2
LN 0014      BUFFER OUT(2,1)(XM1,PH(101))
LN 0015      IF (IFUNIT(2))901,902,901
LN 0016      902 ENDFILE 2
LN 0017      REWIND 2
LN 0018      901 RETURN
LN 0019      END
```

USASI FORTRAN DIAGNOSTIC RESULTS FOR SPECTR

NO ERRORS

UNREFERENCED STATEMENT LABELS

00800



NAME Fscan

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. R1fort calculates the Fourier coefficients. Pspec calculates the four power spectra. Finally Xspec calculates co and quadrature spectra for all pairs.

USAGE

INPUT

X, 4 time series

OUTPUT

P four power spectra  
C six cospectra  
Q six quadrature spectra

SUBROUTINES USED

CALLED BY

Spectra

STORAGE REQUIREMENTS

SOURCE LANGUAGE

MACHINE

PROGRAMMER

REFERENCES

```

LN 0001      SUBROUTINE FSCRAN(X,NPTS,NB,P,COH,PH,ISKIP,NCH,C,Q)
LN 0002      COMMON XXX,XXY,LLL      ,FREP PAR
LN 0003      COMMON NAUG,KIT
LN 0004      DIMENSION X(1026,4),P(101,4),C(101,6),Q(101,6),COH(101,2)
LN 0005      DIMENSION PH(101)
LN 0006      DIMENSION S(513),AVG(4),VAR(4),SUM(4)
LN 0007      KH=5
LN 0008      LP=6
LN 0009      KIT=NPTS
LN 0010      NBPI=NB+1
LN 0011      IF (ISKIP.EQ.1)GO TO 450
LN 0012      IND=0
LN 0013      DO 135 ICH=1,NCH
LN 0014      CALL AVVAR(X(I,ICH),NPTS,AVG(ICH),VAR(ICH))
LN 0015      IF (ICH.LT.4)GO TO 135
LN 0016      WRITE(LP,120)
LN 0017      120  FORMAT( /,9X,13H  MEAN      , 3X,4HVAR ,19X,4HMEAN, 8X,4HVAR
LN 0018      $2(19X,4HMEAN,8X,4HVAR  )/)
LN 0019      WRITE(LP,130)(AVG(I),VAR(I),I=1,4)
LN 0020      130  FORMAT(2X,7HX-COMP,2E11.3,5X ,7HY-COMP,2E11.3,5X,8HXP-COMP,
LN 0021      $2E11.3,5X,8HYP-COMP,2E11.3)
LN 0022      135  CONTINUE
LN 0023      DO 145 ICH=1,NCH
LN 0024      DO 140 I=1,NPTS
LN 0025      X(I,ICH)=X(I,ICH)-AVG(ICH)
LN 0026      140  CONTINUE
LN 0027      AVG(ICH)=0.
LN 0028      145  CONTINUE
LN 0029      FAC=.95
LN 0030      DEL=0
LN 0031      DAL=0
LN 0032      DIL=0.
LN 0033      DUL=0.
LN 0034      DO 146 I=2,NPTS
LN 0035      DEL2=FAC*DEL+.05*X(I,1)
LN 0036      DEL3=FAC*DAL+.05*X(I,2)
LN 0037      DEL4=FAC*DIL+.05*X(I,3)
LN 0038      DEL5=FAC*DUL+.05*X(I,4)
LN 0039      X(I-1,1)=X(I-1,1)-DEL
LN 0040      X(I-1,2)=X(I-1,2)-DAL
LN 0041      X(I-1,3)=X(I-1,3)-DIL
LN 0042      X(I-1,4)=X(I-1,4)-DUL
LN 0043      DAL=DEL3
LN 0044      DIL=DEL4
LN 0045      DUL=DEL5
LN 0046      146  DEL=DEL2
LN 0047      WRITE(6,25)((X(I,J),J=1,NCH),I=1,100)
LN 0048      25  FORMAT(3(1X,4E10.3))
LN 0049      C  WRITE(LP,150)
LN 0050      150  FORMAT(// 30H MEAN REMOVED FROM EACH SERIES )
LN 0051      160  CONTINUE
LN 0052      165  DO 170 ICH=1,NCH
LN 0053      170  CALL TAPER(X(I,ICH),NPTS)
LN 0054      C  WRITE(LP,180)

```

```

LN 0055 180 FORMAT(/ 44H COSINE TAPEK APPLIED TO ENDS OF EACH SERIES
LN 0056 190 CONTINUE
LN 0057 DO 200 ICH=1,NCH
LN 0058 200 CALL ZEFILL(X(1,ICH),NPTS,M,NAUG)
LN 0059 IF (NAUG.EQ.NPTS) GO TO 220
LN 0060 WRITE(LP,210)NPTS,NAUG
LN 0061 210 FORMAT( / 36H EACH SERIES AUGMENTED BY ZEROS FROM 15.8H PTS TO
LN 0062 2 15.4H PTS )
LN 0063 220 CONTINUE
LN 0064 CALL RLFORT(X(1,1),M+1,S,0,IFERR)
LN 0065 DO 270 ICH=1,NCH
LN 0066 CALL RLFORT(X(1,ICH),M+1,S,-3,IFERR)
LN 0067 IF (IFERR.EQ.0) GO TO 270
LN 0068 WRITE(LP,260) IFERR,ICH
LN 0069 260 FORMAT(22H FORT ERROR IFERR,ICH= 214)
LN 0070 115 RETURN
LN 0071 270 CONTINUE
LN 0072 NEB=NAUG/2
LN 0073 NEBP1=NEB+1
LN 0074 DO 290 ICH=1,NCH
LN 0075 CALL PSPEC(X(1,ICH),NEB,P(1,ICH),NB)
LN 0076 DO 290 I=1,NBP1
LN 0077 P(I,ICH)=P(I,ICH)*FLOAT(NAUG)/FLOAT(NPTS)
LN 0078 290 CONTINUE
LN 0079 360 FORMAT(/ 4H SUM,13X,4E12.3)
LN 0080 CALL XSPEC(X(1,1),X(1,2),NEB,C(1,1),Q(1,1),NB)
LN 0081 CALL XSPEC(X(1,3),X(1,4),NEB,C(1,2),Q(1,2),NB)
LN 0082 CALL XSPEC(X(1,1),X(1,3),NEB,C(1,3),Q(1,3),NB)
LN 0083 CALL XSPEC(X(1,1),X(1,4),NEB,C(1,4),Q(1,4),NB)
LN 0084 CALL XSPEC(X(1,2),X(1,3),NEB,C(1,5),Q(1,5),NB)
LN 0085 CALL XSPEC(X(1,2),X(1,4),NEB,C(1,6),Q(1,6),NB)
LN 0086 ZTL=FLOAT(NAUG)/FLOAT(NPTS)
LN 0087 DO 2825 I=1,NBP1
LN 0088 DO 2825 J=1,6
LN 0089 C(I,J)=C(I,J)*ZTL
LN 0090 Q(I,J)=Q(I,J)*ZTL
LN 0091 2825 CONTINUE
LN 0092 SUMC1=0.
LN 0093 SUMC2=0.
LN 0094 SUMC3=0.
LN 0095 SUMC4=0.
LN 0096 SUMC5=0.
LN 0097 SUMC6=0.
LN 0098 SUMQ1=0.
LN 0099 SUMQ2=0.
LN 0100 SUMQ3=0.
LN 0101 SUMQ4=0.
LN 0102 SUMQ5=0.
LN 0103 SUMQ6=0.
LN 0104 DO 460 I=1,NBP1
LN 0105 SUMC1=SUMC1+C(I,1)
LN 0106 SUMC2=SUMC2+C(I,2)
LN 0107 SUMC3=SUMC3+C(I,3)
LN 0108 SUMC4=SUMC4+C(I,4)

```

```

LN 0109      SUMC5=SUMC5+C(I,5)
LN 0110      SUMC6=SUMC6+C(I,6)
LN 0111      SUMQ1=SUMQ1+Q(I,1)
LN 0112      SUMQ2=SUMQ2+Q(I,2)
LN 0113      SUMQ3=SUMQ3+Q(I,3)
LN 0114      SUMQ4=SUMQ4+Q(I,4)
LN 0115      SUMQ5=SUMQ5+Q(I,5)
LN 0116      SUMQ6=SUMQ6+Q(I,6)
LN 0117      460  CONTINUE
LN 0118      SUMC=SUMC1
LN 0119      SUMQ=SUMQ1
LN 0120      WRITE(6,560) SUM(1),SUM(2),SUM(3),SUM(4),SUMC,SUMQ
LN 0121      SUMC=SUMC2
LN 0122      SUMQ=SUMQ2
LN 0123      WRITE(6,560) SUM(1),SUM(2),SUM(3),SUM(4),SUMC,SUMQ
LN 0124      SUMC=SUMC3
LN 0125      SUMQ=SUMQ3
LN 0126      WRITE(6,560) SUM(1),SUM(2),SUM(3),SUM(4),SUMC,SUMQ
LN 0127      SUMC=SUMC4
LN 0128      SUMQ=SUMQ4
LN 0129      WRITE(6,560) SUM(1),SUM(2),SUM(3),SUM(4),SUMC,SUMQ
LN 0130      SUMC=SUMC5
LN 0131      SUMQ=SUMQ5
LN 0132      WRITE(6,560) SUM(1),SUM(2),SUM(3),SUM(4),SUMC,SUMQ
LN 0133      SUMC=SUMC6
LN 0134      SUMQ=SUMQ6
LN 0135      WRITE(6,560) SUM(1),SUM(2),SUM(3),SUM(4),SUMC,SUMQ
LN 0136      560  FORMAT(/ 4H SUM,13X,6E12.3)
LN 0137      847  FAC2=1./(1.-.05/1.95)**2
LN 0138      450  CONTINUE
LN 0139      DO 203 I=1,NBP1
LN 0140      AI=I-1
LN 0141      AL=NB
LN 0142      P(I,1)=P(I,1)*FAC2
LN 0143      P(I,2)=P(I,2)*FAC2
LN 0144      P(I,3)=P(I,3)*FAC2
LN 0145      P(I,4)=P(I,4)*FAC2
LN 0146      DO 942 JA=1,6
LN 0147      C(I,JA)=C(I,JA)*FAC2
LN 0148      Q(I,JA)=Q(I,JA)*FAC2
LN 0149      942  CONTINUE
LN 0150      IF(I.GT. NB/ 6) GO TO 203
LN 0151      B=SIN(3.1416*(AI  )/AL)
LN 0152      A=COS(3.1416*(AI  )/AL)
LN 0153      IF(I.EQ.1) B=SIN(3.1416/(4.*AL))
LN 0154      IF(I.EQ.1) A=COS(3.1416/(4.*AL))
LN 0155      Z1=(.95*(1.-A)*(1.-.95*A)+(.95*B)**2)**2+ (.95*B*.05)**2
LN 0156      Z2=(1.-.95*A)**2+ (.95*B)**2
LN 0157      P(I,1)=P(I,1)*Z2*Z2/Z1 /FAC2
LN 0158      P(I,2)=P(I,2)*Z2*Z2/Z1 /FAC2
LN 0159      P(I,3)=P(I,3)*Z2*Z2/Z1 /FAC2
LN 0160      P(I,4)=P(I,4)*Z2*Z2/Z1 /FAC2
LN 0161      ZP5=Z2*Z2/Z1 /FAC2
LN 0162      DO 203 JA=1,6

```

```
LN 0163      C(I,JA)=C(I,JA)*ZP5
LN 0164      Q(I,JA)=Q(I,JA)*ZP5
LN 0165      203 CONTINUE
LN 0166      DO 280 ICH=1,NCH
LN 0167      SUM(ICH)=0.
LN 0168      DO 280 I=1,NBP1
LN 0169      280 SUM(ICH)=SUM(ICH)+P(I,ICH)
LN 0170      WRITE(LP,560) (SUM(ICH),ICH=1,NCH)
LN 0171      GO TO 115
LN 0172      END
```

USASI FORTRAN DIAGNOSTIC RESULTS FOR FSCRAN

NO ERRORS

UNREFERENCED STATEMENT LABELS

00150    00160    00165    00180    00190  
00360    00847

```
LN 0001      SUBROUTINE AVVAR(X,N,AV,VAR)
LN 0002      DIMENSION X(2)
LN 0003      AV=0.
LN 0004      DO 10 I=1,N
LN 0005      10  AV=AV+X(I)
LN 0006      AV=AV/FLOAT(N)
LN 0007      VAR=0.
LN 0008      DO 20 I=1,N
LN 0009      XR=X(I)-AV
LN 0010      20  VAR=VAR+XR*XR
LN 0011      VAR=VAR/FLOAT(N)
LN 0012      RETURN
LN 0013      END
```

USASI FORTRAN DIAGNOSTIC RESULTS FOR AVVAR

NO ERRORS

```
LN 0001      SUBROUTINE TAPER(X,N)
LN 0002      DIMENSION X(2)
LN 0003      PI=3.1415926
LN 0004      DO 50 I=1,N
LN 0005      ANG=(FLOAT(I)-.5)*PI*10./FLOAT(N)
LN 0006      IF (ANG.GE.PI) GO TO 60
LN 0007      FAC=.5-.5*COS(ANG)
LN 0008      X(I)=X(I)*FAC
LN 0009      IUP=N-I+1
LN 0010      50 X(IUP)=X(IUP)*FAC
LN 0011      60 RETURN
LN 0012      END
```

USASI FORTRAN DIAGNOSTIC RESULTS FOR TAPER

NO ERRORS

```
LN 0001      SUBROUTINE ZEFILL(X,NRCD,M,NAUG)
LN 0002      DIMENSION X(2)
LN 0003      DO 50 J=1,13
LN 0004      M=J
LN 0005      N=2**M
LN 0006      NAUG=2*N
LN 0007      IF (NAUG-NRCD)50,60,70
LN 0008      50 CONTINUE
LN 0009      STOP
LN 0010      60 RETURN
LN 0011      70 ISTART=NRCD+1
LN 0012      DO 80 I=ISTART,NAUG
LN 0013      80 X(I)=0.0
LN 0014      RETURN
LN 0015      END
```

USASI FORTRAN DIAGNOSTIC RESULTS FOR ZEFILL

NO ERRORS



```

LN 0001      SUBROUTINE RLFORT (C,M,S,IFS,IFERR)
LN 0002      DIMENSION C(1),S(1)
LN 0003      N = 2**M
LN 0004      IF (IABS(IFS).LT.2) CALL FORT (C,M,S,0,IFERR)
LN 0005      MCPLX = M - 1
LN 0006      IF (IFS) 10,45,30
LN 0007      10 CALL FORT (C,MCPLX,S,-2,IFERR)
LN 0008      CFACT = 1.
LN 0009      C(N+1) = 0.5*(C(1) - C(2))
LN 0010      C(N+2) = 0.
LN 0011      C(1) = 0.5*(C(1) + C(2))
LN 0012      C(2) = 0.
LN 0013      GO TO 50
LN 0014      30 CFACT = -1.
LN 0015      C(2) = C(1) - C(N+1)
LN 0016      C(1) = C(1) + C(N+1)
LN 0017      GO TO 50
LN 0018      40 CALL FORT (C,MCPLX,S,2,IFERR)
LN 0019      45 RETURN
LN 0020      50 K = N/2 - 1
LN 0021      MSIN = N/4
LN 0022      C(K+3) = -C(K+3)
LN 0023      DO 60 I=3,K,2
LN 0024      IS = (I-1)/2
LN 0025      IC = MSIN - IS
LN 0026      SI = S(IS)
LN 0027      CI = S(IC)*CFACT
LN 0028      A1 = C(I)
LN 0029      B1 = C(I+1)
LN 0030      L = N - I
LN 0031      A2 = C(L+2)
LN 0032      B2 = C(L+3)
LN 0033      C(I) = 0.5*(A1+A2+(B1+B2)*CT-(A1-A2)*ST)
LN 0034      C(I+1) = 0.5*(B1-B2-(B1+B2)*ST-(A1-A2)*CT)
LN 0035      C(L+2) = 0.5*(A1+A2-(B1+B2)*CT+(A1-A2)*ST)
LN 0036      60 C(L+3) = 0.5*(B2-B1-(B1+B2)*ST-(A1-A2)*CT)
LN 0037      IF (IFS+3) 90,70,90
LN 0038      70 C(1)=C(1)*2.
LN 0039      C(N+1)=C(N+1)*2.
LN 0040      DO 80 I=4,N,2
LN 0041      80 C(I)=-C(I)
LN 0042      90 CONTINUE
LN 0043      IF (IFS) 45,45,40
LN 0044      END

```

USASI FORTRAN DIAGNOSTIC RESULTS FOR RLFORT

NO ERRORS

```

LN 0001      SUBROUTINE FORT(A,M,S,IFS,IFERR)
LN 0002      DIMENSION A(1), S(1), K(14)
LN 0003      EQUIVALENCE (K(13),K1), (K(12),K2), (K(11),K3), (K(10),K4)
LN 0004      EQUIVALENCE (K( 9),K5), (K( 8),K6), (K(7),K7), (K( 6),K8)
LN 0005      EQUIVALENCE (K( 5),K9 ), (K( 4),K10), (K( 3),K11), (K( 2),K12)
LN 0006      EQUIVALENCE (K( 1),K13), ( K(1),N2)
LN 0007      IF (M)2,2,3
LN 0008      3 IF (M-13) 5,5,2
LN 0009      2 IFERR=1
LN 0010      1 RETURN
LN 0011      5 IFERR=0
LN 0012      N=2**M
LN 0013      IF ( IABS(IFS) - 1 ) 200,200,10
LN 0014      10 IF ( N-NP )20,20,12
LN 0015      12 IFERR=1
LN 0016      GO TO 200
LN 0017      20 K(1)=2*N
LN 0018      DO 22 L=2,M
LN 0019      22 K(L)=K(L-1)/2
LN 0020      DO 24 L=M,12
LN 0021      24 K(L+1)=2
LN 0022      IJ=2
LN 0023      DO 30 J1=2,K1,2
LN 0024      DO 30 J2=J1,K2,K1
LN 0025      DO 30 J3=J2,K3,K2
LN 0026      DO 30 J4=J3,K4,K3
LN 0027      DO 30 J5=J4,K5,K4
LN 0028      DO 30 J6=J5,K6,K5
LN 0029      DO 30 J7=J6,K7,K6
LN 0030      DO 30 J8=J7,K8,K7
LN 0031      DO 30 J9=J8,K9,K8
LN 0032      DO 30 J10=J9,K10,K9
LN 0033      DO 30 J11=J10,K11,K10
LN 0034      DO 30 J12=J11,K12,K11
LN 0035      DO 30 JI=J12,K13,K12
LN 0036      IF (IJ-JI)28,30,30
LN 0037      28 T=A(IJ-1 )
LN 0038      A(IJ-1)=A(JI-1)
LN 0039      A(JI-1)=T
LN 0040      T=A(IJ)
LN 0041      A(IJ)=A(JI)
LN 0042      A(JI)=T
LN 0043      30 IJ=IJ+2
LN 0044      IF (IFS)32,2,36
LN 0045      32 FN = N
LN 0046      DO 34 I=1,N
LN 0047      A(2*I-1) = A(2*I-1)/FN
LN 0048      34 A(2*I)=-A(2*I)/FN
LN 0049      36 DO 40 I=1,N,2
LN 0050      T = A(2*I-1)
LN 0051      A(2*I-1) =T + A(2*I+1)
LN 0052      A(2*I+1)=T-A(2*I+1)
LN 0053      T=A(2*I)
LN 0054      A(2*I) = T + A(2*I+2)

```

```

LN 0055      40 A(2*I+2)= T - A(2*I+2)
LN 0056      IF (M-1) 2,1 ,50
LN 0057      C   SET FOR L=2
LN 0058      50 LEXP1=2
LN 0059      C   LEXP1=2**(L-1)
LN 0060      LEXP=8
LN 0061      C   LEXP=2**(L+1)
LN 0062      NPL= 2**MT
LN 0063      C   NPL = NP* 2**-L
LN 0064      60 DO 130 L=2,M
LN 0065      DO 80 I=2,N2,LEXP
LN 0066      I1=I + LEXP1
LN 0067      I2=I1+ LEXP1
LN 0068      I3 =I2+LEXP1
LN 0069      T=A(I-1)
LN 0070      A(I-1) = T +A(I2-1)
LN 0071      A(I2-1) = T-A(I2-1)
LN 0072      T =A(I)
LN 0073      A(I) = T+A(I2)
LN 0074      A(I2) = T-A(I2)
LN 0075      T= -A(I3)
LN 0076      T1 = A(I3-1)
LN 0077      A(I3-1) = A(I1-1) - T
LN 0078      A(I3 ) = A(I1 ) - T1
LN 0079      A(I1-1) = A(I1-1) +T
LN 0080      80 A(I1) = A(I1 ) +T1
LN 0081      IF(L-2) 120,120,90
LN 0082      90 KLAST=N2-LEXP
LN 0083      JJ=NPL
LN 0084      DO 110 J=4,LEXP1,2
LN 0085      NPJJ=NT-JJ
LN 0086      UR=S(NPJJ)
LN 0087      UI=S(JJ)
LN 0088      ILAST=J+KLAST
LN 0089      DO 100 I= J,ILAST,LEXP
LN 0090      I1=I+LEXP1
LN 0091      I2=I1+LEXP1
LN 0092      I3=I2+LEXP1
LN 0093      T=A(I2-1)*UR-A(I2)*UI
LN 0094      T1=A(I2-1)*UI+A(I2)*UR
LN 0095      A(I2-1)=A(I-1)-T
LN 0096      A(I2 )=A(I ) - T1
LN 0097      A(I-1) =A(I-1)+T
LN 0098      A(I) =A(I)+T1
LN 0099      T=-A(I3-1)*UI-A(I3)*UR
LN 0100      T1=A(I3-1)*UR-A(I3)*UI
LN 0101      A(I3-1)=A(I1-1)-T
LN 0102      A(I3) =A(I1 )-T1
LN 0103      A(I1-1)=A(I1-1)+T
LN 0104      100 A(I1) =A(I1) +T1
LN 0105      110 JJ=JJ+NPL
LN 0106      120 LEXP1=2*LEXP1
LN 0107      LEXP = 2*LEXP
LN 0108      130 NPL=NPL/2

```

```

LN 0109      140 IF (IFS)145,2,1
LN 0110      145 DO 150 I=1,N
LN 0111      150 A(2*I) =-A(2*I)
LN 0112      160 GO TO 1
LN 0113      200 NP=N
LN 0114              MP=M
LN 0115              NI=N/4
LN 0116              MI=M-2
LN 0117              IF (MT) 260,260,205
LN 0118      205 THETA=.7853981634
LN 0119      C   THETA=PI/2**(L+1)   FOR L=1
LN 0120      210 JSTEP = NT
LN 0121      C   JSTEP = 2**( MT-L+1 ) FOR L=1
LN 0122              JDIF = NT/2
LN 0123      C   JDIF = 2**(MT-L) FOR L=1
LN 0124              S(JDIF) = SIN(THETA)
LN 0125              IF (MT-2)260,220,220
LN 0126      220 DO 250 L=2,MT
LN 0127              THETA = THETA/2.
LN 0128              JSTEP2 = JSTEP
LN 0129              JSTEP = JDIF
LN 0130              JDIF = JDIF/2
LN 0131              S(JDIF)=SIN(THETA)
LN 0132              JC1=NT-JDIF
LN 0133              S(JC1)=COS(THETA)
LN 0134              JLAST=NT-JSTEP2
LN 0135              IF (JLAST-JSTEP)250,230,230
LN 0136      230 DO 240 J=JSTEP,JLAST,JSTEP
LN 0137              JC=NT-J
LN 0138              J0=J+JDIF
LN 0139      240 S(JD)=S(J)*S(JC1)+S(JDIF)*S(JC)
LN 0140      250 CONTINUE
LN 0141      260 IF (IFS)20,1,20
LN 0142      END

```

USASI FORTRAN DIAGNOSTIC RESULTS FOR FORT

NO ERRORS

UNREFERENCED STATEMENT LABELS

00060 00140 00160 00210

```
LN 0001      SUBROUTINE PSPEC(A,N,P,M)
LN 0002      DIMENSION A(2),P(2)
LN 0003      REAL L
LN 0004      FN=N
LN 0005      FM=M
LN 0006      W=FN/FM
LN 0007      MP1=M+1
LN 0008      DO 100 INDL=1,MP1
LN 0009      L=INDL-1
LN 0010      XA=AMAX1(0.0,W*(L-0.5))
LN 0011      XB=AMIN1(FN,W*(L+0.5))
LN 0012      IXA=XA
LN 0013      IXA=MAX0(0,IXA)
LN 0014      IXAIND=IXA+1
LN 0015      IXB=XB+1.0
LN 0016      IXB=MIN0(IXB,N)
LN 0017      IXBIND=IXB+1
LN 0018      FINT=0.0
LN 0019      DO 50 IXIND=IXAIND,IXBIND
LN 0020      IX=IXIND-1
LN 0021      R=A(2*IXIND-1)
LN 0022      S=A(2*IXIND)
LN 0023      AMP=R*R+S*S
LN 0024      IF (IX.LE.IXA+1 .OR. IX.GE.IXB-1) GO TO 40
LN 0025      FINT=FINT+AMP
LN 0026      GO TO 50
LN 0027      40. XAA=AMAX1(XA,FLOAT(IX)-0.5)
LN 0028      XBB=AMIN1(XB,FLOAT(IX)+0.5)
LN 0029      DELX=AMAX1(0.0,XBB-XAA)
LN 0030      FINT=FINT+AMP*DELX
LN 0031      50 CONTINUE
LN 0032      P(INDL)=FINT*0.5
LN 0033      100 CONTINUE
LN 0034      RETURN
LN 0035      END
```

USASI FORTRAN DIAGNOSTIC RESULTS FOR PSPEC

NO ERRORS

```

LN 0001      SUBROUTINE XSPEC(A,B,N,C,Q,M)
LN 0002      REAL L
LN 0003      DIMENSION A(2),B(2),C(2),Q(2)
LN 0004      FN=N
LN 0005      FM=M
LN 0006      W=FN/FM
LN 0007      MP1=M+1
LN 0008      DO 100 INDL=1,MP1
LN 0009      L=INDL-1
LN 0010      XA=AMAX1(0.0,W*(L-0.5))
LN 0011      XB=AMIN1(FN,W*(L+0.5))
LN 0012      IXA=XA
LN 0013      IXA=MAX0(0,IXA)
LN 0014      IXAIND=IXA+1
LN 0015      IXB=XB+1.0
LN 0016      IXB=MIN0(IXB,N)
LN 0017      IXBIND=IXB+1
LN 0018      FINTC=0.0
LN 0019      FINTQ=0.0
LN 0020      DO 50 IXIND=IXAIND,IXBIND
LN 0021      IX=IXIND-1
LN 0022      RA=A(2*IXIND-1)
LN 0023      SA=A(2*IXIND)
LN 0024      RB=B(2*IXIND-1)
LN 0025      SB=B(2*IXIND)
LN 0026      AMPC=RA*RB+SA*SB
LN 0027      AMPQ=RA*SB-RB*SA
LN 0028      IF (IX.LE.IXA+1 .OR. IX.GE.IXB-1) GO TO 40
LN 0029      FINTC=FINTC+AMPC
LN 0030      FINTQ=FINTQ+AMPQ
LN 0031      GO TO 50
LN 0032      40  XAA=AMAX1(XA,FLOAT(IX)-0.5)
LN 0033      XBB=AMIN1(XB,FLOAT(IX)+0.5)
LN 0034      DELX=AMAX1(0.0,XBB-XAA)
LN 0035      FINTC=FINTC+AMPC*DELX
LN 0036      FINTQ=FINTQ+AMPQ*DELX
LN 0037      50  CONTINUE
LN 0038      C(INDL)=FINTC*.5
LN 0039      Q(INDL)=FINTQ*.5
LN 0040      100 CONTINUE
LN 0041      RETURN
LN 0042      END

```

USASI FORTRAN DIAGNOSTIC RESULTS FOR XSPEC

NO ERRORS

001.98 MIN

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

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

---

REFERENCES

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

```

001 PROGRAM OUTPUT
002 C
003 COMMON AM1,AM2,LAG ,FREPAR
004 COMMON BAUG,NPTS
005 COMMON C(1026,4),P(101,4),Q(101,6),R(101,6),PMM(101,2),PPP(101,2)
006 COMMON PANING(101),ROT(101,2),SUMANG(101)
007 COMMON RQ(101) ,CO(1,1,6),PH(101,6)
008 INTEGER PLOTLN(100)
009 EQUIVALENCE(NB,LAG),(L1,NBP1)
010 DIMENSION SUM(4)
011 LP=6
012 NCH=4
013 IKK=0
014 REWIND 2
015 BUFFER IN(2,1)(XM1,Q(101,6))
016 IF(IFUNIT(2))1,801,1
801 CONTINUE
018 WRITE(6,701)XM1,AM2,LAG
701 FORMAT(1H1/////,17H MOORING NUMBERS ,A8,6H AND ,A8,///
$5H LAG=,14)
021 CALL POLAR(C(1,1),Q(1,1),R(1,1),P(1,2),LAG,CO(1,1),PH(1,1))
022 REWIND 2
023 WRITE(6,603)
024 WRITE(6,604)
800 CONTINUE
026 IKK=IKK+1
027 WRITE(6,601)
028 LI=LAG+1
029 DO 2 I=1,LI
030 AI=LAG
031 AI=I-1
032 FREQ=AI/AL*0.5*FREPAR
033 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))0,6,7
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)
LN 049 J=IKK
12 CO(I,J)=CO(I,J)*2
LN 051 CALL INFACTS(P(1,1),P(1,2),C(I,1),Q(I,1),AMAJ,AMIN,AP,AM,THETA,RC,PP
LN 052 IP,PM)
LN 053 IF(IKK.LT.3)PPP(I,IKK)=PP
LN 054 IF(IKK.LT.3)PMM(I,IKK)=PM

```



```

LN 0055      KANGLE=KANGE(KFIX(THETA))
LN 0056      ARC=ABS(RC)
LN 0057      ANGLE=KANGLE
LN 0058      IF (ARC.GT.0.9) ANGLE=0.0
LN 0059      IF (IKK.EQ.2) SUMANG(I)=DANING(I)+ANGLE
LN 0060      IF (IKK.EQ.2) DANING(I)=DANING(I)-ANGLE
LN 0061      IF (IKK.EQ.1) DANING(I)= ANGLE
LN 0062      11  WRITE(6,602)
LN 0063      $      AI,FREQ,ALFREQ,PERIOD,P(I,1),ALPX,P(I,2),ALPY,C(I,1),Q(I,1)
LN 0064      SCU(I,IKK),PH(I,IKK),AMAJ,AMIN,RC,KANGLE
LN 0065      FRQ(I)=PERIOD
LN 0066      ROT(I,IKK)=RC
LN 0067      P(I,1)=ALPX
LN 0068      P(I,2)=ALPY
LN 0069      CU(I,IKK)=6.*CO(I,IKK)-3.
LN 0070      2    CONTINUE
LN 0071      WRITE(6,803)
LN 0072      803  FORMAT(1H1)
LN 0073      XMIN=-3
LN 0074      XMAX=3
LN 0075      C=   A PLOT IS WANTED
LN 0076      SCALFAC=(XMAX-XMIN)/100.
LN 0077      WRITE(61,23)
LN 0078      23  FORMAT(1H1)
LN 0079      WRITE(61,88)SCALFAC
LN 0080      88  FORMAT(40X,15HSCALE FACTOR = ,F8.2)
LN 0081      WRITE(61,82)XMIN,XMAX
LN 0082      82  FORMAT(11X,F6.2,97X,F5.2/)
LN 0083      WRITE(61,79)
LN 0084      79  FORMAT(14X,102(1H+))
LN 0085      AL = LAG
LN 0086      LI=LAG +1
LN 0087      DO 15 I=1,LI
LN 0088      AI=I-1
LN 0089      FREQ=AI/AL*.5*FREPAR
LN 0090      IF (AI)90,90,91
LN 0091      90  PERIOD=0.
LN 0092      GO TO 92
LN 0093      91  PERIOD=1./FREQ
LN 0094      92  CALL BLANKV(PLOTLN(1),100)
LN 0095      JK=(P(I,1)-XMIN)/SCALFAC+.5001
LN 0096      IF (JK.LE.100.AND.JK.GE.1) PLOTLN(JK)=1HX
LN 0097      IF (JK.GT.100)PLOTLN(100)=1H+
LN 0098      IF (JK.LT.1)PLOTLN(1)=1H-
LN 0099      JK=(P(I,2)-XMIN)/SCALFAC+.5001
LN 0100      IF (JK.LE.100.AND.JK.GE.1) PLOTLN(JK)=1HY
LN 0101      IF (JK.GT.100)PLOTLN(100)=1H+
LN 0102      IF (JK.LT.1)PLOTLN(1)=1H-
LN 0103      CA=CO(I,IKK)
LN 0104      JK=(CA -XMIN)/SCALFAC+.5001
LN 0105      IF (JK.LE.100.AND.JK.GE.1) PLOTLN(JK)=1HC
LN 0106      IF (JK.GT.100)PLOTLN(100)=1H+
LN 0107      IF (JK.LT.1) PLOTLN(1) = 1H-
LN 0108      WRITE(61,1000) PERIOD,PLOTLN

```

```

LN 0109 1000 FORMAT(2A,F7.3,5X,1H*,100A1,1H*)
LN 0110 15 CONTINUE
LN 0111 GO TO(20,39),IKK
LN 0112 20 CALL MOVE(P(1,3),P(1,1),101)
LN 0113 CALL MOVE(P(1,4),P(1,2),101)
LN 0114 CALL MOVE(C(1,2),C(1,1),101)
LN 0115 CALL MOVE(Q(1,2),Q(1,1),101)
LN 0116 CALL POLAR(C(1,2),Q(1,2),P(1,3),P(1,4),LAG,CO(1,2),PH(1,2))
LN 0117 WRITE(6,605)
LN 0118 WRITE(6,606)
LN 0119 GO TO 800
LN 0120 39 CONTINUE
LN 0121 DO 2482 I=1,L1
LN 0122 TA=SUMANG(I)
LN 0123 IF (TA.GT.180.) SUMANG(I)=TA-360.
LN 0124 2482 IF (TA.LT.-180.) SUMANG(I)=TA+360.
LN 0125 REWIND 2
LN 0126 BUFFER IN(2,1)(X=1,0(101,0))
LN 0127 IF (IFUNIT(2)) 1,33,1
LN 0128 33 REWIND 2
LN 0129 BUFFER OUT(2,1)(X=1,NPTS)
LN 0130 BUFFER OUT(2,1)(P(101,1),Q(101,6))
LN 0131 IF (IFUNIT(2)) 1,34,1
LN 0132 34 CONTINUE
LN 0133 BUFFER OUT(2,1) (PH(1,1),SUMANG(101))
LN 0134 J=IFUNIT(2)
LN 0135 BUFFER OUT(2,1)(FRQ(1) ,PH(101,6))
LN 0136 IF (IFUNIT(2)) 1,32,1
LN 0137 32 ENDFILE 2
LN 0138 REWIND 2
LN 0139 1 STOP
LN 0140 601 FORMAT(1H1,1X,13AHLAG FREQ LNFREQ PERIOD POWER(U) LNPOWU PO
LN 0141 $WER(V) LNPOWV COSPECTRUM QUADSPEC COHERNC PHASE MAJ M
LN 0142 $IN ROT.COEFF. T )
LN 0143 602 FORMAT(1H ,1X,F4.0,1X,3(E9.2,1X),E 10.2,1X,F6.2,1X,E10.2,1X,F6.2,1
LN 0144 $X,2(E9.2,1X),1X,F7.3,1X,F7.2,3(E9.2),1X,I3)
LN 0145 603 FORMAT(1H) //////////////,25X,3H U )
LN 0146 605 FORMAT(1H) //////////////,25X,3H VP )
LN 0147 604 FORMAT(1H+,30X,3H V )
LN 0148 606 FORMAT(1H+,30X,3H VP )
LN 0149 607 FORMAT(1H1 ////////////// 25X,10H U UP )
LN 0150 608 FORMAT(1H1 ////////////// 25X,10H V VP )
LN 0151 END

```

USASI FORTRAN DIAGNOSTIC RESULTS FOR OUTPUT

NO ERRORS

```
LN 0001      SUBROUTINE POLAR(C,Q,X,Y,L,COHERE,PHASE)
LN 0002      DIMENSION C(2),Q(2),X(2),Y(2),COHERE(2),PHASE(2)
LN 0003      LP1=L+1
LN 0004      PI=3.1415926536
LN 0005      EPS=1.E-15
LN 0006      XYMAX=0.
LN 0007      DO 20 K=1,LP1
LN 0008      XY=X(K)*Y(K)
LN 0009      IF(XY.GT.XYMAX) XYMAX=XY
LN 0010      20  CONTINUE
LN 0011      DO 40 K=1,LP1
LN 0012      XY=X(K)*Y(K)
LN 0013      IF(XY.LE.XYMAX*EPS) GO TO 30
LN 0014      COHERK=SQRT((C(K)*C(K)+Q(K)*Q(K))/XY)
LN 0015      IF((Q(K)**2+C(K)**2).EQ.0.)GO TO 41
LN 0016      PHASE(K)=ATAN2(Q(K),C(K))*180./PI
LN 0017      GO TO 42
LN 0018      41  PHASE(K)=0.
LN 0019      42  CONTINUE
LN 0020      COHERE(K)=COHERK
LN 0021      GO TO 40
LN 0022      30  COHERE(K)=0.
LN 0023      PHASE(K)=0.
LN 0024      40  CONTINUE
LN 0025      RETURN
LN 0026      END
```

USASI FORTRAN DIAGNOSTIC RESULTS FOR POLAR

NO ERRORS

```
LN 0001      FUNCTION KANGE(N)  
LN 0002      5  IF (N.LT.360)GO TO 11  
LN 0003      GO TO 2  
LN 0004      11 IF (N.GE.0)GO TO 1  
LN 0005      GO TO 3  
LN 0006      1  KANGE=N  
LN 0007      RETURN  
LN 0008      2  N=N-360  
LN 0009      GO TO 5  
LN 0010      3  N=N+360  
LN 0011      GO TO 5  
LN 0012      END
```

USASI FORTRAN DIAGNOSTIC RESULTS FOR KANGE

NO ERRORS

```
LN 0001      FUNCTION KFIX(X)
LN 0002      CONV=ATAN(1.)/45.
LN 0003      X=X/CONV
LN 0004      KFIX=IFIX(SGN(X)*(ABS(X)+.5))
LN 0005      RETURN
LN 0006      END
```

USASI FORTRAN DIAGNOSTIC RESULTS FOR KFIX

NO ERRORS

```
LN 0001 FUNCTION SGN(X)  
LN 0002 IF (X.EQ.0.)GO TO 2  
LN 0003 SGN=X/ABS(X)  
LN 0004 1 RETURN  
LN 0005 2 SGN=0.  
LN 0006 GO TO 1  
LN 0007 END
```

USASI FORTRAN DIAGNOSTIC RESULTS FOR SGN

NO ERRORS

```
LN 0001      SUBROUTINE NFACTS(PU,PV,PUV,QUV,MAJ,MIN,AP,AM,THETA,RC,PP,PM)
LN 0002      REAL MAJ,MIN
LN 0003      RC=-2.*QUV/(PU+PV)
LN 0004      PP=.25 * (2.*PU+2.*PV+4.*QUV)
LN 0005      PM=.25* (2.*PU+2.*PV-4.*QUV)
LN 0006      AP=SQRT(PP)
LN 0007      AM=SQRT(PM)
LN 0008      THETA=.5*ATAN2(2.*PUV,PU-PV)
LN 0009      MAJ=ABS(AP+AM)
LN 0010      MIN=ABS(AP-AM)
LN 0011      RETURN
LN 0012      END
```

USASI FORTRAN DIAGNOSTIC RESULTS FOR NFACTS

NO ERRORS

E 000.98 MIN

SE.47

**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

<p><b>INPUT</b></p> <p>See above</p>	<p><b>OUTPUT</b></p> <p>Printed output of correlation parameters by frequency band.</p>
<p><b>SUBROUTINES USED</b></p> <p>COspect</p>	<p><b>CALLED BY</b></p> <p>Data1</p>
<p><b>STORAGE REQUIREMENTS</b></p>	<p><b>SOURCE LANGUAGE</b></p>
<p><b>MACHINE</b></p>	<p><b>PROGRAMMER</b></p>
<p><b>REFERENCES</b></p>	



```

LN 0001 PROGRAM SPECPLI
LN 0002 COMMON XM1,XM2,LAG ,FREPAR
LN 0003 COMMON NAUG,NPTS
LN 0004 COMMON P(101,4),C(101,6),Q(101,6),PMM(101,2),PPP(101,2)
LN 0005 COMMON DANING(101),ROT(101,2) ,SUMANG(101)
LN 0006 COMMON FRQ(101)
LN 0007 INTEGER PLOTLN(102)
LN 0008 KE=IND 2
LN 0009 BUFFER IN(2,1)(XM1,NPTS)
LN 0010 BUFFER IN(2,1)(P(101,1),C(101,6))
LN 0011 BUFFER IN(2,1)(PMM(1,1),SUMANG(101))
LN 0012 IF(IFUNIT(2))1,2,1
LN 0013 1 WRITE(6,000)
LN 0014 000 FORMAT(1H1,6HERROR )
LN 0015 3 STOP
LN 0016 2 CONTINUE
LN 0017 WRITE(6,701)XM1,XM2,LAG
LN 0018 701 FORMAT(1H1//////,17H MOORING NUMBERS ,A8.6H AND ,A8.7//
LN 0019 $5M LAG=,I4)
LN 0020 LI=LAG+1
LN 0021 CALL ZEROV(FRQ,101)
LN 0022 DO 5 I=1,LI
LN 0023 IF(I.EQ.1)GO TO 123
LN 0024 A=FLOAT(I-1)/FLOAT(LAG)*.5*FREPAR
LN 0025 FRQ(I)=I./A
LN 0026 123 CONTINUE
LN 0027 CALL GUSPEC(C(I,3),C(I,6),Q(I,5),Q(I,4),C(I,4),C(I,5),Q(I,6),
LN 0028 $Q(I,3),TPP,PM,DT,PPP(I,1),PMM(I,1),PPP(I,2),PMM(I,2),DANING(I),
LN 0029 IR(I,1),P(I,2),P(I,3),P(I,4),ROT(I,1),ROT(I,2),SUMANG(I))
LN 0030 K=I-1
LN 0031 WRITE(6,700)K,FRQ(I)
LN 0032 700 FORMAT(1H+,I3,2X,+6.2)
LN 0033 5 CONTINUE
LN 0034 5501 FORMAT(1H1)
LN 0035 WRITE(6,5501)
LN 0036 WRITE(6,118)
LN 0037 118 FORMAT(1X ,4H3AN),1X,6HPERIOD,2X,3H0.0,21X,3H0.5,21X,3H1.0,1X,4H-
LN 0038 180,20X,3H0.0,20X,3H180)
LN 0039 DO 15 I=1,LI
LN 0040 CALL BLANKV(PLOTLN(1),102)
LN 0041 JK=P(I,1)*50.+5001
LN 0042 IF(JK.LE.50.AND. JK.GE.1)PLOTLN(JK)=1H+
LN 0043 JK=P(I,3)*50.+5001
LN 0044 IF(JK.LE.50.AND. JK.GE.1)PLOTLN(JK)=1H-
LN 0045 PLOTLN(51)=1H*
LN 0046 JK= P(I,2)*25./180.+77.5001
LN 0047 PLOTLN(77)=1H.
LN 0048 IF(JK.LE.102.AND. JK.GE.51)PLOTLN(JK)=1H+
LN 0049 JK=P(I,4)*25./180.+77.5
LN 0050 IF(JK.LE.102.AND. JK.GE.51)PLOTLN(JK)=1H-
LN 0051 WRITE(6,1000)I,FR Q(I),PLOTLN
LN 0052 1000 FORMAT(1X,I4,2X,F7.1,1X,1H*,102A1,1H*)
LN 0053 15 CONTINUE
LN 0054 WRITE(6,118)

```

```

WRITE(6,5501)
WRITE(6,119)
111 FORMAT (1X,4HEND,1X,8HPEN100,2X,4H-180,18X,4H-90.,23X,3H0.0,21X,3H909,21X
190.,21X,4H180.)
DO 16 I=1,L1
CALL BLANKY(PLOTLEN(I),100)
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,PPM(I),PLOTLN
16 CONTINUE
WRITE(6,119)
REWIND 2
GO TO 3
END

```

PROGRAM DIAGNOSTIC RESULTS FOR SPECPLT

0 ERRORS

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

OUTPUT

SUBROUTINES USED

CALLED BY  
Specplt

STORAGE REQUIREMENTS

SOURCE LANGUAGE

MACHINE

PROGRAMMER

REFERENCES

```

1 SUBROUTINE COSPEC (PUU,PVV,QUIV,QUV1,PUV1,PUIV,QVV,QUU,TPR,TMM,DT,P
2 16,P5,P4,P3,D1,C2,TP,C1,IM,R1,R2,SA)
3 S1=1
4 C=180./3.1415926
5 IF (R1/R2.LT.0.) GO TO 27
6 IF (R1.GT.0..AND.R2.GT.0.) S1=-1.
7 PXPXP=.5*(PUU+PVV-QUIV+QUV1)
8 QXPXP=.5*(PUV1-PUIV-QVV-QUU)
9 QXMXM=.5*(PUV1-PUIV+QVV+QUU)
0 PXMXM=.5*(PUU+PVV-QUV1+QUIV)
1 TPP=-ATAN2(QXPXP,PXPXP)
2 TMM= ATAN2(-QXMXM,PXMXM)*(-1.)
3 QZ=QXPXP
4 Q1=QXMXM
5 P1=PXMXM
6 PZ=PXPXP
7
8 C1=(P1*P1+Q1*Q1)/(P3*P5)
9 C2=(P2*P2+Q2*Q2)/(P4*P6)
0
1 ANG=-.5*ATAN2(Q2+P1-Q1*PZ,P1*P2+Q2*Q1)*C
2 TM=C*TMM
3 TP=C*TPP
4 25 TL=(TP-D1)
5 TU=(D1+TM)
6
7 27 PXPXP=.5*(PUU-PV1-QUV1-QUIV)
8 QXPXP=.5*(PUV1+PUIV+QUU-QVV)
9 QXMXM=.5*(PUV1+PUIV-QUU+QVV)
0 PXMXM=.5*(PUU-PVV+QUV1+QUIV)
1 C2=(PXPXP**2+QXPXP**2)/(P6*P3)
2 C1=(PXMXM**2+QXMXM**2)/(P4*P5)
3 IF (R1.GT.0..AND.R2.GT.0.) S1=-1
4 ANG=.5*ATAN2(QXPXP*PXMXM-QXMXM*PXPXP,PXPXP*PXMXM+QXPXP*QXMXM)*C
5 TMM= ATAN2(QXMXM,PXMXM) *(-1.)
6 TPP= ATAN2(QXPXP,PXPXP)
7 TM=C*TMM
8 TP=C*TPP
9 TL=TP-SA
0 T0=SA+TM
1 15 CONTINUE
2 IF (ANG.GT.0.) ANG2=ANG-180.
3 IF (ANG.LT.0.) ANG2=ANG+180.
4 P3=ANG
5 P4=ANG2
6 P6=TL
7 P5=TU
8 IF (TL.GT.180.) P6=TL-360.
9 IF (TL.LT.-180.) P4=TL+360.
0 IF (T0.GT.180.) P5=T0-360.
1 IF (T0.LT.-180.) P5=T0+360.
2 WRITE(6,600)C2,TP,C1,IM,P0,P5,ANG2,ANG
3 600 FORMAT (1H,12X
4 2 ,4HC+*=?,F5.2,2X,8HTHETA+*=?,F0.1,2X,4HC--=?,F5.2,2X,8HTHET

```

SI FORTRAN(1.1)/MSOS 482. INTEGER WORD SIZE = 1

PSR 287

DATE

11/15/73

```

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

```

USASI FORTRAN DIAGNOSTIC RESULTS FOR COSPEC

NO ERRORS

REFERENCED STATEMENT LABELS

0.71 MIN

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**

X raw data form two vector time series.

**OUTPUT**

Interpolated data series

**SUBROUTINES USED**

Speak

**CALLED BY**

**STORAGE REQUIREMENTS**

**SOURCE LANGUAGE**

**MACHINE**

**PROGRAMMER**

**REFERENCES**



SI FORTRAN(1.1)/MSOS 4.2 INTEGER WORD SIZE = 1

PSR 287

DATE 10/29,

5 509 FORMAT(1X,2I5,F10.3)  
6 STOP  
7 END

USASI FORTRAN DIAGNOSTIC RESULTS FOR DATA1

NO ERRORS



```

01 SUBROUTINE SPEAK(NUM)
02 COMMON XM1, XM2, LAG, FREFAK
03 COMMON NAUG, NPTS
04 COMMON X(1026,4), P(101,4)
05 COMMON C(101,6), Q(101,6), CO(101,2), PH(101)
06 NPTS=NUM
07 NAUG=0
08 CALL ZEROV(P,1919)
09 REWIND 2
10 BUFFER OUT(2,1)(XM1,PH(101))
11 IF(IFUNIT(2))1,2,1
12 2 CONTINUE
13 ENDFILE 2
14 REWIND 2
15 BUFFER IN(2,1)(XM1,PH(101))
16 KANG=IFUNIT(2)
17 1 REWIND 2
18 RETURN
19 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

NAME Salcate

**PURPOSE**

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

**DESCRIPTION**

**USAGE**

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

**INPUT**

X vector contains a scalar and vector time series

**OUTPUT**

**SUBROUTINES USED**

**CALLED BY**

S Data 1

**STORAGE REQUIREMENTS**

**SOURCE LANGUAGE**

**MACHINE**

**PROGRAMMER**

**REFERENCES**

PROGRAM SALCATE

SCALAR - VECTOR TIME SERIES CORRELATIONS

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

CONTINUE

WRITE(6,753)NPTS

FORMAT(1H1,15)

WRITE(6,701)XM1, XM2, LAG

FORMAT(1H1//////,17H MOORING NUMBERS ,A8,6H AND ,A8,///

5H LAG=,14)

CALL SPECTR(NPTS)

REWIND 2

STOP

END

FORTRAN DIAGNOSTIC RESULTS FOR SALCATE

NO ERRORS

```
SUBROUTINE SPECTR(NUM)
COMMON XM1, XM2, LAG, FREPAR
COMMON NAUG, NPTS
COMMON X(1026,3), P(101,3)
COMMON C(101,3), CU(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, CU, 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

NO ERRORS

UNDEFINED STATEMENT LABELS

```

SUBROUTINE FSCRA (X,NPTS,NB,P,COH,PH,ISKIP,NCH,C,Q)
COMMON XXX,XXY,LIL ,FREP
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
NB=PI=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,8HX-COMP,
+2E11.3,5X,8HY-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))
WRITE(LP,150)
150 FORMAT (// 30H MEAN REMOVED FROM EACH SERIES )
160 CONTINUE
DO 170 ICH=1,NCH
170 CALL TAPER(X(1,ICH),NPTS)
WRITE(LP,180)
180 FORMAT (// 44H CO LINE 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 240
WRITE (LP,210) NPTS, NAUG
210 FORMAT ( / 36H EACH SERIES AUGMENTED BY ZEROS FROM 15.8H PTS TO
2 15.4H 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)
7TL=FLOAT(NAUG)/FLOAT(NPTS)
DO 2825 I=1,NBP1
DO 2825 J=1,3
C(I,J)=C(I,J)*7TL
Q(I,J)=Q(I,J)*7TL
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)

```

```

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
R=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*A*.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

```

FORTRAN DIAGNOSTIC RESULTS FOR FSCRAN

NO ERRORS

STATEMENT LABELS:

160      00165      00180      00190

347



SUBROUTINE AVVAR(X,N,AV,VAR)

DIMENSION X(2)

AV=0.

DO 10 I=1,N

AV=AV+X(I)

AV=AV/FLOAT(N)

VAR=0.

DO 20 I=1,N

XR=X(I)-AV

VAR=VAR+XR\*XR

VAR=VAR/FLOAT(N)

RETURN

END

FORTRAN DIAGNOSTIC RESULTS FOR AVVAR

NO ERRORS

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)

  Y(I)=X(I)\*FAC

  TOP=N-I+1

50  X(IUP)=X(IUP)\*FAC

60  R=TURN

  END

1 FORTRAN DIAGNOSTIC RESULTS FOR TAPER

2 ERRORS

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

CONTINUE

STOP

RETURN

ISTART=NRCD+1

DO 80 I=ISTART,NAUG

X(I)=0.0

RETURN

END

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
DO 60 I=3,K,2
IS = (I-1)/2
IC = MSIN - IS
SI = 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) 45,45,45
END

```

FORTRAN DIAGNOSTIC RESULTS FOR RLFORT

NO ERRORS

SUBROUTINE FORT( ,M,S,IFS,IFERR)	FORT 004
DIMENSION A(1), S(1), K(14)	FORT 051
EQUIVALENCE (K(1),K1),(K(2),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
DO 40 I=1,N,2	FORT 105
T = A(2*I-1)	FORT 107
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(I2+J+2) = T - A(I1+2)	FORT 111
	IF(M-1) 2,1 .50	FORT 112
C	S=T 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
	T1=I + LEXP1	FORT 123
	T2=I1+ LEXP1	FORT 124
	T3 =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
	T1 = A(I3-1)	FORT 133
	A(I3-1) = A(T1-1) - T	FORT 134
	A(I3 ) = A(T1 ) - T1	FORT 135
	A(I1-1) = A(I1-1) +T	FORT 136
80	A(I1) = A(I1) +T1	FORT 137
	IF(L-2) 120,120,00	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(I1)*UI	FORT 150
	T1=A(I2-1)*UI+A(I2)*UR	FORT 151
	A(I2-1)=A(I-1)-T	FORT 152
	A(I2 )=A(I ) - T1	FORT 153
	A(I-1) =A(I-1)+T	FORT 154
	A(I) =A(I)+T1	FORT 155
	T=-A(I3-1)*UI-A(I3)*UR	FORT 156
	T1=A(I3-1)*UR-A(I3)*UI	FORT 157
	A(I3-1)=A(I1-1)-	FORT 158
	A(I3) =A(I1) -T1	FORT 159
	A(I1-1)=A(I1-1)+T	FORT 160
100	A(I1) =A(I1) +T1	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

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)200,230,230	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

FORTAN DIAGNOSTIC RESULTS FOR FORT

NO ERRORS

FOUND STATEMENT LABELS

140 00160 00210

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

I=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)

XAB=AMIN1(XB,FLOAT(IX)+0.5)

DELX=AMAX1(0.0,XAB-XAA)

FINT=FINT+AMP\*DELX

CONTINUE

P(INDL)=FINT\*0.5

CONTINUE

RETURN

END

FORTRAN DIAGNOSTIC RESULTS FOR PSPEC

NO ERRORS



```

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

```

1 FORTRAN DIAGNOSTIC RESULTS FOR XSPEC

NO ERRORS

MIN

PROGRAM SUTPUT

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))1,8,1,1

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

AT=I-1

FREQ=AT/AL\*0.5\*FREPAR

IF(AT)3,3.4

3 AFREQ=0.

PERIOD=0.

GO TO 5

4 PERIOD=1./FREQ

AFREQ=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,5),P(I,3),C(I,2),Q(I,2),AMAJ,AMIN,AP,AM,THETA,RC,P

1 P,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
FREQ(I)=PERIOD
P(I,2)=ALPA
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(14X,715HSCALE FACTOR = ,F8.2)
WRITE(61,82) XMIN,XMAX
42 FORMAT(11X,F6.2,47X,F5.2/)
WRITE(61,79)
79 FORMAT(14X,102(1H+))
AL = LAG
LI=LAG+1
DO 15 I=1,LI
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+

```

```

1000 IF (JK.LT.1) PLOTLN(1) = 1H-
15 WRITE(6,1000) PERIOD,PLOTLN
FORMAT(2X,F7.3,5X,1H*,100A1,1H*)
CONTINUE
WRITE(6,605)
WRITE(6,606)
39 CONTINUE
REWIND 2
BUFFER IN(2,1)(XMI,Q(101,3))
IF (IFUNIT(2))1,33,1
33 REWIND 2
BUFFER OUT(2,1)(XMI,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)
POWER(V) COSPECTRUM QUADSPEC COHERNC PHASE MAJ M
$IN ROT COEF T )
602 FORMAT(1H,1X,F4.0,1X,3(E6.2,1X), E10.2,1X,E10.2,1X,E10.2,
1X,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

```

FORTRAN DIAGNOSTIC RESULTS FOR SUTPUT

NO. ERRORS

CEB STATEMENT LABELS

039 00604 00607 00608

```

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
  COHFRK=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
  COHFRK(K)=COHERK
  GO TO 40
30 COHFRK(K)=0.
  PHASE(K)=0.
40 CONTINUE
RETURN
END

```

[ FORTRAN DIAGNOSTIC RESULTS FOR POLAR

NO ERRORS

```
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

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

FORTRAN DIAGNOSTIC RESULTS FOR KFIX

NO ERRORS

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

FORTRAN DIAGNOSTIC RESULTS FOR SGN

NO ERRORS



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

PROGRAM SSECPLT

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),TTP,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=1-1

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  
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=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,11\*.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,
190.,21X,4H180.)
DO 16 I=1,L1
CALL BLANKV(PLOTLN(1),102)
JK=PPP(I )*50./180.+50.5+1
PLOTLN(76)=JH.
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

SUBROUTINE COSPEC(PUU,QUV1,PUV1,QUU,TPP,TMM,DT,P6,P5,D1,C2,TP,C1,

TM,SA)

P7=C2\*.5

C=180./3.14159265

QXPXP=.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

C1=(P1\*P1+Q1\*Q1)/(P7\*P5)

C2=(P2\*P2+Q2\*Q2)/(P7\*P6)

ANG=-.5\*ATAN2(Q2\*P1-Q1\*P2,P1\*P2+Q2\*Q1)\*C

TM=C\*TMM

TP=C\*TPP

25 TL=(TP-D1)

T0=(D1+TM)

IF (ANG.GT.0.) ANG2=ANG-180.

IF (ANG.LT.0.) ANG2=ANG+180.

PHU=ANG

QUV1=ANG2

P6=TL

P5=T0

IF (TL.GT.180.) P6=TL-360.

IF (TL.LT.-180.) P6=TL+360.

IF (T0.GT.180.) P5=T0-360.

IF (T0.LT.-180.) P5=T0+360.

WRITE(6,600)C2,TP,C1,TM,P6,P5,ANG2,ANG

600 FORMAT(1H,11X

?,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

STATEMENT LABELS

```

PROGRAM SDATA1
SCALAR - VECTOR TIME SERIES CORRELATIONS
REAL MOOR
COMMON XM1,XM2,LAG ,FREPAR
COMMON NAUG,NPTS
COMMON X(1024,3)
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
T5=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.*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)
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

FORTRAN DIAGNOSTIC RESULTS FOR SPEAK

NO ERRORS

MIN

18165

