N.SMITH

NATIONAL INSTITUTE OF OCEANOGRAPHY

WORMLEY, GODALMING, SURREY

N.I.O. Computer Programs 9

N.I.O. Internal Report No. N 9

OCTOBER 1967

N.I.O. COMPUTER PROGRAMS 9

edited

by

B. J. Hinde

N.I.O. Internal Report No. N.9

National Institute of Oceanography

N.I.O. PROGRAMS 9

1) PROGRAMS

59/A	Thermometer corrections for deep-sea reversing thermometers
* 86	Bartlett's method for fitting an exponential functional relationship (95% confidence limits)
* 86/A	Bartlett's method for fitting an exponential functional relationship (70% confidence limits)
* 86/B	Bartlett's method for fitting an exponential functional relationship (Compares two slopes)
* 87	Gravity anomalies
* 88	Atlas Costs
* 89	Analysis of shipborne wave records
90	Tables of the Integral of Associated Legendre Polynomials
* 91	Matthews Revision (2, Gulf Stream)
* 92	General spectral analysis
93	Least squares polynomial fitting
* 94	Preparation of Biological Station File
* 94/A	Accessing Biological Station File
94/B	Initialising Biological Station File
94/C	Updating Master Biological Station File
95	Determination of position from Decca coordinates (BALLARIN) (Individual positions)
95/A	Determination of position from Decca coordinates (BALLARIN) (Table of positions)
* 96	Decoding Plessey Current Meter tapes
96/A	Decoding Plessey Current Meter tapes of temperature and depth
96/B	Decoding Bergen meter tapes of temperature, current speed and direction.

2) SUB-PROGRAMS

-1	Spectral analysis
-2	Spectral analysis with correction for pre-whitening
-3	Analysis of pitch, roll, heave buoy data
-4	Graph plotting
-5	General Spectral analysis

* These programs are available in compiled form on magnetic tape.

N.I.O. PROGRAM 59/A

Title Thermometer corrections for deep-sea reversing thermometers.

Language EMA

Machine ATLAS I

Purpose This program is identical in purpose and method to NIO 59 written in CHLF 3/4 for Mercury. One additional data terminator is required.

Inputs 0) Program.

1) Job description and data.

Program

```
COMPILER EMA
S002 NIO PROGRAM 59/A THERMOMETER CORRECTIONS 6/4/1967
                              (or date of appropriate version)
MAIN 250
AUXILIARY (0, 0)
DUMPS 0
```

Then follows the rest of the program, ending with

```
CLOSE
***Z
```

Job description and data

```
JOB
Job Number, Job Title
INPUT
0 S002 NIO PROGRAM 59/A THERMOMETER CORRECTIONS 6/4/1967
SELF = 1                       (or date of appropriate version)
OUTPUT
```

[Either tape or lineprinter output may be obtained. Up to 50 lines, or 1 block of paper tape, per thermometer will be required. Output 0 must be used]

```
STORE 20/70 BLOCKS
COMPUTING q INSTRUCTIONS where q = (1500 + 100n) where
                              n = number of thermometers.
DATA
```

Then follow the parameters and data exactly as for NIO 59. After each / another set of data may follow. The last set of data should end with / in the usual way followed by > on a newline. (This replaces the ―> on a Mercury input data tape. The use of > instead of / as a final data terminator does not follow the usual N.I.O. convention but means that data prepared for NIO 59 can be run with 59/A with a minimum of alteration).

Output As for NIO 59. A fault in the data will cause the words

FAULT IN DATA. SPURIOUS CHARACTER IS

followed by the faulty character, to be output. The program will then stop.

Restrictions

Number of values of T, I, t or $I_t \leqslant 50$.

Cost About £2 + 4/- per thermometer.

Method Exactly as for NIO 59.

Programmers

James Crease and Brian Hinde

N.I.O. PROGRAM 86

Title Bartlett's method for fitting an exponential functional relationship.

Language EMA

Machine ATLAS 1

Purpose To compute the best values of a and b in the relation $y = ax^b$ given a set of points (x, y) by using BARTLETT'S method to fit a straight line of the form log y = log a + b log x. The 95% confidence limits for the slope, b, are also found.

Inputs

0) Program
1) Job description and data

Program Begins:-

```
COMPILER EMA
S002 NIO PROGRAM 86 BARTLETTS STRAIGHT LINE FIT 26/7/1966
```

[or date of appropriate version]

and ends

```
* * * Z
```

Job description and data

The output is on channel 0 and may be lineprinter or five or seven track tape. (1 + B/10) blocks or (100 + 10B) lines will be required where B is the number of sets of data.

The store required for compilation is 60 blocks.
The store required for execution is 20 blocks.
The computing instructions ≃ 1000 + 2nB where n is the average number of points per set of data.

The data should begin:-

DATA

DATA TITLE

One line of title

1) A data code number, which must be an integer
2) The number of data points in the data (N)
3) A integer L approximately equal to N/3 (see Method)
4) The N points, both co-ordinates being positive,

x_1 y_1

x_2 y_2

----

x_N y_N

in ascending order of x.

Then may follow another set of data beginning with 1), the data code number.

The last set of data should end with / on a new line.

Output NIO PROGRAM 86 OUTPUT

The data title

For each set of data:

1) The data code number
2) The number of data points, N

3) The equation of the line in the form

$$\log y = \log a + b \log x$$

4) The 95% confidence limits of b

5) Two points at the extremes of the fitted line to aid plotting, (x_1, y_1) and (x_2, y_2)

The output ends with the words

END OF DATA

Failures

Any faults in the data, such as spurious characters, are indicated by

FAULT IN DATA, CODE NUMBER

followed by the code number of the faulty set of data.

If a fault, such as trying to take the square root of a negative number, should occur during execution the program will output

DATA CODE NUMBER

followed by the actual number,

FAULT NUMBER

and a fault number according to the following list:

33 Square root of a negative number

34 Exponent overflow

35 Logarithim of a negative number

36 Division by zero

After detection of such a fault the program goes on to the next set of data.

If the confidence limits are imaginary the words IMAGINARY CONFIDENCE LIMITS will be output.

The confidence limits will not be calculated if N=3.

Restrictions

$(x,y) > 0$

$4 \leqslant N \leqslant 500$

$(N/3-1) \leqslant L \leqslant (N/3+1)$ and L must be integer.

Method

The N data points are divided into three groups, the two end groups having the same number, L, of points chosen to be as near N/3 as possible. The three groups must be non-overlapping in the x-direction, and if two or more different values of y occur for the same values of x either side of a division, suitable adjustment must be made to the y values.

e.g. (386, 10) | (386, 12) (386, 14)

should be altered to

(386, 12) | (386, 12) (386, 12)

$\log_e x$ and $\log_e y$ are then calculated and hereafter are represented by x and y. The means of each of the two end groups $(\bar{x}_1, \bar{y}_1)$ and $(\bar{x}_3, \bar{y}_3)$ are first evaluated. The line joining these points gives the value of slope

$$b = (\bar{y}_3 - \bar{y}_1)/(\bar{x}_3 - \bar{x}_1)$$

The functional relation is then a line with this slope passing through the grand mean $(\bar{x}, \bar{y})$

$$a = e^{(\bar{y} - b\bar{x})}$$

and $\log y = \log a + b \log x$.

The confidence limits of the slope are provided by the solutions β_1, β_2 of the quadratic equation:

$$\tfrac{1}{2} L (x_3-x_1)^2 (b-\beta)^2 = t^2 (C_{yy}-2\beta C_{xy}+\beta^2 C_{xx})/(N-3)$$

where t is the value of the t statistic for the 95% confidence interval for the (N-3) degrees of freedom available within the groups,

where

$$C_{yy} = \sum_{1}^{N} (y_i)^2 - \left\{ \left(\sum_{1}^{L} y_i\right)^2 \Big/ L + \left(\sum_{L+1}^{N-L} y_i\right)^2 \Big/ (N-2L) + \left(\sum_{N-L+1}^{N} y_i\right)^2 \Big/ L \right\}$$

C_{xx} is similarly defined

$$C_{xy} = \sum_{1}^{N} (x_i\, y_i) + \left(\sum_{1}^{L} x_i \sum_{1}^{L} y_i\right) \Big/ L + \left(\sum_{L+1}^{N-L} x_i \sum_{L+1}^{N-L} y_i\right) \Big/ (N-2L)$$
$$+ \left(\sum_{N-L+1}^{N} x_i \sum_{N-L+1}^{N} y_i\right) \Big/ L$$

The confidence limits are then expressed as

$+ (\beta_1 - b)$

$- (b - \beta_2)$

where $\beta_1 > \beta_2$

Finally two points on the line are evaluated

(e^{x_1}, ae^{bx_1}) and (e^{x_N}, ae^{bx_N})

Notes

The method is described fully in

"Statistical Methods in Research and Production", edited by O.L. Davies, pub. by Oliver & Boyd (1958) p. 175

and

Bartlett, M.S. "Fitting a straight line when both variables are subject to error", Biometrics, Vol. 5, No. 3 (1949) p. 207.

Programmer BRIAN HINDE

N.I.O. PROGRAM 86/A

Title Bartlett's method for fitting an exponential functional relationship.

Language EMA

Machine ATLAS I

Purpose This program is identical to NIO 86, fitting the curve $y = ax^b$ to sets of values of x and y. However, the 70% confidence limits for b are found instead of the 95% limits.

Inputs 0) Program

1) Job description and data

Program Begins:-

COMPILER EMA

S002 NIO PROGRAM 86/A BARTLETTS STRAIGHT LINE FIT 20/3/1967 [or date of appropriate version]

and ends

CLOSE

***Z

Job description and data

Exactly as for NIO 86.

Output NIO PROGRAM 86/A OUTPUT

The data title.

For each set of data:

1) The data code number.

2) The number of data points, N.

3) The equation of the line in the form

$$\log y = \log a + b \log x.$$

4) The 70% confidence limits of b.

5) Two points at the extremes of the fitted line to aid plotting, (x_1, y_1) and (x_2, y_2).

The output ends with the words

END OF DATA

Failures As for NIO 86

Restrictions $(x, y) > 0$.

$0 <$ Data code number $< 10^6$.

$4 \leq N \leq 500$ for full execution; $N = 3$ will result in the confidence limits not being calculated.

$(N/3 - 1) \leq L \leq (N/3 + 1)$; L must be integer; $N \geq (2L + 1)$.

Method As for NIO 86, except that the values of the t statistic for the 70% confidence interval for the $(N - 3)$ degrees of freedom have been substituted in place of the 95% values.

Programmer Brian Hinde

N.I.O. PROGRAM 86/B

Title

Bartlett's method for fitting an exponential functional relationship.

Language

EMA

Machine

ATLAS I

Purpose

This program is similar to NIO 86, fitting the curve $y = ax^b$ to sets of values of x and y, but also compares the values of b obtained from two different sets of values of x and y.

Inputs

0) Program
1) Job description and data (1)
2) Data (2)

Program

Begins:-

COMPILER EMA

SO02 NIO PROGRAM 86/B SLOPE COMPARISON 13/9/1967
(or date of appropriate version)

and ends

CLOSE

***Z

Job description

The output is on channel 0 and may be lineprinter or five or seven track tape. (1 + B/15) blocks or (100 + 6 B) lines will be required where B is the number of pairs of sets of data.

The store required for compilation is 60 blocks
The store required for execution is 20 blocks
The computing instructions ≈100 + 3nB where n is the average number of points per set of data.

Data

The sets of data to be compared must be input on streams 1 and 2, each stream containing the same number of sets of data in corresponding order.

Following the appropriate heading, the data begins:-

DATA TITLE
One line of title

1) A data code number, which must be an integer
2) The number of data points (N or N′) in this set
3) An integer L or L′, approximately equal to N/3 or N′/3 (see Method of N.I.O. 86)
4) The N or N′ points, both co-ordinates being positive, in ascending order of x,

e.g. x_1 y_1
x_2 y_2
....
x_N y_N

Then may follow another set of data beginning with 1), the data code number.

The last set of data should end with / on a new line.

Output

NIO 86/B OUTPUT
The data title on input stream 1
The data title on input stream 2
The two data code numbers
The values of b and b′
The number of degrees of freedom $(N + N' - 6)$
The value of the t statistic (see Method)

The output ends with the words

END OF DATA

Failures

Any faults in the data, such as spurious characters are indicated by

FAULT IN DATA, CODE NUMBER

followed by the code number of the data being read and that of the last set to be read on the other input stream.

If either N or N′ is less than 4, an error message will be printed and the program will proceed to the next sets of data.

Restrictions

$(x, y) > 0$

$0 <$ Data Code number $< 10^6$

$4 \leqslant N_1 \leqslant 500, \; 4 \leqslant N_2 \leqslant 500$

$(N/3 - 1) \leqslant L \leqslant (N/3 + 1)$; L must be integer; $N \geqslant (2L + 1)$

$(N'/3 - 1) \leqslant L' \leqslant (N'/3 + 1)$; L′ must be integer; $N' \geqslant (2L' + 1)$

Method

After evaluating the two slopes b and b′ (see Method of N.I.O. 86), t is calculated according to the following formula:

$$t = \frac{(\bar{x}_3 - \bar{x}_1 - \bar{x}'_3 + \bar{x}'_1)(b - b')}{\sqrt{\dfrac{2 s_B^2}{L} + \dfrac{2 s_{B'}^2}{L'}}}$$

where t has $(N + N' - 6)$ degrees of freedom,

where

$$s_B^2 = s_y^2 - 2Bs_{xy} + B^2 s_x^2$$

and

$$s_x^2 = \frac{1}{N-3}\left[\Sigma (x - \bar{x}_1)^2 + \Sigma (x - \bar{x}_2)^2 + \Sigma (x - \bar{x}_3)^2\right]$$

making s_x^2 the pooled within-group variance of x where $\Sigma (x - \bar{x}_1)^2$ is the sum of the squared deviations from the mean of the first group of observations in the first group only, and so on.

Similarly,

$$s_y^2 = \frac{1}{N-3}\left[\Sigma (y - \bar{y}_1)^2 + \Sigma (y - \bar{y}_2)^2 + \Sigma (y - \bar{y}_3)^2\right]$$

and

$$s_{xy} = \frac{1}{N-3}\Big[\Sigma (x - \bar{x}_1)(y - \bar{y}_1) + \Sigma (x - \bar{x}_2)(y - \bar{y}_2) + \Sigma (x - \bar{x}_3)(y - \bar{y}_3)\Big]$$

The significance of the difference of slopes can then be looked up in a table of the t-distribution.

Notes

The method is described in

"Quantitative Zoology", by Simpson, Roe and Lewontin, pub. by Harcourt, Brace and Company (1960), p.237.

Programmers

Catherine McEwen and Brian Hinde

N.I.O. PROGRAM 87

Title Gravity anomalies

Language EMA

Machine ATLAS 1

Purpose To compute the anomaly in the earth's gravitational field along a profile perpendicular to parallel horizontal prisms of infinite lateral extent and specified density.

Inputs
1) Program
2) Job description and data

Program (Capitals denote actual punching)

COMPILER EMA

S002 NIO PROGRAM 87 GRAVITY ANOMALIES 7/9/1966 (or date of appropriate version)

MAIN 1010

AUXILIARY (0,0)

DUMPS 0

Then follows the program, ending with

***Z

Job description and data

JOB

Job number, Job Title

INPUT

0 S002 NIO PROGRAM 87 GRAVITY ANOMALIES 7/9/1966 (or other date as on program tape)

SELF = 1

OUTPUT

0 FIVE-HOLE PUNCH m BLOCKS

STORE 15/60 BLOCKS

COMPUTING t INSTRUCTIONS

The number of blocks of output

$$m = 1 + 0\cdot005\ BN$$

where B is the number of sets of data, and N is the number of points at which the anomaly is to be calculated.

The computing instructions $t = 500 + 0\cdot1\ (Nn')$

where n' is the number of points of data.

Then follows the data:-

DATA

DATA TITLE

Title of the first set of data (one line only)

A' B' C' Z' 0 (or 1) where the anomaly is to be calculated at points on z = Z' at interval B' in x between A' and C' (B' must be positive). 0 or 1 is inserted according to whether the units of x and z are nautical miles (0) or kilometres (1)

D = Density of first prism (grms/cc)

$X_0, Z_0, X_1, Z_1, \ldots, X_n, Z_n, X_0, Z_0$ = co-ordinates of the corners of the first prism listed in clockwise direction round the prism. Note that the co-ordinates of the first corner are repeated at the end of the list.

The remaining prisms are also listed, starting with their density, and the final prism is terminated with

ȸ ȸ >

The next set of data then follows, beginning with

DATA TITLE

The last set of data should be terminated with

ȸ ȸ /

<u>Output</u>

For each set of data:
The data title;
The x co-ordinate at the points at which the anomaly is to be calculated and the anomaly in milligals, printed in two columns.
The output ends with the words "END OF DATA".

<u>Restrictions</u>

No prism may have more than 200 corners, and the number of anomalies to be computed ≤ 200.

<u>Failures</u>

1) If the data contains any spurious character the words "FAULT IN DATA" are output. The program then proceeds to the next set of data.

2) If any prism contains more than 200 corners, "K > 200" is output and the program proceeds to the next set of data.

3) If the initial co-ordinates of a prism are not repeated at the end of the list of points, the words "ERROR IN PRISM" followed by the prism number will be output. If the error was in the first prism the program will proceed to the next set of data; if the error is in the second or subsequent prism the anomalies so far computed will be output.

<u>Cost</u>

About 15/- per set of data

<u>Method</u>

The method employed is that described in Talwani, M., J.L. Worzel and M. Landisman, 1959 - Rapid gravity computations for two-dimensional bodies, with application to the Mendocino submarine fracture zone. J. Geophys. Res. 64, 49-59.

Let A B C D E F .... be a given polygon with n sides and let P be the point at which the attraction due to this polygon has to be determined. Imagine P to be the origin of an xz system of co-ordinates, where the polygon also lies in the xz plane. Let z be positive downwards and x positive to the right.

Let R be any point on AB and let θ be the angle the line PR makes with the x axis, measured clockwise.

The vertical and horizontal components of gravitational attraction due to the polygon, at the origin, are

$$2G\rho \oint z \, d\theta$$

and $2G\rho \oint x \, d\theta$ respectively where G is the universal gravitational constant

$(6{\cdot}67 \times 10^{-8})$

and ρ is the volume density of the body.

Now it can be shown that

$$\int_{AB} z\, d\theta = \int_A^B \frac{a_i \tan\theta \tan\phi_i}{\tan\phi_i - \tan\theta}\, d\theta \equiv Z_i$$

where BA produced meets the x axis at Q at an angle ϕ_i and $PQ = a_i$

∴ The vertical gravity anomaly G_j due to polygon j is

$$G_j = 2G\rho \sum_{i=1}^{n} Z_i$$

The program thus computes the anomaly due to each prism (i.e. polygonwith infinite y dimension) and sumsthen over all prisms giving a total G_N where $N = \frac{C'-A'}{B'} + 1$.

For the general case:

$$Z_i = a_i \sin\phi_i \cos\phi_i \left[\theta_i - \theta_{i+1} + \tan\phi_i \log_e \frac{\cos\theta_i(\tan\theta_i - \tan\phi_i)}{\cos\theta_{i+1}(\tan\theta_{i+1} - \tan\phi_i)} \right]$$

where $\theta_i = \tan^{-1}\frac{z_i}{x_i}$ $\qquad \phi_i = \tan^{-1}\frac{z_{i+1} - z_i}{x_{i+1} - x_i}$

and $a_i = x_{i+1} + z_{i+1} \cdot \frac{x_{i+1} - x_i}{z_i - z_{i+1}}$

When P is not the origin, x_i is replaced by $(x_i - X)$ where X is the x co-ordinate of P, etc.

The program takes appropriate action in the following cases, giving the correct gravity anomalies:

1) $x_i = 0$
2) $x_{i+1} = 0$
3) $z_i = z_{i+1}$
4) $x_i = x_{i+1}$
5) $\theta_i = \theta_{i+1}$ (when $Z_i = 0$)
6) $x_i = z_i = 0$ (when $Z_i = 0$)
7) $x_{i+1} = z_{i+1} = 0$ (when $Z_i = 0$)

<u>Programmer</u> BRIAN HINDE (This program is based on a flow diagram prepared by F.J. Vine and published in his Cambridge Ph.D. dissertation, August 1965)

N.I.O. PROGRAM 88

Title	Atlas Costs
Language	EMA
Machine	ATLAS 1
Purpose	To evaluate the cost of each job run on Atlas 1 in pounds, shillings and pence.
Inputs	O) Program 1) Job description and data
Program	COMPILER EMA S002 NIO PROGRAM 88 ATLAS COSTS 23/9/1966 Then follows the program ending with CLOSE ~~xxx~~Z

Job description and data

JOB

Job No., Job title

INPUT

O S002 NIO PROGRAM 88 ATLAS COSTS 23/9/1966

SELF = 1

OUTPUT

O FIVE-HOLE PUNCH n BLOCKS

STORE 15/60 BLOCKS

COMPUTING 1000 INSTRUCTIONS

where n = 1/40 the total number of jobs run.

Then follows the data:-

1) DATA

2) DATA TITLE

3) ATLAS COSTS followed by the two dates covering the current Atlas invoice.

4) Each set of data (one for each job run) consisting of the following:-

a) Job number

b) Run number

c) Data link number, of O if the job was not run over the link

d) The day and month on which the job was run. The numbers are separated by a full stop

e) The total number of instructions used

f) The number of instructions used in compilation

g) The requested number of blocks for compilation store

h) The requested number of blocks for execution store

i) The number of tape decks used

j) The number of tape block transfers used

k) Total number of blocks input

l) Total number of blocks of tape output

m) Number of records of line printer output

n) Number of cards punched

c) Level of job:- E = express
N = normal
P = priority

The data ends with > on a new line followed by xxxZ

Output

The output is headed with the data title, and with the following heading on the next line:-
JOB RUN D/L DATE INS £ s d

The appropriate information is printed in the various columns followed by the cost of the run.

If a spurious character is found in the data, the program prints "FAULT IN DATA, P = "followed by the spurious character.

Cost

Approximately £1. 14. 0. for 20-30 sets of data. £1 of this is the fixed data link charge. For a normal run by post this job would cost about 11/6.

Method

The program computes the following formula

$$\left[\frac{(CS + AN + 4JN)}{263} + 120\ M \right] \text{ pence + input/output charges}$$

where C = number of instruction interrupts obeyed during compilation

S = number of store blocks requested for compilation

A = number of instruction interrupts obeyed during execution

N = number of store blocks requested for execution

J = number of magnetic tape block transfers during execution

M = number of magnetic tape decks loaded.

If a job is run at express rate the program computes the input/output charges as follows:- 1/- per block of cards or tape input + 2/- per block of tape output + 2d. per card output + $\frac{1}{3}$d per line of line printer output. For a priority or normal job the charges are 1/6 per block of cards or tape input + 4/- per block of tape output + 2d. per card output + $\frac{1}{3}$d per line of line printer output. For a normal job the formula cost + input cost + output cost are simply added. If the job is run at express rate the complete cost is [£1 + 3/2 (formula + input + output)]. For a prioity job the cost is [£2 = 4/3 (formula + input + output)].

The program evaluates the relevant formula for each run in pounds, shillings and pence. Any fraction of a penny is rounded upwards. If the cost is less than £1 then there is a fixed charge of £1.

Programmer

Eileen Squire

N.I.O. PROGRAM 89

Title	Analysis of shipborne wave records
Language	EMA
Machine	ATLAS 1
Purpose	Given values of the highest and second highest crests, the lowest and second lowest troughs, the number of zero crossings and the number of crests in a short record from the N.I.O. shipborne wave recorder, to compute the spectral width parameter and the significant wave height and also the predicted maximum height in a period of three hours.
Inputs	0) Program 1) Job description and data

Program

COMPILER EMA

S002 NIO PROGRAM 89 SBWR ANALYSIS 15/12/1966

Then follows the program ending with

CLOSE

***Z

Job description and data

JOB

Job No., Job Title

INPUT

0 S002 NIO PROGRAM 89 SBWR ANALYSIS 15/12/1966

SELF = 1

OUTPUT

0 LINEPRINTER k LINES

STORE 20/60 BLOCKS

COMPUTING (500 + 2k) INSTRUCTIONS

where k ≃ 3 times the number of lines of data.

Then follows the data:-

1) DATA
2) DATA TITLE
3) The name of the ship from which the records were taken and the year of recording.
4) Three parameters consisting of the depth (d) in feet of the instrument below the sea surface, followed by the year and month (in figures) in which the records were taken.
5) Each line of data consists of the following:-
   a) The day of the month
   b) The time at which the record started (e.g. 2100)
   c) The length of record in minutes (U)
   d) The height in feet of the highest crest above the mean water level (A)
   e) The height in feet of the second highest crest above the mean water level (B)
   f) The height in feet of the lowest trough below the mean water level (C)
   g) The height in feet of the second lowest trough below the mean water level (D)

h) The number of zero crossings in the record ($2N_z$)

i) The number of crests (above and below the mean water level) in the record (N_c)

A change of month is indicated on the data tape by s̸s̸> after the last day's data. The data ends with s̸s̸/ the last line of data followed by Z in the usual way.

If data is not available or the record not measurable, one of the following options can be taken:-

a) Record Missing:- Insert M after the day and time (i.e., in the length of record column)

b) Record Faulty:- Insert F

c) Waves too small to measure:- Insert C (for calm)

Output

The output is headed with the name of the ship on one page, and the month and year at the top of the next page. Each page is then headed with

DATE TIME TZ H1 H2 E H1' H2' HS1 HS2 HS HMAX(3 HRS) DURATION

The program then computes the following, and prints the answers in the appropriate columns:-

a) the day of the month

b) the time at which the record started

c) the wave period in seconds, correct to two decimal places (T_z)

d) the height, in feet, correct to one decimal place, of the highest crest plus the lowest trough (H_1)

e) the height, in feet, correct to one decimal place, of the second highest crest plus the second lowest trough (H_2)

f) the spectral-width parameter (E) to three decimal places

g) H_1 corrected to allow for instrumental response (H_1')

h) H_2 corrected to allow for instrumental response (H_2')

i) and j) the mean height of the highest one-third of the waves each correct to one decimal place calculated from H_1' and H_2' respectively (Hs_1 and Hs_2)

k) the average of Hs_1 and Hs_2 to give Hs. The program computes the average of the numbers contained in the machine store and not those printed under Hs_1 and Hs_2.

l) the most probable height of the highest wave which would occur in a period of three hours, correct to one decimal place ($H_{max(3\ hrs.)}$)

m) the duration of the record in minutes.

If no measurements were available, the words

"RECORD FAULTY" or
"RECORD MISSING"

will be printed on the right-hand side of the page against the appropriate date and time.

If the record was too small to measure, the word "CALM" will be printed in the TZ column.

Cost

Approximately £3. 10s. 0d. for a month's data.

Method

The zero-crossing period is given by

$$T_z = 120U/[2N_z] \text{ seconds}$$

then $H_1 = A + C$

$$H_2 = B + D$$

and these are corrected for instrumental and hydrodynamic factors using the formulae

$$H_1' = k\,H_1$$

$$H_2' = k\,H_2$$

where $k = 0{\cdot}83\,[1 + (8{\cdot}8\mu)^{-2}]^{3/2} \exp\,(2{\cdot}5d\mu^2/g)$

where $\mu = \dfrac{2\pi}{T_z}$ and $g = 32{\cdot}174$ ft. sec.$^{-2}$

The spectral-width parameter

$$E = \sqrt{1 - \frac{(T_c)^2}{(T_z)^2}}$$

where $T_c = 60U/N_c$ seconds.

The values of significant wave height as determined from H_1' and H_2' are

$$Hs_1 = \frac{2\,H_1'}{(2\theta)^{\frac{1}{2}}\,(1 + 0{\cdot}289\theta^{-1} - 0{\cdot}247\theta^{-2})}$$

and $$Hs_2 = \frac{2\,H_2'}{(2\theta)^{\frac{1}{2}}\,(1 - 0{\cdot}211\theta^{-1} - 0{\cdot}103\theta^{-2})}$$

where $\theta = \log_e\,(N_z)$

The average significant wave height is then given by

$$Hs = \frac{Hs_1 + Hs_2}{2}$$

The prediction of $H_{max(3hrs.)}$ is done by computing

$$y = \sqrt{\log_e\left(\frac{180\,N_z}{U}\right)}$$

and then

$$H_{max(3\ hrs.)} = \frac{\sqrt{2}}{2}\,(0{\cdot}006361y^4 - 0{\cdot}073968y^3 + 0{\cdot}330573y^2 + 0{\cdot}316548y + 0{\cdot}566405)\ Hs$$

The coefficients in this equation were derived from a least squares fit of the points contained in a table by Longuet-Higgins (J. Mar. Res. 11, 259) relating the most probable value of a_{max} in a given interval to the square root of the number of waves in the interval. For $10 \leq N_z \leq 100{,}000$ the maximum residual is less than 0·05%.

Programmer Brian Hinde

N.I.O. PROGRAM 90

Title: Tables of the Integral of Associated Legendre Polynomials

Language: EMA

Machine: ATLAS 1

Purpose: To compute tables of the integral $\int_{-1}^{1} P_n^m P_s^r \, dz$ for integer n, m, s, r up to a given integer N.

Input: 0) Job description, program, data

Tape:

JOB
S002 Job number, data link number LEGENDRE POLYNOMIALS
OUTPUT
O LINE PRINTER ℓ LINES
STORE 15/60 BLOCKS
COMPUTING i INSTRUCTIONS

COMPILER EMA
MAIN 60
AUXILIARY (0,0)
DUMPS 0

>> NIO PROGRAM 90 date
>> DATA IS FOR DEGREE N

Then follows the program, ending
CLOSE

Then follows the data, the single integer N which represents the highest value of n required in the tables,

and the tape ends with
~~***~~Z

ℓ, the number of lines of output, $= 1{\cdot}2 \left[(N/2 + 1)^4 - (N/2)^2 (N/2 + 2) \right]$

or if paper tape is required as output, number of blocks is $(1 + \ell/70)$

i, the number of instructions, = 3500 approx.

Output: This is a table of values of the integral as a function of n, m, s, r with an additional column when (m+r) is odd giving the integral/Π.

Values are given for $N \geq n \geq m$, $n \geq s \geq r$. If n=s, then $m \geq r$. When (n+m+s+r) is odd, the integral is zero and is not given.

Restrictions: N < 23

If this restriction is to be waived, then the X directive and main store specification must be increased to (2N + 4).

Cost: About £5 for degree 10

Method: The method used is to evaluate the integral using a single series expression found by J. CREASE (see N.I.O. Internal Report A.28).

Attention has been paid to the programming of the expression to retain integer arithmetic as far as possible and hence keep the maximum accuracy. Double length arithmetic has been used in summing the terms of the series.

The program and strategy adopted are not particularly suitable for adaption as a routine to find individual values of the integral.

Programmer J. Crease

N.I.O. PROGRAM 91

Title Revision of Matthews Tables (2, Gulf Stream)

Language EMA

Machine ATLAS I

Purpose This program was written and used for the same project as NIO Program 70, i.e. to investigate Matthews Tables.

In this instance the program was used with hydrographic data for the Gulf Stream within the latitudes 33 to 44 °N, longitudes 50 to 76 °W.

Inputs 0) Program and program constants
1) Job description and data (including parameters and titles).

Program and program constants

Tape 1 consists of the program; it starts

COMPILER EMA

S002 NIO PROGRAM 91 MATTHEWS REVISION (2) 8/12/1966

MAIN 80

AUXILIARY (0, 0)

DUMPS 0

Then follows the program, ending with

ACROSS 1/1

***T

Tape 2 contains the 101 program constants (or basic data) in floating point decimal, as a table; it starts with

101) 980.62

and ends with

CLOSE

***Z

Job description and data

This starts

JOB

Job number, data link number and job title

INPUT

0 Program title

SELF = 1

OUTPUT

0 LINEPRINTER ℓ LINES

STORE 20/60 BLOCKS

COMPUTING q INSTRUCTIONS

where ℓ = 200 + 200 (no. of stations)
and q = 1000 + 400 (no. of stations)
Both ℓ and q are estimates for deep stations and both may be considerably reduced for shallow stations.

The job description is followed by

DATA

and the parameters m and a_i (as in NIO Programs 58 and 70) where a_i are the standard pressures and intervals for $i = 0(1)m-1$.

Then comes the data, starting with one line of data title which usually contains the ship and station number, geographical position and date. Then on a new line follows the nearest integer value of latitude. The values of depth, temperature and salinity follow as described in NIO Program 58, followed by > .

New sets of data follow, starting with the data title and ending with > .

The final set of data is followed by

END OF DATA

/

***Z

<u>Output</u>

The output for each station consists of:-
station title

a) results at true depths i.e. observed depths
b) results interpolated from a) to standard true depths; this is headed "standard depths corrected".
c) results interpolated from b) to standard echo-sounding depths; this is headed "standard depths uncorrected".

Sections a), b) and c) are on separate pages on the line printer output.

At each depth a Matthews-type correction is calculated so that

echo-sounding depth + correction = true depth

In output sections a) and b) (which are the same as in NIO Program 70) the output is in terms of true depth and correction, so that

true depth (as listed) - correction (as listed) = echo-sounding depth

In section c) the echo-sounding depth (as listed) + correction (as listed) = true depth

In section a) the output consists of:-

sample number
depth, metres
correction
pressure, db
sound velocity, metres/second
sounding velocity, metres/second
potential temperature, °C
potential density, i.e. σ_θ

In sections b) and c) the output consists of:-

depth, metres
correction
interpolation error
sound velocity, metres/second
interpolation error
sounding velocity, metres/second
interpolation error

See program descriptions for NIO Programs 58 and 70 for details of output and method.

Cost — About 7/6 per station

Method — Basically as for NIO Program 58. Note that when depths are input they are immediately converted to pressure (db) and working is done in terms of pressure.

Note — This program has been compiled onto magnetic tape and may be used from there, the whole of input 0 constituting the program.

When the magnetic tape version is used the store required is 20/25 blocks and computing instructions q = 20 + 400 (no. of stations).

Programmer — James Crease

N.I.O. PROGRAM 92

Title General spectral analysis

Language EMA

Machine ATLAS 1

Purpose To compute the auto- and cross-covariances, and auto- and cross-spectra of up to 99 time series. Facilities include error detection and logging, selection of series, selection of pairs for cross-correlation, pre-whitening, corrections for the amplitude and phase response of the recording system, omission of parts of the computations not required and plotting of auto-spectra.

Inputs

a) Using paper tape program:-

0) Program
1) Job description and first data tape
2) to 14) Data tapes.

b) Using magnetic tape compiled program:-

0) Job description
1) to 14) Data tapes.

Long sets of data may be split up into convenient lengths and input as several consecutive documents. Every document ends with *** Z or *** C.

Program

```
COMPILER EMA

S001 NIO PROGRAM 92 GENERAL SPECTRAL ANALYSIS 10/3/1967
                              (or date of appropriate version)

MAIN p

AUXILIARY (0, q)

DUMPS 0


CHAPTER 1
     - Rearranges auto-spectra in auxiliary store ready
       for plotting.


CLOSE


CHAPTER 2
     - Reads covariance data ready for Fourier transformation.


CLOSE


PROGRAMME -4

>> GRAPH PLOTTING

CHAPTER 1
     - Plots auto-spectra.

CLOSE


PROGRAMME -5

>> GENERAL SPECTRAL ANALYSIS

CHAPTER 1
     - Computes auto- and cross-covariances.

CLOSE

CHAPTER 2
     - Computes auto- and cross-spectra.

CLOSE
```

CHAPTER O

- Reads, sorts and selects data.

CLOSE

***Z

The directives at the head of each chapter should be set as follows (they will only need alteration for very large jobs):-

CH.1 $X \geq 2(M + 1)$

$Z \geq K^2(M + 1)$

CH.2 $X \geq K^2(L + 1)$

$Y \geq K(M + 1)$

$B \geq K(K - 1)/2$

PR.-4

CH.1 $D, E, X, Y \geq (M + 1)$

PR.-5

CH.1 $X \geq K(L + 1), 8(2L + 1), 1000$

$Y \geq K^2(L + 1)$

$A, C, U, V, W, \Pi \geq K$

$B \geq K(K - 1)/2$

CH.2 $Y \geq K^2(L + 1)$

$Z \geq K^2(M + 1)$

$B \geq K(K - 1)/2$

$F, G \geq K(M + 1)$

$U, V \geq K$

CH.O $A \geq R$

$B \geq K(K - 1)/2$

$X > 100\ R$

$Y > 100\ K$

$\Pi \geq R$

where L = number of covariance lags, M = number of frequencies, K = number of series selected for analysis and R = total number of series read in.

The MAIN store requirement (p) should be set to be greater than the maximum number of main variables required in any of the six chapters. (Note that X ⟶ 1000 allocates 1001 storage locations) (Remember to allow 8 for index stores in programme -4 and 4 in programme -5). The AUXILIARY store requirement (q) should be set to a value greater than $[D' + K^2(L + M + 2)]$ or $[D' + NK]$, whichever is the greater, where $D' = (K/2)(4M + K + 11)$ and where N is the number of terms per series.

When using the program with several sets of data, the auxiliary directive should be set to the largest required value.

Job description

INPUT: Up to 14 data documents may be listed. If a data tape is split into parts, they must be input on consecutive documents, numbered in the correct order.

OUTPUT: The amount of output depends on the setting of parameters F, K, L, M, P, V′ and H′ (see "data" below). Numerical results appear on output 0 and auto-spectral graphs on output 1. Even if no graphs are required, output 1 MUST be specified as a nominal 50 lines. On output 0 allow

B[200

+L(1 + int. pt.(K − 1)/7) for auto-covariances

+ 2L(1 + int. pt.(KP(K − 1)/14) for cross-covariances

+ M(1 + int. pt.(K − 1)/7) for auto-spectra

+ (M + 6)(KP(K − 1)/2) for cross-spectra] lines,

where B = number of sets of data and P = the fraction of the total options required. e.g. P = 8/15 for 8 out of 15 possible pairs.

On output 1 (if required) allow

BK(50 + 2M) lines for graph plotting.

For output on 7-track paper tape (5-track should not be used) the formulae are

B[1·0

+ 0·005 LK for auto-covariances

+ 0·005 LKP(K − 1) for cross-covariances

+ 0·005 MK for auto-spectra

+ 0·01 MKP(K − 1) for cross-spectra] blocks on output 0 and

BK[2 + M/25] blocks on output 1 for graph plotting.

The auto- and cross-spectral outputs will be doubled if pre-whitening is used.

STORE: The store required for execution is

$$B' + 40 + \left\{ \begin{array}{l} \text{Maximum of} \\ (M+1)(K^2+2), \\ K(K(L+2)+M+1), \\ 4(M+1), \\ [K(KL+2K+6) + \text{Maximum of } K(L+1), \\ \qquad 8(2L+1),\ 1000], \\ K(K(M+L+3)+2M+5), \\ R(2+100)+K(K+100) \end{array} \right\} /512$$

$$+ \left\{ \begin{array}{l} \text{Maximum of} \\ D' + NK, \\ D' + K^2(L+M+2) \end{array} \right\} /512$$

where $D' = K/2[4M + K + 11]$

and B′ = Number of input streams used.

The store required for compilation is 85 blocks.

COMPUTING: The number of computing instructions used will be less than

$$\left[4000 + \frac{BKNL}{80}(1 + P^2K)\right]$$

$$+\left[B(1000 + 10MK)\right]$$ if graph plotting is required.

If H′ = 4, 8 or 12 (spectra only) the first formula becomes:

$$[4000 + 10BK^2L]$$

Data

Each set of data begins with a number of parameters outlining the user's requirements. Long data tapes may be split into shorter lengths and input as several documents, in which case the parameters appear only once. If desired, the parameters may be on a separate document, but their document number must be one less than that of the data to which they refer. Data input numbers begin at 1 and must be consecutive. A typical arrangement might be:-

INPUT 0	Job description (prog. on mag. tape)	Probably on one tape.
INPUT 1	Parameters for 1st set of data	
INPUT 2	1st set of data	
INPUT 3	Parameters and 2nd set of data	
INPUT 4	Parameters for 3rd set of data	
INPUT 5	1st part of 3rd set of data	
INPUT 6	2nd part of 3rd set of data.	

There are three data terminators as follows:-

>> End of document and this set of data,

)) End of document, but not end of this set of data,

// End of last document.

The terminators must be punched following a correctly terminated term in series R and they may not appear before the * in the parameters [see (22) below].

Thus, in the above example the terminators would be:

INPUT 0		***C
INPUT 1	))	***Z
INPUT 2	>>	***Z
INPUT 3	>>	***Z
INPUT 4	))	***Z
INPUT 5	))	***Z
INPUT 6	//	***Z

It follows that if there is only one data document it will end with // ***Z. Covariance data and parameters MUST be on one document only [H′ = 4, 8, 12 - see (2) below]. The parameters which precede each set of data are as follows:-

Restrictions

1) DATA TITLE

One line of title

	Restrictions
2) An integer (H′) having values between 1 and 15, this being the total of up to four integers chosen from the following: 1 if auto-covariances are required 2 if cross-covariances are required 4 if auto-spectra are required 8 if cross-spectra are required [e.g. 9:- auto-covariances and cross-spectra only will be output.] Note that if H′ = 4,8,12 the program will expect to read covariance data [see (24) below].	
3) An integer (A) equal to the number of documents to be used for this set of data.	Omit when H′ = 4,8,12
4) An integer (L) equal to the number of covariance lags required or read in.	
5) An integer (M) equal to the number of frequencies required for spectral analysis.	Omit when H′ = 1,2,3
6) An integer (R) equal to the total number of series to be read in.	Omit when H′ = 4,8,12
7) An integer (K) equal to the number of series to be selected for analysis. (When H′ = 4,8,12 K must be set equal to the total number of series to be read in.	
8) K integers, being the numbers of the series to be selected for covariance and/or spectral analysis. [N.B. The first series is 1 and the last is R]. The numbers may be in any order. The series will now be referred to as 1.......K on input and output, e.g. If series 2,5,4 have been selected from 1,2,3,4,5 they will now be numbered 1,3,2	Omit when H′ = 4,8,12
9) An integer (P) equal to the number of cross-covariance and/or cross-spectral pairs required. P = -1 may be used to signify that all possible combinations are required. N.B. The definition of P given here is NOT the same as that in the job description section above.	Omit when H′ = 1,4,5
10) P pairs of integers, being the numbers of the series to be cross-correlated and/or cross-spectrally analysed. The pairs may be in any order but, within each pair, the lower series number must be punched first.	Omit when P = O, -1
11) K numbers, being the numbers required to convert the data into physical units (each number being a ratio of physical/digital units).	Omit when H′ = 4,8,12
12) K numbers, being the variances of the series in physical units.	Only when H′= 4,8,12

Restrictions

13) K numbers, being the time lags of each series relative to series 1 expressed as decimal fractions of the time between successive readings of the same series. Thus the first number will normally be zero.

e.g. 4 series, sampled at equal intervals:-

0 0·25 0·50 0·75

14) An integer (I) = 1 if the amplitude and phase response of the recording system was the same at all frequencies,

= 0 otherwise.

Omit when H′ = 1,2,3

15) (M + 1)K values of amplitude and phase response at each frequency, starting at zero, for each of the K series. i.e. for series j a unit input to the recording system at frequency s/2LE cycles/time unit has a response

$$a_s^j \exp\left(\frac{i\pi\phi_s^j}{180}\right), \quad \phi \text{ in degrees.}$$

Omit when H′ = 1,2,3, I = 1

The values should be arranged thus:-

$$a_0^1\ \phi_0^1 \quad a_0^2\ \phi_0^2 \ldots\ldots\ldots\ldots a_0^K\ \phi_0^K$$

$$a_1^1\ \phi_1^1 \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots$$

$$\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots$$

$$a_M^1\ \phi_M^1 \ldots\ldots\ldots\ldots\ldots\ldots a_M^K\ \phi_M^K$$

16) The sampling interval (E) between successive terms of the same series. The units of E will govern the units of the resultant spectral frequencies (see Method of programme -5).

17) The pre-whitening factor (F) (see Method). If pre-whitening is not required, set F = 0.

18) An integer (V′) = 1 if plots of the auto-spectra are required,

= 0 otherwise.

Omit when H′ = 1,2,3,8, 9,10,11

19) An integer (S) = 0 if the auto-spectral graphs are all to occupy the maximum available page width (see Method of programme -4)

= 1 if the y ordinates are to be scaled by a factor U′ [see (20) below]

Omit when V′ = 0, H′ = 1,2,3,8,9, 10,11

20) A scaling factor (U′) for auto-spectral graphs (see Method). If U′ = 1 the y scale will run from 0 to 118 energy units.

Omit when V′ = 0, S = 0, H′ = 1,2,3,8, 9,10,11

Restrictions

21) An integer (Q′); plotting of auto-spectra will begin at a frequency of Q′/2LE cycles/time unit. — Omit when V′ = 0, H′ = 1,2,3,8,9,10,11

22) An asterisk (*) after the last parameter has been correctly terminated. This is used by the program to check that the correct number of parameters have been punched. The data terminators (>>,)) and //) may not appear before this point in a data tape. A data tape containing only parameters will therefore end with

```
*

))
***Z
```

(e.g. INPUT 1 in the example above).

Then follows:-

23) The R series of data points, the first term of each series first, followed by the second and so on. — Omit when H′ = 4,8,12

24) The (L + 1)K normalised auto-covariances of the K series, the values with zero lag first for each series, followed by the values for one term lag and so on. — Only when H′ = 4,8,12

25) The P(2L + 1) normalised cross-covariances of the K series, taken in pairs in the strict order: — Only when H′ = 8,12

$R_{12}, R_{13}, \ldots\ldots\ldots, R_{1K},$

$R_{23}, R_{24}, \ldots\ldots\ldots, R_{2K},$

$R_{34}, \ldots\ldots\ldots, R_{(K-1)K},$

Only those pairs should be included that have been specified in 9) and 10) above. For each pair the covariances should run in the order $r(-L), r(-L+1), \ldots, r(0), \ldots, r(L)$. i.e. All the covariances for R_{12} are punched, followed by those for R_{13} and so on. This arrangement is different from that specified for the auto-covariances.

Output

(For complete details, refer to the descriptions of programmes -4 and -5).

The output will include some, or all, of the following items, depending on the values of H′, F and V′. The series will be referred to as numbers 1 to K in the order in which they were read in, not in the order in which they were selected.

Output stream 0:-

Data title.

Means of each digital series 1 to R.

Number of terms in each series (H′ ≠ 4,8,12).

The time lags of each series (H′ ≠ 4,8,12).

The means of each series, in digital and physical units ($H' \neq 4,8,12$).

The variances of each series, in digital and physical units ($H' \neq 4,8,12$).

The lagged normalised auto-covariances of each of the selected series (H' odd).

The cross-covariances of each series, in digital and physical units ($H' = 2,3,6,7,10,11,14,15$).

The lagged normalised cross-covariances for each of the selected pairs of series ($H' = 2,3,6,7,10,11,14,15$).

The frequency increment between successive spectral estimates ($H' \neq 1,2,3$).

The auto-spectra of each of the selected series, in physical units ($H' = 4,5,6,7,12,13,14,15$).

The cross-spectra of each of the selected pairs of series as follows: ($H' \geq 8$).

Co- and quad-spectra in physical units.

Intensity, phase (in degrees) and coherence.

When $F \neq 0$, the auto-spectra, co-spectra, quad-spectra and intensity corrected for the pre-whitening will also be output provided the reciprocal of the attenuation factor due to pre-whitening is not greater than 10^{10}.

Output stream 1:-

The auto-spectra of series 1,2,.....K with the y-axis across the page. The x-axis scale is always

$\frac{1}{3}$ inch = 1 frequency increment.

The y-axis scale is obtained by multiplying the "Scale on y-axis = " by the factor given two lines earlier. e.g. given the following output:-

"Y scale must be multiplied by 0·25 to give energy/cycle/ time unit per inch"

and

"Scale on y-axis = 10·000 units/inch"

the y-scale would be 2·5 energy units/cycle/time unit per inch

or 0·4 inch = 1 energy unit/cycle/time unit.

Thus, if the sampling interval was 2 seconds and the number of lags (L) was 25 then

x-scale: $\frac{1}{3}$ inch = 0·01 cycles/second

y-scale: 0·4 inch = 1 energy unit/cycle/second.

In order to identify the graphs with the appropriate numerical output, each graph is headed with an output number. This number corresponds to that given immediately after the data title on the numerical output and bears no relation to output or input stream numbers.

Error messages

Faults, such as non-numeric characters, in the data parameters will be detected and the words

FAULT IN PARAMETERS ON INPUT

printed followed by the input stream number at fault. This message will also appear if the asterisk between parameters and data has been omitted or punched in the wrong position,

,9

or if a parameter has been added or omitted.

Non-numeric characters in data will be logged with the words

ERROR IN DATA

followed by the faulty term number and series number. Errors in any of the R series will be recorded and at this stage the series will be referred to by the numbers 1 to R. Letters and other printing characters will all be detected, but a single space will not. This means that two adjacent numbers may be read as one number resulting finally in there being one too few terms in the last series. This will be logged with the words

SERIES HAVE DIFFERENT NUMBERS OF TERMS.

Single >,) or / will be logged as errors but >>,)) or // will be read as data terminators.

If the total number of incorrect terms exceeds 4R the logging will cease and the program will go on to the next set of data (i.e. it will search this and succeeding documents for a >>). This will also happen if the total number of individual faulty characters exceeds 100. On 5-track tape a spurious letter shift would cause all the succeeding characters to be in error, including the terminators. For this reason it is good practice to precede the terminators with a short length of runout (figure shifts). On 7-track tape, // would not be recognised following a spurious lower case character; all terminators should therefore follow runout (upper cases). The first, or only, error in covariance data, such as an incorrect number of terms input, will be logged with the words

ERROR IN COVARIANCE DATA ON INPUT

followed by the input stream number that is faulty.

If a set of data contains any errors at all, even in the series not used, the program will not carry out any computations but will proceed to reading the next set of data.

Restrictions

Integers:-

$1 \leqslant A \leqslant 14$

$1 \leqslant H' \leqslant 15$

$1 \leqslant K \leqslant 99$

$1 \leqslant M \leqslant L \leqslant 1000$

$N \leqslant 100{,}000$

$-1 \leqslant P \leqslant K(K-1)/2$

$0 \leqslant Q' < M$

$1 \leqslant R \leqslant 500$

$K \leqslant R$

$K^2(M+1) \leqslant 40{,}000$

Non-integers:-

$-1 \leqslant F \leqslant 1$

$U' > 0$

Maximum number of data documents = 14.

Cost

An estimate of cost (in £) may be made by multiplying the execution store required by the total number of computing instructions estimated and dividing by 60,000.

Method

The method of calculating covariances and spectra is fully described in the description of N.I.O. programme -5; the method of plotting the graphs is fully described in the description of N.I.O. programme -4.

The program first reads in all the parameters. The value of H′ determines whether a jump is made to chapter 2 for reading covariance parameters and data; otherwise all reading is done in chapter O. The selection of series is carried out by setting main variables A_1 to A_R equal to the numbers of the series to be retained or to -100. These variables are then tested as data is read in. The selection of pairs is carried out by setting auxiliary variables (2M + 6)K to K/2 (4M + K + 11) -1 equal to 1 for a pair required or -100 for a pair not required. The first auxiliary variable corresponds to pair (1,2), the second to (1,3) and so on to (1,K) then (2,3) etc. to (K - 1,K).

Data is read into main variables X_1 to X_{100R} and then selected according to the values A_1 to A_R into Y_1 to Y_{100K} before being transferred to the auxiliary variable store.

When a terminating character is detected, a test of the subsequent character is made. If a correct double character terminator has been found the appropriate action will be taken, otherwise an error will be logged.)) causes an immediate selection of the next data document and reading continues where it left off. >> and // cause the last set of data in X to be sorted and transferred. The means of the original R series are calculated and pre-whitening is carried out if required. After various essential indices relevant to graph plotting have been preserved a jump is made to programme -5.

Errors are logged in two ways. Individual non-numeric characters are counted in S and terms containing one or more faults are counted in T. An error message is only printed for each wrong term, not for every wrong character. Logging of errors ceases when T > 4R or S > 100. The program then searches the data document for a terminator.

Counts are kept of input stream number (F′ + 1), input number (0) and document number in any one set of data (A). Z′ is used to keep track of where reading ceases when an error occurs. Z′ = -10 during the reading of parameters, Z′ = O during the reading of data if it is the only or last document in a set of data, and Z′ = -20 during the reading of other data.

Chapter 2 is similar to chapter O except that there is no selection of series, the data being read into main variables X_O to $X_{K^2(L + 1)-1}$ and then transferred to the auxiliary store in one operation.

Chapter 1 is entered only if graph plotting is required. The auto- and cross-spectral estimates are transferred to the main store Z and the auto-spectral estimates picked out and stored in main store X after scaling by U′ . The maximum value of ordinate is found and, is this exceeds 118, all the values are multiplied by 118/max. value; U′ is adjusted similarly. Certain indices are preserved, the x and y scales are printed and programme -4 is entered once for each series 1........K.

Notes

This program uses N.I.O. programme -4 (Graph plotting; programmer: Margaret Ringrose) and N.I.O. programme -5 (General spectral analysis; programmers: James Crease and Brian Hinde).

Programmer

Brian Hinde.

N.I.O. PROGRAM 93

Title Least squares polynomial fitting

Language EMA

Machine ATLAS I

Purpose To fit polynomials $y = f(x)$ having degrees up to k to a set of n points, not necessarily of equal weight.

Inputs
0) Program
1) Job description and data

Program Begins:-

COMPILER EMA

S002 NIO PROGRAM 93 LEAST SQUARES FITTING 14/3/1967
(or date of appropriate version)

and ends

*** Z

Job description and data

The output is on channel 0 and may be line printer or five or seven track tape. $(N/60 + k/120\ (k + 5))$ blocks for each set of data plus 2 blocks, or $(100 + N + k/2\ (k + 5))$ lines will be required for each set of data if all possible output printing is required. (N is the number of points and k is the maximum degree required).

The store required for compilation is 65 blocks. The maximum store required for execution is 30 blocks. The computing instructions $\approx 1500 + 2Bkn$ where B is the number of sets of data, and n is the average number of points per set.

The data should begin:-

DATA

DATA TITLE

One line of title.

An integer, n, the number of points.

An integer, k, the maximum degree required.

An integer, c, = 0 if the x co-ordinates are in arithmetic progression, 1 otherwise.

An integer, d, = 0 if the points are to be equally weighted, 1 otherwise.

An integer, e, = 0 for printing of standard and orthogonal data,
= 3 for omission of orthogonal data
= 6 for final polynomial only.

[To secure the printing of residuals at degree k, add 1 to the above values of e. If 2 is added instead, both residuals and predicted values are printed. [e = 4 or 5 will be the most usual requirement.]]

The initial value of x [only if c = 0]

The common difference [only if c = 0]

For each point:-

The x co-ordinate [omit if c = 0]

The y co-ordinate

The weight [omit if d = 0]

[If c = O, the data points must be arranged in ascending order of x to match the x values generated in the computer].

Each new set of data begins with:-

DATA TITLE

and the last set is followed by the sequence

DATA TITLE

END OF DATA

/

and then

***Z

Output

NIO PROGRAM 93 OUTPUT

The data title.

The polynomial coefficients for all polynomials and orthogonal polynomials up to degree k followed by the divisor, D_j (see Method).

The residual at each point for the kth degree polynomial and the y value predicted from the polynomial at each point.

The component of the sum of squares of the residuals after fitting each polynomial, the coefficients, o_j (see Method) and the value and position of the maximum residual are next printed.

The sum of squares of residuals at degree k is printed followed by a check value (see Method) found by direct evaluation of the standard polynomials. Finally, the mean square (i.e. the sum of squares of residuals divided by the number of degrees of freedom) is printed.

The residual at any other degree m (m < k) can be found by adding to the sum of squares of residuals the components at degrees (m + 1), (m + 2).....k. If this and the maximum residual at degree m are acceptable, the coefficients at degree m can be used.

Restrictions

$n \leqslant 500$

$k \leqslant 12$

Cost

For n = 150, k = 12, cost is under £4.0s.0d.

Method

The method used is fully described in "Forsythe, G.E., J. Soc. Indust. Appl. Math. 5, No. 2, June 1957" and the results are printed in both standard and orthogonal forms:-

$$y = \sum_{j=0}^{\alpha} e_{\alpha j}\, x^j \qquad y = \sum_{j=0}^{\alpha} o_j\, \phi_j\,(x)$$

$$\text{var}\,(o_j) = \frac{\alpha^2}{D_j} \quad \text{for each } \alpha \text{ up to k.}$$

The estimate of α^2 is made on the assumption that

$$\text{var}\,(y_i) = \alpha^2/w_i$$

Notes

This program is based on the R.A.E. program 109/A written by J.H. Cadwell.

Programmer

Brian Hinde.

N.I.O. PROGRAMS 94, 94/A, 94/B and 94/C

Title	Biological Station File
Language	EMA
Machine	ATLAS I
Purpose	The four programs constitute a complete system for preparing, maintaining and accessing a magnetic tape file of biological station data.

Outline of the system

Biological station and sample data is held on two 1" wide magnetic tapes. One of these tapes is a master file and the other is used for temporary storage prior to updating the master. It is also desirable to keep a further copy of the master tape in case of accident. The tape is divided into about 5000 blocks and each block can hold details of twenty samples and their associated station data. Each block, therefore, normally holds only one station but if there are more than twenty samples then more blocks may be used. The layout of the data within each block is as follows (each block is divided into 512 words each of which can hold one number or four characters):

Word		
0:	Station number	
1:	Marsden square number	
2:	Marsden square letter	
3:	Latitude (degrees)	
4:	Latitude (minutes)	
5:	N or S	
6:	Longitude (degrees)	
7:	Longitude (minutes)	
8:	W or E	
9:	Year	
10:	Month number	
11:	Date	
12:	Time Zone	
13:	Sunrise time	
14:	Sunset time	
15:	Moonrise time	
16:	Moonset time	
17:	Age of moon in days	
18:	Sounding (in metres)	} *
19:	E or M (sounding estimated or measured)	} *
20:	Cruise number	
21-71:	Unused at present; available for other station data	
72:	Sample number	
73-75:	Gear used (12 characters)	
76-77:	Depth range proposed (in metres)	*
78-79:	Method of measuring depth (8 characters)	
80-81:	Actual depth range (in metres)	*
82:	C if the reading in word 81 is the upper limit of net closure	*
83:	Speed of towing (in metres/second)	*
84-85:	Fishing times in local time	*
86:	Length of tow (in nautical miles)	*
87:	Volume of water filtered (in metres $^3 \times 100$)	*
88:	Displacement volume (in c.c.s)	*
89:	S or H (for shot or haul)	*
90-93:	Notes (12 characters)	*

The sequence of words 72-93 may then be repeated up to nineteen further times beginning with word 94 and ending with word 511. * denotes a word which may be filled with an asterisk if no data is available. Blocks containing no data will have -100 in word 0 and if any of the twenty groups of words available for sample data are not used there will be a -100 in the first word (i.e. word 72 or 94 or 116 etc.).

Operation of the system

Program 94/B is first used to initialise both magnetic tapes to be used (i.e. the master tape and the update tape). This places -100 in the first word of each block and zero elsewhere.

The data to be entered on the file is then typed onto special station data forms (white) and at the same time a 7-track paper tape is prepared. Each form must have a block number on it to tell the program where the data is to be stored on the update tape. Program 94 is then used to "copy" the data from paper tape to the update magnetic tape. A directory giving a list of blocks written to and the station numbers in each will be output as will a complete print-out of each station on yellow station data forms. These are then checked against the original hand-written sheets and new forms (and paper tape) prepared for any that contain errors. Program 94 is then used again to overwrite data that was in error.

When the user is satisfied about the accuracy of his data, program 94/C is used to update the master file. All newly acquired data is copied from the update tape to exactly the same position on the master tape. Both tapes now contain the same information but further additions or alterations should always be made first to the update tape and only copied to the master tape when the user is quite satisfied that the data is free from errors and in the correct block.

To access the file (i.e. to obtain a print-out of all or selected stations or to search the file for data fulfilling certain requirements) program 94/A is used. A "Request for Access" form is filled in and at the same time a paper tape produced. The program can then be used to search either magnetic tape and output:

a) a list of station numbers satisfying the required criteria, or

b) all the station data and a selection (or all) of the sample data satisfying the required criteria, and/or

c) a histogram, by days or months, of the stations (or stations containing samples) satisfying the required criteria. (Not available until late 1967.)

The various conditions that can be specified are:

Certain station numbers only (either singly or in blocks);

Stations in a given geographical area only;

Stations within a certain period each year;

Samples taken within certain local times;

Samples taken within a given number of hours either side of sunrise, sunset, moonrise or moonset;

Samples taken within a certain range of depths;

Samples taken with certain gear.

The output will normally be printed on yellow station data forms but lists of station numbers or histograms will be on plain paper.

Format of input documents

Programs 94 and 94/A are held in a compiled form on a magnetic tape, programs 94/B and 94/C being on paper tape. Below is an example of a complete run including writing 50 stations to the update tape and copying them to the master tape, both tapes having been initialised. The access program is for a print-out of up to 20 stations satisfying certain criteria regarding the times of samples. We assume that the magnetic tapes are allocated as follows:

OCEANS LO444 - Master file

OCEANS LO132 - Update tape

SLOAD2 LS169 - Compiled NIO 94 11/4/1967 on file number 0033.

SLOAD2 LS169 - Compiled NIO 94/A 2/5/1967 on file number 0023.

a) Initialising tape LO444. (Similarly for tape LO132)

```
JOB
Job No., * D/L HINDE 96 * INITIALISING MASTER TAPE
INPUT
O S002 NIO PROGRAM 94/B BIO STN FILE INITIALISATION 2/5/1967
OUTPUT
O SEVEN-HOLE PUNCH 1 BLOCK (for monitoring information only)
STORE 15/60 BLOCKS
COMPUTING 5000 INSTRUCTIONS
EXECUTION 8 MINUTES
TAPE
O OCEANS LO444*WRITE PERMIT
***Z
Input O:- Program 94/B 2/5/1967
```

b) Preparation of data on update tape LO132.

```
Input O:-
JOB
Job No., WRITING STNS 5001-5050 TO TAPE
INPUT
1 S002 BIOLOGICAL STATIONS 5001-5050 10/8/1967
OUTPUT
O SEVEN-HOLE PUNCH 26 BLOCKS
STORE 20/30 BLOCKS
COMPUTING 4000 INSTRUCTIONS
EXECUTION 3 MINUTES
TAPE
O OCEANS LO132*WRITE PERMIT
TAPE 99 SLOAD2 LS169*WRITE INHIBIT
COMPILER LOAD
0033/NIO 94 11/4/1967
***Z (or ***C)
Input 1:-
DATA
S002 BIOLOGICAL STATIONS 5001-5050 10/8/1967
(Data)
***Z
```

c) Updating master file on LO444.

Input O:- Program 94/C 23/5/1967

Input 1:-

```
JOB
Job No., * D/L HINDE 98*UPDATING MASTER FILE
INPUT
O SOO2 NIO PROGRAM 94/C BIO STN FILE MASTER UPDATE 23/5/1967
SELF = 1
OUTPUT
O SEVEN-HOLE PUNCH 1 BLOCK
STORE 15/60 BLOCKS
COMPUTING 2500 INSTRUCTIONS
EXECUTION 4 MINUTES
TAPE
O OCEANS LO132*WRITE INHIBIT
TAPE
1 OCEANS LO444*WRITE PERMIT
DATA
1
***Z
```

d) Accessing master file on LO444.

Input O:-

```
JOB
Job No., *D/L HINDE 99*ACCESSING MASTER FILE
INPUT
1 SOO2 SORT ON LOCAL TIMES 17/8/1967
OUTPUT
O SEVEN-HOLE PUNCH 12 BLOCKS
STORE 20/30 BLOCKS
COMPUTING 5000 INSTRUCTIONS
EXECUTION 2 MINUTES
TAPE
O OCEANS LO444*WRITE INHIBIT
TAPE 99 SLOAD2 LS169*WRITE INHIBIT
COMPILER LOAD
OO23/NIO 94/A 2/5/1967
***Z (or ***C)
```

Input 1:-

```
DATA
SOO2 SORT ON LOCAL TIMES 17/8/1967
(Data)
***Z
```

Programmers: Margaret Ringrose and Brian Hinde

N.I.O. PROGRAM 94

Title Preparation of Biological Station File.

Language EMA

Machine ATLAS 1

Purpose To prepare or update from paper tape a magnetic tape file of biological station data.

Inputs

a) Program on paper tape:

0) Program

1) Job description and data

b) Program on magnetic tape:

0) Job description

1) Data

Program

COMPILER EMA

S002 NIO PROGRAM 94 BIO STN FILE PREPARATION 11/4/1967
(or date of appropriate version)

MAIN 1135

AUXILIARY (0, 0)

DUMPS 0

Then follows the program consisting of two chapters ending with

CLOSE

***Z

Job description

The store required for compilation is 65 blocks

The store required for execution is 20 blocks

The number of computing instructions = (1000 + 50N) where N is the number of stations to be written to magnetic tape

The execution time is [(1000 + 50N)/250 + 60] seconds [plus an additional 60 seconds if program is on magnetic tape].

The output specification should be for (1 + N/2) blocks of seven-track tape, the output being on channel 0

One magnetic tape is required (apart from that used to hold the compiled program) and this should be numbered 0.

Data

The data must be punched with the aid of a special form, a copy of which is attached ('Station Data')

Block number: An integer in the range 1 to 5000 specifying the position on the magnetic tape that the data is to occupy.

Station number: An integer in the range 1 to 99999.

Marsden square: An integer specifying the square number followed by the letter a, b, c or d specifying the quadrant in the square.

Latitude and longitude: Degrees specified as an integer and minutes specified with one decimal place. Latitude is then followed by N or S and longitude by W or E.

Date: Year, Month and Day, e.g. 1965 12 29.

Time zone: Positive or negative numbers in the range -12·0 to +12·0 with one decimal place [e.g. G.M.T. will be represented by 0 or 0·0].

Sunrise, sunset, moonrise, moonset: Four-digit integers giving local times (e.g. 1932). If any time is not available put 9999 in that column.

Age of moon: An integer in the range 0 to 28 representing days.

Sounding: An integer giving the sounding in metres, followed by E for "estimated" or M for "measured". If no sounding is available put *(only) in this column.

Cruise number: A positive integer in the range 1 to 999.

Next follows the data for each sample taken at the station as follows:-

Sample number: An integer in the range 1 to 99.

Gear used: 12 characters representing the gear used. Any alphanumeric character may be used plus the special characters / : ' [ ] < > = | ( ) Π α β ½ ? + - & , . and 'space'. Letters may be in upper or lower case and may be underlined if required. The sequence of characters must normally agree exactly with the standard N.I.O. abbreviations for gear but other sequences may be used if required. Only the first ten characters will be used by program 94/A to search for particular gear, so the last two positions may be used for information such as whether a depth telemeter was attached or not. If all the character positions are not required, the remaining ones should be filled with full stops. The first character must never be a space.

Depths proposed: Two integers giving the proposed range of depths, in metres, the deeper reading first. If either value is not available insert "*" in the appropriate column instead.

Method: 8 characters representing the method used to measure the depth range. The character set is as for 'gear used'. If no data is available an asterisk may be inserted in the first character position.

Depths actual: Two integers giving the actual range of depths, in metres, the deeper reading first. If either value is not available insert "*" in the appropriate column instead. The depths are followed by a "C" if the second reading is the upper limit of net closure or else by "*".

Speed in metres/second at which the towing was carried out, printed in the format x.y. Insert "*" if speed is unknown.

Fishing Times: Two four-digit integers giving the times of starting and finishing fishing in local time. If either is not available insert "*" in the appropriate column instead.

Length of Tow in nautical miles in the format x.yy. Insert "*" if the value is not known.

Volume filtered: The volume of water filtered in units of (metres^3 x 100) in the format xx.yy. If the volume has not been measured insert "*".

Displacement: The displacement volume in cubic centimetres in the format xx.yy. If the volume is not available insert "*".

S/H: An "S" or "H" according to whether the readings recorded on this line refer to a shot or haul. If a shot, then the next line must have the same sample number and must contain "H" in the S/H column. Insert "*" if sample was not one of a pair.

Notes: A sequence of 12 characters chosen from the set listed under 'gear used'. The first character must not be a space. If all twelve character positions are not required, the remaining ones should be filled with full stops or spaces. If no notes are required an asterisk followed by eleven spaces, full stops or a tab to the end of the line may be used.

Each sheet of data must end with >> on a newline except the last sheet which should end with // on a newline. / and > must not appear elsewhere in numerical data. Up to twenty lines of sample data may be included on one station sheet. If more than twenty lines are required, continue on a new sheet, specifying the same station number and data, but the next adjacent block number and sample number.

All items on any one line must be separated by at least two consecutive spaces.

e.g. a latitude must be punched as

52ƀƀ16·2ƀƀNƀƀ

a sounding as

2340ƀƀMƀƀ

and a Marsden square as

85ƀƀaƀƀ

Apart from this any convenient formatting may be used and tabs may be set in any position provided there is no backspacing past a position previously reached by a tab. It is suggested that errors are deleted by using () (crossed parentheses) and then continuing on the next available line.

Output

A tape directory will be printed giving the station number corresponding to each block number on the tape. This will be output before anything is written on the magnetic tape. If the data contains no errors, the words END OF STATION will be printed immediately before the data in core store is written to tape. Detected faults include

TOO MUCH DATA	- More than twenty lines of sample data.
FAULT IN DATA	- Spurious characters in numeric data.
NO TERMINATOR FOUND	- Causes the job to be abandoned.

Whenever the data is faulty (except due to the omission of a terminator) the program will abandon the current station and proceed to the next. The last station to be written will be logged with the words END OF DATA. Whenever a fault of any kind occurs, nothing will be written onto the magnetic tape in the current block.

After the tape directory is complete, the magnetic tape will be rewound to the position of the first station and all the data written during the present job will be output. This should be listed on yellow station data forms ready for checking. The form should be inserted in the typewriter with the upper guide mark, 0, located in the printing position. Tabs should be set up in character positions 79, 88 and 98. After each station the typewriter will stop and wait for the operator to change the paper. The last station will be followed by the words END OF DATA.

Restrictions
Maximum number of blocks to be written = 100

Maximum number of lines of sample data = 20 per page

Data restrictions are covered in the 'Data' section above.

Cost
Approximately 5/- per block of magnetic tape written, i.e. 5/- per station when the number of samples per station is less than 21.

Notes
Before this program is run, the magnetic tape to be used must have been initialised using N.I.O. program 94/B.

Programmer
Brian Hinde

0

BLOCK NO....... STATION NO.......

STATION DATA

MARSDEN SQUARE	POSITION Latitude ° '	POSITION Longitude ° '	DATE Year	DATE Month	DATE Day	TIME ZONE	SUNRISE (hours)	SUNSET (hours)	MOONRISE (hours)	MOONSET (hours)	AGE of MOON	SOUNDING (metres)	CRUISE NUMBER

SAMPLE DATA

SAMPLE NUMBER	GEAR USED	DEPTHS (metres) proposed	DEPTHS (metres) method	DEPTHS (metres) actual	SPEED (m/s)	FISHING TIMES from	FISHING TIMES to	LENGTH OF TOW (mins)	VOL. FILT. ($m^3 x 10^2$)	DISPLAC. Vol (cc)	S/H	NOTES
		-	( )	-		-						
		-	( )	-		-						
		-	( )	-		-						
		-	( )	-		-						
		-	( )	-		-						
		-	( )	-		-						
		-	( )	-		-						
		-	( )	-		-						
		-	( )	-		-						
		-	( )	-		-						
		-	( )	-		-						
		-	( )	-		-						
		-	( )	-		-						
		-	( )	-		-						
		-	( )	-		-						
		-	( )	-		-						
		-	( )	-		-						
		-	( )	-		-						
		-	( )	-		-						
		-	( )	-		-						

N.I.O. PROGRAM 94/A

Title: Accessing Biological Station File

Language: EMA

Machine: ATLAS I

Purpose: To search for and print out selected items from the file of biological station data.

Inputs:

a) Program on paper tape:-

   0) Program
   1) Job description and data

b) Program on magnetic tape:-

   0) Job description
   1) Data

Program:

COMPILER EMA

SC02 NIO PROGRAM 94/A BIO STATION FILE ACCESS 2/5/1967 (or date of appropriate version)

MAIN 1000

AUXILIARY (0, 0)

DUMPS 0

Then follows the program, consisting of three chapters, ending with

CLOSE

***Z

Job description: The store required for compilation is 65 blocks. The store required for execution is 20 blocks. The number of computing instructions = (1500 + 50N) where N is the number of stations currently on the file. The execution time is [(1500 + 50N)/250 + 60] seconds [plus an additional 60 seconds if program is on magnetic tape]. The output specification should be for (1 + N'/2) blocks of seven-track tape, the output being on channel 0, where N' is the estimate of the number of stations that will be output. If a list of station numbers only is required, allow (1 + N'/200) blocks. Allow 2 blocks if a histogram is required. One magnetic tape is required (apart from that used to hold the compiled program) and this should be numbered 0.

Data: The data must be punched with the aid of a special form, a copy of which is attached. ('Request for Access to Biological Station File'). The form consists of fourteen options and these are selected as required by typing a 1 in the appropriate box. Options not required should have a 0 (zero) typed in the box. The options 3 to 11 are conjunctive; that is to say, the station will only be selected for printing if all the marked options are satisfied.

1) a) Type '1' if individual stations are required for printing or selection according to options 3 to 11.

   b) An integer, the number of stations required.

   c) The station numbers required.

2) a) Type '1' if blocks of stations are required for printing or selection according to options 3 to 11.

b) An integer, the number of blocks of stations required.

c) The stations required.

e.g. 5010 to 5020

will cause the eleven stations numbered 5010 to 5020 on the file to be extracted for printing or further selection. It is not necessary for all the stations in a requested block to be present on the file. If the whole file is to be printed or searched, one block of stations extending over all the recorded stations, as indicated in the latest tape directory output by program 94, should be requested.

3) a) Type '1' if stations within a certain geographical area are required. The limits should be inclusive and if the latitudes are both S or both N, or the longitudes are both W or both E then the numerically lower value should be typed first in b) and c) below. A longitude range going from W to E or E to W will be taken as being via 0°. If a range straddling 180° is required it must be made the subject of two separate requests.

b) The required latitude limits in the form (e.g.)

$52^{\circ}45.5'$S to $10^{\circ}40.2'$N

c) The required longitude limits in the form (e.g.)

$10^{\circ}00'$W to $15^{\circ}30.5'$E

4) a) Type '1' if stations within a certain period each year are required. The limits should be inclusive.

b) The date limits in the form (e.g.)

Month	Day	Month	Day
8	1	9	30

If the second date is earlier than the first, the period will cover all days between the first date and December 31st and between January 1st and the second date inclusive.

5) a) Type '1' if samples taken only within a certain local time range are required. The limits should be inclusive.

b) The local time limits in the form (e.g.)

0721 to 1932

If the second time is earlier than the first, the period will cover all samples taken between the first time and 2400 hours and between 0001 hours and the second time inclusive.

6) to 9) a) Type '1' if samples taken only within a certain time range of sunrise, sunset, moonrise or moonset are required.

b) The time ranges in the form (e.g.)

-0230 to + 0300 hours

(meaning $-2\frac{1}{2}$ hours to +3 hours; Note that -2 to +3 hours would be interpreted as meaning -2 minutes to +3 minutes!).

10) a) Type '1' if samples taken only within a certain depth range are required. The limits should be inclusive with the deeper value first.

b) The depth range in the form (e.g.)

600 to 450 metres.

11) a) Type '1' if samples taken with particular gear are required.

b) An integer giving the number of types of gear (1 to 6) to be listed under c) below.

c) 12 characters representing the gear used. The codes must correspond exactly to those used in entering gear data during preparation of the file. Only the first ten characters will be used to search for particular gear, so that IKMTCDB....E will be treated exactly as if it were IKMTCDB.....

12) Type '1' if station numbers rather than complete station and sample data are required.

13) Type '1' if a 12 month histogram is required of stations satisfying the stipulated conditions. Item 4 must be zero if a histogram is required.

14) Type '1' if a histogram by days is required of stations satisfying the stipulated conditions. The range of days (maximum 93) must be in 4) above.

Only one 'request for access' form may be input on any run. All items on any one line must be separated by at least two consecutive spaces. Tab may be used providing there is no back-spacing past a position previously reached by a tab.

Output

The output will normally consist of the station and sample details in the format described in program 94. A list of station numbers, if requested, will have the numbers printed one to a line. Histograms will be printed with their x-axis down the page and the scales clearly indicated. Errors in the request form will be indicated by the words 'FAULT IN SPECIFICATION' followed by a value of the index K to show where the fault occurred. K will always equal the item number where the fault occurred except when the fault is one of format (e.g. only one space between the two depths in item 10) when the item number may equal (K-1). Each sheet of output will be followed by a stop code to allow the operator to change the paper in the typewriter.

<u>Restrictions</u> Any parameter outside the normally accepted range (e.g. a station number that does not yet exist, or a date of 32nd May) will be accepted but no false information will be produced.

<u>Cost</u> Approximately 56/- plus 1/6 per station output plus N/3/= where N is the number of stations currently held on the file. e.g. To search a file of 500 stations and output 20 would cost about £12.12s. if all the stations had to be examined.

<u>Programmer</u> Margaret Ringrose

NATIONAL INSTITUTE OF OCEANOGRAPHY

Request for Access to Biological Station File

Mark required options with 1, all others with 0.

1) Read selected individual stations from file [ ]

Total number of stations required:

List of stations required:

2) Read selected blocks of stations from file [ ]

Number of blocks required:

List of stations required:

to

to

to

to

3) Sort and read all stations within the following area [ ]

Latitudes . ' to . '

Longitudes . ' to . '

4) Sort and read all stations within the following

period each year [ ]

Month Day Month Day

to

5) Sort and read all samples within the following

local times [ ]

to

Sort and read all samples within the following periods

6) over sunrise [ ] - to + hours

7) over sunset [ ] - to + hours

8) over moonrise [ ] - to + hours

9) over moonset [ ] - to + hours

10) Sort and read all samples within the following

depth range [ ]

to metres (deepest first)

11) Sort and read all samples using the following gear [ ]

Total number of gear required:

Gear: [ ]

[ ]

[ ]

[ ]

[ ]

[ ]

[ ]

12) Print station numbers only [ ]

13) Print histogram by months [ ] (Item 4 must be 0)

14) Print histogram by days [ ] (Max. 93 days in 4 above)

Signed ....

Date ....

N.I.O. PROGRAM 94/B

Title	Initialising Biological Station File
Language	EMA
Machine	ATLAS I
Purpose	To initialise an Ampex 1" magnetic tape prior to using it for storing biological station data.
Inputs	Job description 0) Program
Program	COMPILER EMA S002 NIO PROGRAM 94/B BIO STATION FILE INITIALISATION 2/5/1967 (or date of appropriate version) MAIN 512 AUXILIARY (0,0) DUMPS 0 Then follows the program, consisting of one chapter, ending with CLOSE ***Z
Job description	The store required for compilation is 60 blocks. The store required for execution is 15 blocks. The number of computing instructions used is 5000. The execution time is 8 minutes. No output is generated but one magnetic tape is required and this must be numbered 0.
Restrictions	If the tape is not long enough to hold 5000 blocks of data, the words END TAPE plus a post-mortem will be output.
Method	The program writes -100 into the first word of blocks 1 to 5000 inclusive and zero into all other words.
Cost	£6.10s.0d.
Programmer	Margaret Ringrose

N.I.O. PROGRAM 94/C

Title	Updating Master Biological Station File
Language	EMA
Machine	ATLAS I
Purpose	To update from magnetic tape a master magnetic tape file of biological station data.
Inputs	0) Program 1) Job description and data

Program

COMPILER EMA

S002 NIO PROGRAM 94/C BIO STATION FILE MASTER UPDATE 23/5/1967
(or date of appropriate version)

MAIN 512

AUXILIARY (0, 0)

DUMPS 0

Then follows the program, consisting of one chapter, ending with

CLOSE

***Z

Job description

The store required for compilation is 60 blocks.
The store required for execution is 15 blocks.
The number of computing instructions used is 200 + 5 N where N is the number of new blocks to be added to the file.
The execution time is 4 minutes.
The output specification on channel 0 should be for 50 lines or 1 block of paper tape.
Two magnetic tapes are required as follows:-

Tape 0: Tape containing new data.
Tape 1: Master tape file to be updated.

Data

The data consists of one positive integer (I), the block number on each tape file at which updating is to begin.

Output

The output consists of one line as follows:-

MASTER TAPE UPDATED FROM BLOCK x TO BLOCK y INCLUSIVE

where x = I and y = (I + N - 1).

Restrictions

The data on magnetic tape 0 need not be on consecutive blocks but there must not be more than 10 blocks not used between any two blocks filled with data. All unused blocks must have -100 in their first word.

Cost

Approximately £3. 0s. 0d. for 50 stations.

Notes

Both magnetic tapes used by this program must have been initialised by N.I.O. program 94/B before they were first used.

Programmer

Margaret Ringrose

N.I.O. PROGRAM 95

Title — Determination of position from Decca coordinates (BALLARIN).

Language — EMA

Machine — ATLAS 1

Purpose — To compute the geographical position from the two Decca readings of a Decca chain having a common master and two slaves.

Inputs
0) Program
1) Job description and data

Program — The program starts

```
COMPILER EMA

S002 NIO 95 POSITION FROM DECCA COORDINATES (BALLARIN),
     26/7/1967 (or date of appropriate version)
```

and ends

```
CLOSE

***Z
```

Job description and data

In the job description, the number of lines of output = (number of Decca pairs for which position is required + number of captions + 100),

the store required = 15/70 blocks,
the number of computing instructions = 700 + 8 (number of positions to be computed) + 8 (number of Decca chains).

The job description is followed by

DATA

1) DATA TITLE
2) one line of general description of the data
3) geographical position of master station in degrees and minutes (with decimals). N and W are +ve, S and E are -ve
4) geographical position of first slave
5) geographical position of second slave
6) reciprocal of comparison frequency wavelength (master - slave 1) in metres^{-1}
7) reciprocal of comparison frequency wavelength (master - slave 2) in metres^{-1}
8) / (solidus)
9) DATA TITLE
10) one line of specific comment on the data to follow
11) Decca readings of first slave
12) Decca readings of second slave

Items 11) and 12) may be repeated as often as required, i.e. any number of pairs of Decca readings may be input.

A new specific comment may be introduced by repeating items 8) and 9).

A new Decca chain may be introduced by following item 12) by * (asterisk) and returning to item 1). The last set of data should be followed by > and ***Z.

Output

General data title
Specific data title
Headed columns as follows:-

- slave 1 coordinates
- slave 2 "
- geographical position, i.e. latitude and longitude
- error - should be zero

If a second geographical position appears on the same line this means that the two hyperbolae defined by the pair of Decca readings have a second intersection within $\frac{1}{2}$ radian (about 1700 miles) of the master. The choice between the two positions then depends on the user's knowledge of approximate position.

Method

This is as described by BALLARIN in "The International Hydrographic Review" vol. XXVII, 2, p.31-50. It involves transferring coordinates on a spheroid to coordinates on a sphere. The spheroid used is HAYFORD'S.

Note

The common station of the three must be the master.

Programmer

JAMES CREASE

N.I.O. PROGRAM 95/A

Title: Determination of position from Decca coordinates (BALLARIN).

Language: EMA

Machine: ATLAS 1

Purpose: To compute a table of geographical positions from the Decca readings of a Decca chain having a common master and two slaves.

Inputs:
0) Program
1) Job description and data

Program: The program starts

COMPILER EMA

S002 NIO 95/A POSITION FROM DECCA COORDINATES (BALLARIN)
21/3/67 (or date of appropriate version)

and ends

CLOSE

***Z

Job description and data

In the job description, the number of lines of output = (number of Decca pairs for which position is required + number of captions + 100),

the store required = 15/70 blocks,
the number of computing instructions = 700 + 8 (number of positions to be computed) + 8 (number of Decca chains);

for long jobs the second term can be reduced to be 4 (number of positions to be computed).

The job description is followed by

DATA

1) DATA TITLE

2) one line of general description of the data

3) geographical position of master station in degrees and minutes (with decimals). N and W are +ve, S and E are -ve

4) geographical position of first slave

5) geographical position of second slave

6) reciprocal of comparison frequency wavelength (master - slave 1) in metres^{-1}

7) reciprocal of comparison frequency wavelength (master - slave 2) in metres^{-1}

8) / (solidus)

9) DATA TITLE

10) one line of specific comment on the data to follow

11) First Decca reading (in lane numbers) of first pair, followed by the increment (+ve or -ve) for successive readings, followed by the number of increments. Lane numbers are numerical starting from 0 at master (i.e. the zone letters must be converted to lane numbers).

12) Second Decca reading (in lane numbers) of second pair, followed by the increment (+ve or -ve) for successive readings, followed by the number of increments. Lane numbers are numerical starting from 0 at master (i.e. the zone letters must be converted to lane numbers).

Items 11) and 12) may be repeated as often as required, i.e. any number of pairs of Decca readings may be input.

A new specific comment may be introduced by repeating items 8) and 9).

A new Decca chain may be introduced by following item 12) by * (asterisk) and returning to item 1). The last set of data should be followed by > and ***Z.

Output

General data title
Specific data title
Headed columns as follows:-

slave 1 coordinates
slave 2 "
geographical position, i.e. latitude and longitude
error - should be zero

If a second geographical position appears on the same line this means that the two hyperbolae defined by the pair of Decca readings have a second intersection within $\frac{1}{2}$ radian (about 1700 miles) of the master. The choice between the two positions then depends on the user's knowledge of approximate position.

Method

This is as described by BALLARIN in "The International Hydrographic Review" vol. XXVII, 2, p.31-50. It involves transferring coordinates on a spheroid to coordinates on a sphere. The spheroid used is HAYFORD'S.

Note

The common station of the three must be the master.

Programmer

James Crease

N.I.O. PROGRAM 96

Title — Decoding Plessey Current Meter tapes.

Language — EMA

Machine — ATLAS I

Purpose — To convert data from a pseudo-binary form produced on 8-track paper tape by the Plessey magnetic tape/paper tape translator into B.C.D. suitable as input for Atlas. Conversion factors are applied, errors are flagged and the subsequent results are output graphically.

Inputs — 0) Program
1) Job description and parameters, 8-track data, terminator

Program

COMPILER EMA

S002 NIO PROGRAM 96 CURRENT METER DECODING 7/7/1967
(or date of appropriate version)

MAIN 110

AUXILIARY (0,0)

DUMPS 0

Then follows the program ending with

***Z

Job description and data (3 tapes)

Tape 1 (7-track)

JOB

Job No., Job Title

INPUT

0 S002 NIO PROGRAM 96 CURRENT METER DECODING 7/7/1967

SELF = 1

OUTPUT

0 LINEPRINTER n LINES

OUTPUT 1 TAPE x/LINEPRINTER n LINES

OCEANS IO402* WRITE PERMIT   Output tape title

STORE 25/60 BLOCKS

COMPUTING m INSTRUCTIONS

where n = number of samples to be decoded

x = starting block on the output tape

and m = 1000 + 8 n instructions

Then follows the parameters and data:-

DATA

DATA TITLE

One line of title

An integer (A) the meter number.

B, C, the calibration constants for speed and direction (see Method).

An integer (W) the number of units in the dead space + 1023.

U, a constant to be added to the calibrated speed.

Runout
***P
Runout
***E
Runout

2nd Tape (8-track)

The 8-track data to be decoded, ending with at least 7 blank characters.

The data is assumed to be in the form of 4 numbers between markers, each consisting of two characters. The values of these characters are:-

1st character value	Bit Position	2nd character value
32	1	1
64	2	2
128	3	4
256	4	8
parity	5	parity
512	6	16
0	7	0
marker	8	not used

3rd Tape (7-track)

Runout
***Z
Runout

Output

Output stream 0:-

Data title

Meter number, calibration constants, and number of units in dead space.

For each set of readings:

Meter number on the data tape, if not the same as that read from the parameters.

Line count.

Rev. counter reading and speed of current in cm/sec unless previous reading was faulty.

Direction in degrees.

Should the rev. counter have passed over the dead space between the present reading and the preceding one, a D is printed to denote this.

The speed and direction are then plotted against time, + being used to denote speed and * to denote direction. The output ends with the words "END OF DATA".

Output stream 1:-

Meter number.

For each set of readings:

Line count, speed and direction.

Restrictions Number of characters between markers must not exceed 100.

Failures

1) If the number of characters between markers on the 8-track tape does not equal 8, a negative line count will be output on both streams and the words:-

   "NUMBER OF CHARACTERS BETWEEN MARKERS = n"

   will be output on output stream 0, where n = the number of characters.

   On the next correct line of output no speed will be printed as this is derived from differencing the present and preceding rev. count.

2) If a parity error occurs on the 8-track input tape, a P is output in the corresponding column of results on output stream 0.

   Whenever either of the above faults occurs spaces are output on stream 1.

Cost About 1½d per set of readings (meter number, speed and direction).

Method

The 8-track data is read in binary using Atlas Routine 811, until the first marker is encountered. (1) The character is then tested for parity errors, and if correct, 128 is subtracted from it, and the next 8 characters are read, each of which is tested for the presence of a marker code or parity error. If the eighth character is not a marker, the data is read until a marker is encountered, and a fault line is output, as is the case if a marker occurs before the eighth character.

Using the original marker and the next 7 characters, the binary values are then paired, the first being multiplied by 32 and added to the second, so that 4 B.C.D. values now remain.

The first value is compared with the meter number read in, and only printed out if it differs. The second value is ignored, as it is a dummy reading. The fourth value is subtracted from the preceding fourth value, (unless the preceding value was in error, in which case the present value is stored) the result is multiplied by a calibration factor (B) and a constant (U) is added, to give the speed in cm/sec. Should the result of the differencing be negative, a constant (W) is added to the previous value, and the differencing is repeated. (This constant consists of 1023 plus the number of units necessary to allow for the dead space.) The original value is then stored until the process is repeated on the next set of readings.

The third value is multiplied by a constant (C) to give the direction of the current.

The results are then output on both streams, with graphs plotted on output stream 0. For this the maximum speed is taken to be 200 cm/sec (with other versions of the program this value may differ), and anything faster than this is taken to be 200 cm/sec for plotting purposes only. If both speed and direction should be the same, the speed will be plotted first, with the direction plotted in the next space. The method is then repeated from (1), using the original ninth character (marker) as the first character until the program encounters 7 consecutive blank characters on the data tape, when the job is terminated.

Programmer Margaret Ringrose

N.I.O. PROGRAM 96/A

Title — Decoding Plessey meter tapes of temperature and depth.

Language — EMA

Machine — ATLAS I

Purpose — To convert data from a pseudo-binary form produced on 8-track paper tape by the Plessey magnetic tape/paper tape translator into B.C.D. suitable as input for Atlas. Conversion factors are applied, errors are flagged and the subsequent results are output graphically.

Inputs — The same as PROGRAM 96 except the program heading and the data parameters vary slightly, and there is no output 1.

The program heading is

S002 NIO PROGRAM 96/A PLESSEY METER DECODING 10/10/1967

The data parameters are:-

An integer, A, the meter number

B, C, the calibration constants for temperature and depth (see method of Program 96).

E, F, the maximum values to which graphs of temperature and depth are to be scaled.

Output — Data title

Meter number, calibration constants, the maximum values to which graph of temperature and depth are scaled.

For each set of readings:-

Meter number on the data tape if different from that read from the parameter A.

Line count

Temperature

Depth

The temperature and depth are plotted against time + denoting temperature and * denoting depth

The output ends with

'END OF DATA'

Failures — As for Program 96.

Cost — Less than $1\frac{1}{2}$d per line.

Method — The 8-track data is read in binary using Atlas Routine 811, until the first marker is encountered (1) The character is then tested for parity errors, if correct 128 is subtracted from it and the next eight characters are read, each of which are tested for the presence of a marker code or a parity error. If the eighth character is not a marker, the data is read until a marker is encountered and a fault line is output as is the case if a marker occurs before the eighth character.

Using the original marker and the next seven characters the binary values are then paired, the first being multiplied by 32 and added to the second, so that 4 B.C.D. values now remain.

The first value is compared with the meter number read in and printed out if it differs. The second value is multiplied by a calibration factor (B) to give temperature. The third value is multiplied by a calibration factor (C) to give depth, and the fourth value is ignored as it is a dummy reading.

The results are then output, together with graphs of temperature and depth against time. The maximum limits of the graph for temperature and depth are (E) and (F) respectively, and any value larger than the limit is taken to equal the limit for plotting purposes.

The method is repeated from (1) using the original ninth character (marker) as the first character until the program encounters 7 consecutive blank characters on the data tape. When the tape is terminated.

Programmer   Margaret Ringrose

N.I.O. PROGRAM 96/B

Title — Decoding Bergen meter tapes of temperature, current speed and direction.

Language — EMA and ATLAS I MACHINE CODE.

Machine — ATLAS I

Purpose — To convert data from a pseudo-binary form produced on 7-track tape by a modified Plessey magnetic tape/paper tape translator into a B.C.D. code suitable as input for Atlas. Conversion factors are applied, errors are flagged and the subsequent results are output graphically and to computer-compatible magnetic tape.

Inputs — 0) Program
1) Job description, parameters and data

Program

COMPILER EMA

SO02 NIO PROGRAM 96/B BERGEN METER DECODING 27/11/1967
(or date of appropriate version)

MAIN 120

AUXILIARY (0, 0)

DUMPS 0

Then follows the program ending with

***Z

Job description and data

JOB

Job No., Job title

INPUT

0 SO02 NIO PROGRAM 96/B BERGEN METER DECODING 27/11/1967

SELF = 1

OUTPUT

0 LINEPRINTER (100 + n) LINES

OUTPUT 1

TAPE x/SEVEN-HOLE PUNCH (1 + n/50) BLOCKS
OCEANS LO402*WRITE PERMIT (Output tape title)

STORE 25/65 BLOCKS

COMPUTING (1500 + 10n) INSTRUCTIONS

where n = number of samples to be decoded
x = starting block on the output tape

DATA

DATA TITLE
One line of title

An integer (A) the meter number; an integer (T) the meter number tolerance

B, C, H the calibration constants for speed, direction and temperature (see Method) (cm/sec. per digit, degrees per digit and °C per digit respectively).

E, F the maximum points to be plotted on the temperature and speed scale (°C and cms/sec.).

W, the number of units in the dead space + 1023

U, V constants to be added to the calibrated speed and direction (cms/sec. and degrees).

A '+' or '-' depending on whether successive speed readings are normally increasing or decreasing.
***P

Runout

***B (if data to follow on same tape)

***F (if data to follow on a separate tape)

Pseudo-binary data, ending with at least 12 blank characters and no terminator.

The data is assumed to be in the form of blocks of 6 two-character numbers whose values are:

1st character value	Bit position	2nd character value
32	1	1
64	2	2
128	3	4
256	4	8
Parity	5	Parity
512	6	16
0	7	0

The recording sequence is:

Channel	1	Meter number
"	2	Temperature
"	3	Not used
"	4	Not used
"	5	Direction
"	6	Speed

Output stream 0 (Normally lineprinter):-

Data title
Meter number and tolerance, calibration equations and scales of the three graphs.

For each block of readings:

Meter number on the data tape
Line count
Rev. counter reading
Speed of current in cm/sec. unless the previous reading was faulty
Direction in degrees
Temperature in °C.

Should the rev. counter have passed over the dead space between the present reading and the preceding one, a D. is printed to denote this.

The speed, direction and temperature are then plotted against time, + being used to denote speed, * to denote direction and o to denote temperature. The output ends with the words "END OF DATA".

Output stream 1 (Normally magnetic tape):-

Meter Number on line 1
For each block of readings:
Line count, speed, direction and temperature

Restrictions

Number of blocks < 10,000 for correct graph layout.

T < 25.

Failures

1) If the number of characters in a block is not equal to 12, a negative line count will be output on both streams and

WRONG NUMBER OF CHARACTERS IN BLOCK

will be output on stream 0. On the next correct line of output no speed will be printed as this is derived from differencing adjacent rev. counts.

2) If a parity error occurs in the data, a P is output in the corresponding column of results on output stream 0.

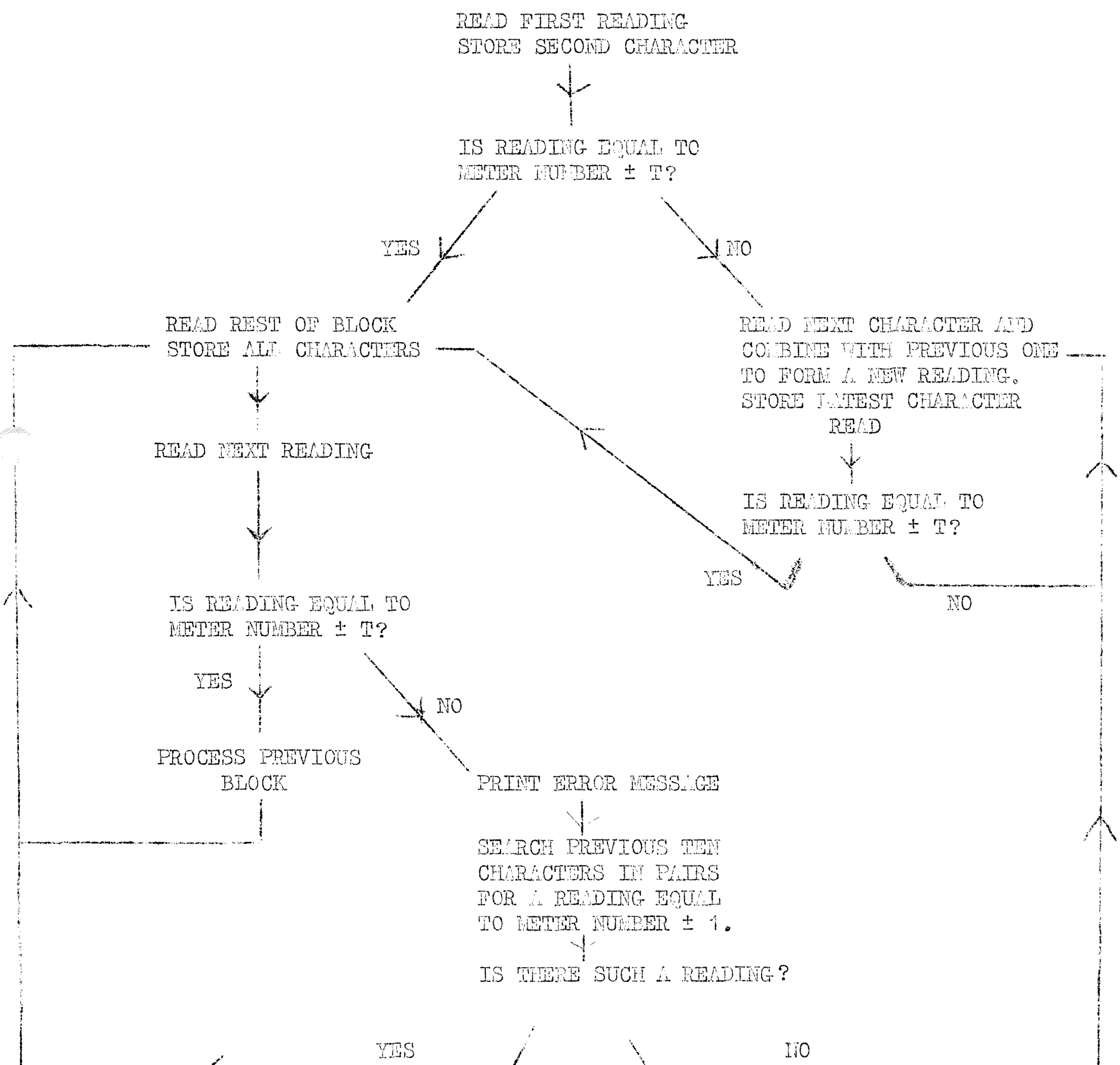
Whenever either of the above faults occurs, spaces are output on stream 1. The magnetic tape may later be corrected using NIO program 98.

Cost

About 1½d per block of readings.

Method

The logic of the decoding part of the program is illustrated below; (N.B. 1 character = 1 row on the tape, 1 reading = 2 rows on the tape, 1 block = 12 rows on the tape, "READ" as used below includes checking for parity and decoding).



A check is made for blank characters and 12 consecutive blanks indicate the end of the data.

The calibrations are applied as follows:-

Speed cms/sec. = $B(N_1 - N_2) + U$

where N_1 and N_2 are succeeding rev. counter readings. If $N_1 - N_2 < 0$ a constant (W) is added to N_1. This constant consists of 1023 plus the number of units necessary to allow for the dead space.

Direction, degrees = $CN_D + V$

Temperature, °C = HN_T

where N_D and N_T are the binary readings of direction and temperature.

The results are then output on both streams and a line count is added to facilitate later editing. Graphs are plotted on output 0. For these the maximum speed is taken to be F cms/sec., the maximum direction to be 360° and the maximum temperature E °C. Any values above the maximum are plotted as maximum values. If any two (or three) readings require to be plotted in the same character position the order of printing will be speed (in its correct position) followed by direction, then temperature.

Notes — The program uses the Atlas machine code routine R811.

Programmer — Elizabeth Palethorpe.

N.I.O. PROGRAMME -1

Title Spectral analysis

Language EMA

Machine ATLAS 1

Purpose To compute the variances, normalised auto-covariances and auto-spectra of 1-7 time series; also, for two or more series, to compute the normalised cross-covariances, co- and quad-spectra, phase lags, intensities and coherences for the series taken in pairs. All possible combinations of series will be used.

Compatibility The programme needs a chapter O to read in the data or, alternatively, it can be added to an existing program. Output routines are included but the output is also available in the auxiliary store for further use if required.

Structure of programme

The programme consists of chapters 1 and 2 with the heading

PROGRAMME -1

CHAPTER 1

The programme ends with

CLOSE

There are several lines of explanatory comment preceding the heading in some versions of the programme.

Parameters to be set before entry

K, the numbers of series to be used.
L, the number of covariance lags required.
M, the number of frequencies required.
N, the number of terms/series.
E, the sampling interval between successive terms of the same series.
D' ($\geq$ K (2M + 6)), the starting location of the series in the auxiliary store.

The auxiliary store should contain in locations O to (K - 1) the K means of the series.

The conversion factors from digital to physical units (see Method) should be in K to (2K - 1).

The time lags of each series relative to the first, in decimal fractions of the time between successive terms of the same series should be in 3K to (4K - 1) (the first number will normally be zero).

The amplitude response factors (a_s^x) at each frequency and for each series arranged sequentially for each of (M + 1) frequencies should be in 4K to ((M + 5)K -1) (see Method)

The phase factors of each series (ϕ_s^x) arranged sequentially for each of (M + 1) frequencies should be in ((M + 5)K) to ((2M + 6)K -1).

In D' onwards there should be the NK terms of the K series arranged sequentially (i.e. the first term of each series followed by the second term of each series).

Indices and variables used by the programme

I, J, K, L, M, N, P, Q, R, S,
I', J', K', L', M', N', P', Q', R', S',
A, B, C, D, E, G, H, U, V, W, X, Y, Z,
A', B', C', D', X',

Entry — Entry is at label 8), the first instruction in chapter 1.

Store and computing instructions.

The store required for compilation = 65 blocks

The store required for execution will be 15 + main store required + auxiliary store required (see below)

The computing instructions used will be less than 1800 for compilation and $(3200 + K^2NL/80)$ for execution of each set of data.

Different versions of the programme will be needed for varying data requirements. The directives (and hence the total main store requirement) should be altered as follows:

Chapter 1: $X \geq \dot{K}\ (L + 1)$, $8\ (2L + 1)$, and 1000
$Y \geq KK\ (L + 1)$

Chapter 2: $Y \geq KK\ (L + 1)$
$Z \geq KK\ (M + 1)$
$U, V > K$
$F, G \geq K\ (M + 1)$

The total auxiliary store should be greater than $D' + (L + M + 2)\ KK$ or $D' + NK$, whichever is the greater; D' must be at least $K\ (2M + 6)$.

Output — $150 + L + M + 2L\ (\text{INTPT}\ K/2) + (M + 6)\ (K(K - 1)/2)$ lines are required for each set of data. The output is as follows:-

Number of terms in each series.

Time lags (decimal fractions of time between successive terms of the same series).

X, the means of the series.

X', the means converted to physical units.

V_x, the variances of the series.

V_x', the variances converted to physical units.

Ψ_{xx}, the lagged normalised auto-covariances (preceded by the value r) for $r = 0(1)L$, where L is the number of lags specified.

V_{xy}, V_{xy}', the un-normalised cross-covariances at zero lag, firstly in computing units and secondly in physical units.

Ψ_{xy}, the lagged normalised cross-covariances for each pair of series (preceded by the value r) for $r = -L(1)L$.

The frequency increment between successive spectral estimates, in cycles/time unit.

E_{xx}, auto-spectra, in (physical units)^2/cycle/time unit (preceded by s, where $s = 0(1)M$).

Cross-spectra for each pair of series, preceded by s and consisting of:-

E_{xy}, E_{xy}^{*}, co- and quad-spectra (in physical units^2/cycle/ time unit).

I, intensity.

ϕ, phase (in degrees).

γ^2, coherence.

On exit from the programme the auxiliary store contains:-

means in locations	O........(K -1),
conversion factors in locations	K........(2K -1),
standard deviations in locations	2K.......(3K -1),
time lags in locations	3K.......(4K -1),
covariances in locations in the form	D'.......(D'+(KK(L+1)-1))

$R_{11}(O)$......$R_{K1}(O)$......$R_{1K}(O)$......$R_{KK}(O)$
....
$R_{11}(L)$......$R_{K1}(L)$......$R_{1K}(L)$......$R_{KK}(L)$,

and auto, co- and quad-spectra in (D'+KK(L+1))...... ......(D'+(KK(L+M+2)-1)) in the form

$S_{11}(O)$ $C_{12}(O)$...................$C_{1K}(O)$
$Q_{12}(O)$ $S_{22}(O)$ $C_{23}(O)$..........$C_{2K}(O)$
$Q_{13}(O)$ $Q_{23}(O)$ $S_{33}(O)$................
....
$Q_{1K}(O)$...............................$S_{KK}(O)$
$S_{11}(1)$ $C_{12}(1)$.......................
....
......................................$S_{KK}(M)$

Restrictions

$1 \leqslant K \leqslant 7$
$1 \leqslant M \leqslant L \leqslant N/2$
$N < 10{,}000$

Method

Let the first series of data x_i be $x_0, x_1, x_2, x_3 \ldots x_{N-1}$,

and the second series of data y_i be

$y_0, y_1, y_2, \ldots\ldots y_{N-1}$, etc.

Let F, G, etc., be the respective conversion factors from computing to physical units, and if the mean

$X = \frac{1}{N} \sum_{i=0}^{N-1} x_i$ for each series of N terms has been calculated,

then $X' = FX$

Variances:

$$V_x = \frac{1}{N-1} \sum_{i=0}^{N-1} x_i'^2$$

where $x_i' = x_i - X$, then $V_x' = F^2 V_x$

Lagged Normalised auto-covariances:

$$\Psi_{xx}(r) = \left[(N-r-1)\ V_x\right]^{-1} \sum_{i=0}^{N-r-1} x_{i+r}'\ x_i'$$

Cross-covariances:

$$V_{xy} = \frac{1}{N-1} \sum_{i=0}^{N-1} x_i'\ y_i'$$

$$V_{xy}' = FG\ V_{xy}$$

Lagged Normalised cross-covariances:

Negative lags:-

$$\Psi_{xy}(r) = \left[(N-r-1)\sqrt{V_x V_y}\right]^{-1} \sum_{i=0}^{N-r-1} x_{i+r}'y_i'$$

Positive lags:-

$$\Psi_{yx}(r) = \left[(N-r-1)\sqrt{V_x V_y}\right]^{-1} \sum_{i=0}^{N-r-1} x_i'\ y_{i+r}'$$

Frequency increment between successive spectral estimates is
0·5/LE cycles/time unit (e.g., c/sec.)

i.e., 0·5/L cycles/time interval (E).

Auto-spectra:

$$E_{xx}(s) = 4TV_x' \sum_{r=0}^{L}{}'' \phi_{xx}(r) \cos\left(\frac{rs\pi}{L}\right)$$

where $\sum''$ means the sum with the first and last terms

halved, and $\phi_{xx}(r) = \Psi_{xx}(r) \cos^2\left(\frac{\pi r}{2L}\right)$

Cross-spectra:

Co-spectra:

$$E_{xy}(s) = 4TFG\sqrt{V_x V_y} \sum_{r=0}^{L}{}'' \tfrac{1}{2}\left[\phi_{yx}(r) + \phi_{xy}(r)\right] \cos\left(\frac{rs\pi}{L}\right)$$

where $\phi_{yx}(r) = \phi_{yx}(r) \cos^2\left(\frac{\pi r}{2L}\right)$

and $\phi_{xy}(r)$ is similarly defined.

Quad-spectra:

$$E_{xy}^*(s) = 4TFG\sqrt{V_x V_y} \sum_{r=0}^{L}{}'' \frac{1}{2}\left[\phi_{yx}(r) - \phi_{xy}(r)\right] \sin\left(\frac{rs\pi}{L}\right)$$

The Cross-spectrum is then defined as $E_{xy} + iE_{xy}^*$

Intensity: $I = \sqrt{E_{xy}^2 + E_{xy}^{*2}}$

Phase lag: $\phi = \tan^{-1} \dfrac{E_{xy}^*}{E_{xy}}$

If ϕ is positive series x leads y in time.

Coherence: $\gamma^2 = \dfrac{I^2}{E_{xx}E_{yy}}$

Corrections for time lags between series

If there exists a time lag xT between corresponding terms of series n and series 1 (n being an integer with values from 2 to 7 inclusive, and x being a decimal having values between 0 and 1 inclusive), then the cross-spectral estimates $E_{xy}(s)$ and $E_{xy}^*(s)$ are modified as follows to $F_{xy}(s)$ and $F_{xy}^*(s)$ where:-

$$F_{xy}(s) = E_{xy}(s) \cos\left(\frac{\pi sx}{L}\right) - E_{xy}^*(s) \sin\left(\frac{\pi sx}{L}\right)$$

$$\text{and } F_{xy}^*(s) = -E_{xy}^*(s) \cos\left(\frac{\pi sx}{L}\right) - E_{xy}(s) \sin\left(\frac{\pi sx}{L}\right)$$

Then the computations of intensity, phase lag and coherence proceed using $F_{xy}(s)$ and $F_{xy}^*(s)$ in place of $E_{xy}(s)$ and $E_{xy}^*(s)$.

Corrections for amplitude and phase response of recording system

If the recording system of series x has a response to unit input at frequency s/2LE cycles/time unit of

$$a_s^x \exp\left(\frac{i\pi\theta_s^x}{180}\right)$$

and series y has a response

$$a_s^y \exp\left(\frac{i\pi\theta_s^y}{180}\right) \qquad (\theta \text{ in degrees})$$

then the auto-spectral estimate, $E_{xx}(s)$, will be divided by $(a_s^x)^2$, the cross-spectral estimates, $E_{xy}(s)$ and $E_{xy}^*(s)$, will be divided by $(a_s^x)(a_s^y)$ and the phase lag, ϕ, will be replaced by $\phi + (\theta_s^x - \theta_s^y)$.

Programmers James Crease and Brian Hinde

N.I.O. PROGRAMME -2

Title Spectral analysis with correction for pre-whitening.

Language EMA

Machine ATLAS 1

Purpose To compute the variances, normalised auto-covariances and auto-spectra of 1-7 pre-whitened time series; also, for two or more series, to compute the normalised cross-covariances, co- and quad-spectra, phase lags, intensities and coherences for the series taken in pairs. All possible combinations of series will be used.

Compatibility The programme needs a chapter 0 to read in the data or, alternatively, it can be added to an existing program. Output routines are included but the output is also available in the auxiliary store for further use if required.

Structure of programme

The programme consists of chapter 1 and 2 with the heading

PROGRAMME -2

CHAPTER 1

The programme ends with

CLOSE

There are several lines of explanatory comment preceding the heading in some versions of the programme.

Parameters to be set before entry

K, the numbers of series to be used.
L, the number of covariance lags required.
M, the number of frequencies required.
N, the number of terms/series.
E, the sampling interval between successive terms of the same series.
D′ ($\geq K(2M + 6)$) the starting location of the series in the auxiliary store.
Y′, the pre-whitening factor (see Method).

The auxiliary store should contain in locations 0 to (K-1) the K means of the series.

The conversion factors from digital to physical units (see Method) should be in K to (2K - 1).

The time lags of each series relative to the first, in decimal fractions of the time between successive terms of the same series should be in 3K to (4K -1) (the first number will normally be zero).

The amplitude response factors (a_s^x) at each frequency and for each series arranged sequentially for each of (M+1) frequencies should be in 4K to ((M+5)K-1) (see Method).

The phase factors of each series (θ_s^x) arranged sequentially for each of (M+1) frequencies should be in ((M+5)K) to ((2M+6)K - 1).

In D′ onwards there should be the (N-1)K terms of the series arranged sequentially (i.e. the first term of each series followed by the second term of each series, after pre-whitening).

Indices and variables used by the programme

I, J, K, L, M, N, P, Q, R, S,
I′, J′, K′, L′, M′, N′, P′, Q′, R′, S′,
A, B, C, D, E, G, H, U, V, W, X, Y, Z,
A′, B′, C′, D′, X′, Y′, Z′.

Entry — Entry is at label 8), the first instruction in chapter 1.

Store and computing instructions

The store required for compilation = 65 blocks.

The store required for execution will be 15 + main store required + auxiliary store required (see below).

The computing instructions used will be less than 1800 for compilation and $(3200 + K^2NL/80)$ for execution of each set of data.

Different versions of the programme will be needed for varying data requirements. The directives (and hence the total main store requirement) should be altered as follows:

Chapter 1:	$X \geqslant K(L+1)$, $8(2L+1)$ and 1000.
	$Y \geqslant KK(L+1)$
Chapter 2:	$Y \geqslant KK(L+1)$
	$Z \geqslant KK(M+1)$
	$U, V > K$
	$F, G \geqslant K(M+1)$

The total auxiliary store should be greater than $D' + (L+M+2)\,KK$ or $D' + MK$, whichever is the greater; D' must be at least $K(2M+6)$.

Output — 150+L+2M+2L (INTPT K/2) + (2M+6) (K(K-1)/2) lines are required for each set of data. The output is as follows:-

Number of terms in each series.

Times lags (decimal fractions of time between successive terms of the same series).

X, the means of the series.

X′, the means converted to physical units.

V_x, the variances of the series.

V_x', the variances converted to physical units.

Ψ_{xx}, the lagged normalised auto-covariances (preceded by the value r) for r = 0(1)L, where L is the number of lags specified.

V_{xy}, V_{xy}', the un-normalised cross-covariances at zero lag, firstly in computing units and secondly in physical units.

Ψ_{xy}, the lagged normalised cross-covariances for each pair of series (preceded by the value r) for r = -L(1)L.

The frequency increment between successive spectral estimates, in cycles/time unit.

E_{xx}, E'_{xx}, auto-spectra, in (physical units)^2/cycle/time unit (preceded by s, where s = 0(1)M), first for the pre-whitened data and then after correcting for the pre-whitening factor.

Cross-spectra for each pair of series, preceded by s and consisting of:-

E_{xy}, E_{xy}^{*}, E'_{xy}, $E'_{xy}{}^{*}$, co- and quad-spectra (in physical units^2/cycle/time unit) both uncorrected and corrected.

I, uncorrected intensity.

ϕ, phase (in degrees).

γ^2, coherence.

The corrected values will not be printed for any frequency where the reciprocal of the attenuation factor due to pre-whitening is greater than 10^{10}.

On exit from the programme the auxiliary store contains:-

means in locations	0	.....(K-1),
conversion factors in locations	K	.....(2K-1),
standard deviations in locations	2K	....(3K-1),
time lags in locations	3K	....(4K-1),
covariances in locations	D′	....(D′+(KK(L+1)-1))

in the form

$$\begin{matrix} R_{11}(0) \ldots\ldots R_{K1}(0) \ldots\ldots R_{1K}(0) \ldots\ldots R_{KK}(0) \\ \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \\ R_{11}(L) \ldots\ldots R_{K1}(L) \ldots\ldots R_{1K}(L) \ldots\ldots R_{KK}(L) \end{matrix}$$

and uncorrected auto, co- and quad-spectra in (D′+KK(L+1))....... (D′+(KK(L+M+2)-1)) in the form

$$\begin{matrix} S_{11}(0) & C_{12}(0) & \ldots\ldots\ldots & C_{1K}(0) \\ Q_{12}(0) & C_{22}(0) & C_{23}(0) \ldots & C_{2K}(0) \\ Q_{13}(0) & Q_{23}(0) & S_{33}(0) \ldots & \ldots \\ \ldots & \ldots & \ldots & \ldots \\ Q_{1K}(0) & \ldots & \ldots & S_{KK}(0) \\ S_{11}(1) & C_{12}(1) & \ldots & \ldots \\ \ldots & \ldots & \ldots & \ldots \\ \ldots & \ldots & \ldots & S_{KK}(M) \end{matrix}$$

Restrictions $1 \leqslant K \leqslant 7$
$1 \leqslant M \leqslant L \leqslant N/2$
$N < 10{,}000$

Method Let the first series of data x_i be $x_0, x_1, x_2, x_3 \ldots x_{N-1}$, and the second series of data y_i be $y_0, y_1, y_2, y_3 \ldots$ $\ldots y_{N-1}$, etc.

Let F, G, etc. be the respective conversion factors from computing to physical units and if pre-whitening has been carried out using

$$x_i = x_i + C\,x_{i+1}$$

and the means have been calculated

$$X = \frac{1}{N-1} \sum_{i=0}^{N-2} x_i \text{ for each series of N terms}$$

then $X' = FX$

Variances:

$$V_x = \frac{1}{N-2} \sum_{i=0}^{N-2} x_i'^2$$

where $x_i' = x_i - X$, then $V_x' = F^2 V_x$

Lagged Normalised auto-covariances:

$$\Psi_{xx}(r) = \left[(N-r-2) V_x \right]^{-1} \sum_{i=0}^{N-r-2} x_{i+r}' \, x_i'$$

Cross-covariances:

$$V_{xy} = \frac{1}{N-2} \sum_{i=0}^{N-2} x_i' \, y_i'$$

$$V_{xy}' = FG \, V_{xy}$$

Lagged normalised cross-covariances:

Negative lags:-

$$\Psi_{xy}(r) = \left[(N-r-2) \sqrt{V_x V_y} \right]^{-1} \sum_{i=0}^{N-r-2} x_{i+r}' \, y_i'$$

Positive lags:-

$$\Psi_{yx}(r) = \left[(N-r-2) \sqrt{V_x V_y} \right]^{-1} \sum_{i=0}^{N-r-2} x_i' \, y_{i+r}'$$

Frequency increment between successive spectral estimates is

$0{\cdot}5/LE$ cycles/time unit (e.g., c/sec.)

i.e., $0{\cdot}5/L$ cycles/time interval (E).

Auto-spectra:

$$E_{xx}(s) = 4TV_x' \sum_{r=0}^{L}{}'' \phi_{xx}(r) \cos\left(\frac{rs\pi}{L}\right)$$

where $\sum''$ means the sum with the first and last terms halved,

and $\phi_{xx}(r) = \psi_{xx}(r) \cos^2\left(\frac{\pi r}{2L}\right)$

$$E'_{xx}(s) = E_{xx}(s) \Big/ \left[1 + C^2 \mp 2C \cos\left(\frac{\pi s}{M}\right) \right]$$

Cross-spectra:

Co-spectra:

$$E_{xy}(s) = 4TFG\sqrt{V_x V_y} \sum_{r=0}^{L}{}'' \tfrac{1}{2}\left[\phi_{yx}(r) + \phi_{xy}(r)\right] \cos\left(\frac{rs\pi}{L}\right)$$

where $\phi_{yx}(r) = \psi_{yx}(r) \cos^2\left(\frac{\pi r}{2L}\right)$

and $\phi_{xy}(r)$ is similarly defined.

$$E'_{xy}(s) = E_{xy}(s) \Big/ \left[1 + C^2 + 2C \cos\left(\frac{\pi s}{M}\right)\right]$$

Quad-spectra:

$$E_{xy}^{*}(s) = 4TFG\sqrt{V_x V_y} \sum_{r=0}^{L}{}'' \tfrac{1}{2}\left[\phi_{yx}(r) - \phi_{xy}(r)\right] \sin\left(\frac{rs\pi}{L}\right)$$

$$E'^{*}_{xy}(s) = E_{xy}^{*}(s) \Big/ \left[1 + C^2 + 2C \cos\left(\frac{\pi s}{M}\right)\right]$$

The cross-spectrum is then defined as $E_{xy} + iE_{xy}^{*}$

Intensity: $I = \sqrt{E_{xy}^2 + E_{xy}^{*2}}$

Phase lag: $\phi = \tan^{-1} \dfrac{E^{*}_{xy}}{E_{xy}}$

If ϕ is positive series x leads series y in time.

Coherence: $\gamma^2 = \dfrac{I^2}{E_{xx}E_{yy}}$

Corrections for time lags between series

If there exists a time lag xT between corresponding terms of series n and series 1 (n being an integer with values from 2 to 7 inclusive, and x being a decimal having values between 0 and 1 inclusive), then the cross spectral estimates $E_{xy}(s)$ and $E_{xy}^{*}(s)$ are modified as follows to $F_{xy}(s)$ and $F_{xy}^{*}(s)$ where:-

$$F_{xy}(s) = E_{xy}(s) \cos\left(\frac{\pi sx}{L}\right) - E_{xy}^{*}(s) \sin\left(\frac{\pi sx}{L}\right)$$

$$\text{and } F_{xy}^{*}(s) = - E_{xy}^{*}(s) \cos\left(\frac{\pi sx}{L}\right) - E_{xy}(s) \sin\left(\frac{\pi sx}{L}\right)$$

Then the computations of intensity, phase lag and coherence proceed using $F_{xy}(s)$ and $F_{xy}^{*}(s)$ in place of $E_{xy}(s)$ and $E_{xy}^{*}(s)$.

Corrections for amplitude and phase response of recording system

If the recording system of series x has a response to unit input at frequency s/2LE cycles/time unit of

$$a_s^x \exp\left(\frac{i\pi\theta_s^x}{180}\right)$$

and series y has a response

$$a_s^y \exp\left(\frac{i\pi\theta_s^y}{180}\right) \qquad (\theta \text{ in degrees})$$

then the auto-spectral estimate, $E_{xx}(s)$, will be divided by $(a_s^x)^2$, the cross-spectral estimates, $E_{xy}(s)$ and $E_{xy}*(s)$, will be divided by $(a_s^x)(a_s^y)$ and the phase lag, ϕ, will be replaced by $\phi + (\theta_s^x - \theta_s^y)$.

<u>Programmers</u> James Crease and Brian Hinde

N.I.O. PROGRAMME -3

Title — Analysis of pitch, roll, heave buoy data

Language — EMA

Machine — Atlas I

Purpose — To compute the angular harmonics and directional spectrum from the auto, co- and quad-spectra between pitch, roll and heave.

Compatibility — The programme is designed for use with programmes -1 or -2 and can thus be used as an addition to N.I.O. programs 67, 74 or 82. It can be used on its own if a reading chapter is added.

Structure of programme

The programme consists of a chapter 1 only with the headings

PROGRAMME -3

>> Comments

CHAPTER 1

The programme ends with

CLOSE

The number of main variables required = 5000
The number of auxiliary variables required = C′+10(M+1)+1
where M is an integer to be set to represent the highest frequency required (e.g. 50 for frequencies 0(0·01) 0·5c/s), and C is to be set before entry also.

Parameters to be set before entry

M as described above
C′ = Starting location in the auxiliary store of the auto, co- and quad-spectra between heave (series 1), pitch (series 2) and roll (series 3).

The series should be arranged as follows:

$S_{11}(0)$	$C_{12}(0)$	$C_{13}(0)$
$Q_{12}(0)$	$S_{22}(0)$	$C_{23}(0)$
$Q_{13}(0)$	$Q_{23}(0)$	$S_{33}(0)$
$S_{11}(1)$	etc. to	

$S_{33}(M)$ in C′+9(M+1)-1

Then should follow the doubly integrated acceleration spectrum

$E_{11}(0)$ ... ... $E_{11}(M)$ ending at C′+10(M+1)

All the spectral estimates should be in comparable physical units (e.g. co-spectra in $deg^2/c/sec$ and doubly integrated acceleration spectra in $ft^2/c/sec$).

Indices and variables used by the programme

I, K, K′, M, S, S′

A, B, C, C′, D, F, G, H, U, V, X, Y, Z

Entry — Entry is at label 1), the first instruction in Chapter 1.

Store and computing instructions

The main store required for compilation = 60

The main store required for execution = 15 + C′/512
The number of computing instructions is less than 700 for compilation and less than 50M for execution.

Output

The output is suitable for lineprinter or seven-track paper tape and requires (100 + 10M) lines or 2 + int. part [M/4] blocks.

For each value of S from 1 to M is given the values of a_1, a_2, a_3, a_4, b_1, b_2, b_3, b_4 and C (see Method)

For each value of S the angular distribution of energy (normalised) is given at 15° intervals where the angles are the directions of propagation of energy.

The directional spectrum of energy is given at 15° intervals and at each value of S where the angles are the directions of arrival of energy.

Restrictions

$M \leqslant 100$

Method

The various auto, co- and quad-spectra are combined as follows:

$$a_1 = -Q_{12}/S_{11}$$
$$a_2 = (S_{22}-S_{33})/(S_{22}+S_{33})$$
$$b_1 = -Q_{13}/S_{11}$$
$$b_2 = 2C_{23}/(S_{22}+S_{33})$$

at each frequency σ (corresponding to values of S)

Then $C_1 = +\sqrt{a_1^2 + b_1^2}$

$$C_2 = +\sqrt{a_2^2 + b_2^2}$$

$$S_1 = C_1/(1-C_1)$$

$$S_2 = \left(1+3C_2 + \sqrt{1+14C_2 + C_2^2}\right)\Big/2\,(1- \;{}_2)$$

and $C = (S_1+S_2)/2$

Then $\begin{matrix} a_3 \\ b_3 \end{matrix} = \dfrac{C(C-1)(C-2)}{(C+1)(C+2)(C+3)} \begin{matrix} \cos 3\bar{\theta} \\ \sin 3\bar{\theta} \end{matrix}$

where $\bar{\theta} = \tan^{-1} b_1/a_1$

and $\begin{matrix} a_4 \\ b_4 \end{matrix} = \dfrac{C(C-1)(C-2)(C-3)}{(C+1)(C+2)(C+3)(C+4)} \begin{matrix} \cos 4\bar{\theta} \\ \sin 4\bar{\theta} \end{matrix}$

The angular distribution is then evaluated:-

$$G_S(\phi) = \frac{1}{\Pi}\left[0{\cdot}50 + \left\{\sum_1^4 W_n\,(a_n \cos\phi + b_n \sin\phi)\right\}\right]$$

where
$W_1 = 0{\cdot}8889$
$W_2 = 0{\cdot}6222$
$W_3 = 0{\cdot}3394$
$W_4 = 0{\cdot}1414$

for $\phi = 0°, 15°, 30° \ldots\ldots\ldots\ldots\ldots 345°$

Finally the directional spectrum is evaluated:-

$$F(\sigma,\phi) = E(\sigma)\; G_S(\phi')$$

where $E(\sigma)$ is the energy of the doubly integrated heave and where

$$\phi' = \phi + 180°$$

Notes

It will be noticed that the values of C_{12}, C_{13} and Q_{23} at all frequencies and all the values of energy at frequency 0 are not used in the calculation. They need not, therefore, be input to the auxiliary store. They are referred to only for compatibility with N.I.O. programmes -1 and -2.

Programmer

BRIAN HINDE

N.I.O. PROGRAMME -4

Title Graph plotting

Language EMA

Machine ATLAS 1

Purpose To scale and linearly plot a series of points on the line-printer or typewriter.

Compatibility The programme is designed for use as an addition to any existing program, and should normally be the last part of that program to be executed.

Structure of programme

The programme consists of a chapter 1 only, with the headings

```
PROGRAMME -4
>> GRAPH PLOTTING
>> comments
CHAPTER 1
```

The programme ends with

```
CLOSE
```

The number of main variables required $\geq 4I + 8$
The number of auxiliary variables required $\geq 2I$
where I = number of points to be plotted.

Parameters to be set before entry

I, number of points to be plotted.
E′, starting location in the auxiliary store of the numbers to be plotted.
The numbers should be arranged as follows:-

$X_1, X_2, X_3 \ldots\ldots\ldots\ldots\ldots XI$

$Y_1, Y_2, Y_3 \ldots\ldots\ldots\ldots\ldots YI$

T = 1 if a printout of the sorted data is required
= 0 if no printout of the sorted data is required

S = 1 if no scaling of points along the y-axis is required
= 0 if scaling of points along the y-axis is required

S′ = 1 if no scaling of points along the x-axis is required
= 0 if scaling of points along the x-axis is required

N = 1 if the x-axis is to go through the minimum value of y if the origin is not within the range of the values of y
= 0 if the x-axis is to go through the y-origin

N′ = 1 if the y-axis is to go through the minimum value of x if the origin is not within the range of the values of x
= 0 if the y-axis is to go through the x-origin

Indices and variables used by the programme

I, M′, N, N′, R′, S, S′, T

A, A′, B, B′, C, C′, D, D′, E′, F, F′, G, G′, Z, Z′

Entry Entry is at label 1), the first instruction of chapter 1

Store and computing instructions

The main store required for compilation = 60 blocks
The main store required for execution = 25 blocks
The computing instructions requested should be 5000

Output

The output is suitable for lineprinter or seven-track paper tape and requires (50 + I + (A - B)) lines or (50 + I + (A - B))/50 blocks, where I = number of points to be plotted, A = maximum value of x, and B = minimum value of x.

Restrictions

$I \leq 400$
Scaling will occur on y-axis, even if none is requested if the difference between the maximum and minimum values of y is greater than 118.

Method

The programme scales the values along the y axis so that $D^* \ (A - B) \leq 118$, where D = the spacing interval, A = the maximum value of y, and B the minimum value. If all the values are positive and a displaced origin is not required, B is set to equal O. If scaling along the x-axis is required, $D'^* \ (A - B) \leq 200$

C and C'; the scaling intervals are set equal to O - B, so that if either or both are negative the corresponding origin is displaced.

The values of y are then sorted into numerical order, working from minimum to maximum. The corresponding values of x are put into the same order as the values of y. The process is then repeated for sorting the values of x into numerical order.

The scale along the y axis is computed by the equation $Z = 10/A'$ where A' is the spacing interval, and printed out. The scale is then computed for the x-axis by the equation $Z' = 6/B'$ where B' is the spacing interval.

The graph is then plotted. The y-axis is marked by a series of minus signs with 0 above it where the y-origin occurs. If the first point occurs on the y-axis, the axis is printed up to that point, the point is plotted, and the axis line continues. The x-axis is marked with a vertical line at the start of each new line, whether or not there is a point to be plotted on that line. 0 beside the x-axis indicates where the x-origin occurs.

After each point has been plotted, the number of new lines between that value of x and the succeeding value of x is output, followed by the number of spaces between the value of y at the x-axis and the value of y to be plotted. If two values of y occur for the same value of x, there is no new line, but only the number of spaces between the new value of y and the preceding one.

Programmer

Margaret Ringrose

Data Processing Group

N.I.O. PROGRAMME -5, Change of specification

The description of programme -5 should be amended as follows:-

1) p.1 lines 37-40 inclusive should be deleted. At present they read:

"P, an integer equal to the number of cross-covariance and/or cross-spectral pairs required. (= 0 when H^1 = 1, 4, 5)
P = -1 may be used to signify that all possible combinations are required."

2) p.2 lines 4-9 should be altered to read:

"The cross-covariance and cross-spectra options should be in ((2M + 6)K) to ((K/2) (4M + K + 11) -1). The order in store is series

1,2 1,3 .... 1,K
2,3 2,4 .... 2,K
....(K-1), K

A 1 in the first location indicates that 1,2 is required; a 2 in the second location that 1,3 is required, and so on until K(K-1)/2 in the last location indicates that (K-1), K is required. Locations at which the options are not required should be set equal to -100.

B. J. Hinde

18th August, 1967

DPG/P/1

N.9

N.I.O. PROGRAMME -5

Title General spectral analysis

Language EMA

Machine ATLAS 1

Purpose To compute the covariances, auto- and cross-spectra for up to 99 time series. Facilities include selection of series for analysis, pre-whitening, corrections for amplitude and phase response of the recording system and entry at various points in the calculations.

Compatibility
The programme needs a chapter 0 to read in the data or it can be incorporated into an existing program. Output routines are included but the output is also available in the auxiliary store for further use if required.

Structure of programme

The programme consists of chapters 1 and 2 with the heading

PROGRAMME -5

>> GENERAL SPECTRAL ANALYSIS

The programme ends with

CLOSE

Parameters to be set before entry

K, the number of series to be used,

L, the number of covariance lags required,

M, the number of frequencies required (= 0 when H′ = 1,2,3),

N, the number of terms per series,

E, the sampling interval between successive terms of the same series,

D′ (≥ (K/2)(4M + K + 11)), the starting location of the series in the auxiliary store,

F, the pre-whitening factor (= 0 for no pre-whitening),

H′ an integer having values between 1 and 15, this being the total of up to four integers chosen from the following:-

1 if auto-covariances are required

2 if cross-covariances are required

4 if auto-spectra are required

8 if cross-spectra are required

P, an integer equal to the number of cross-covariance and/or cross-spectral pairs required, (= 0 when H′ = 1,4,5)

P = -1 may be used to signify that all possible combinations are required.

The auxiliary store should contain in locations 0 to (K - 1) the K means of the series.

The conversion factors from digital to physical units should be in K to (2K - 1) (Replace by standard deviations in 2K to (3K - 1) when H′ = 4,8,12).

The time lags, of each series relative to the first, in decimal fractions of the time between successive terms of the same series should be in 3K to (4K - 1).

The amplitude response factors (a_s^x) at each frequency and for each series arranged sequentially for each of the (M + 1) frequencies should be in 4K to ((M + 5)K - 1) (Omit for H′ = 1,2,3).

The phase factors of each series (θ_s^x arranged sequentially for each of the (M + 1) frequencies should be in ((M + 5)K) to ((2M + 6)K - 1) (Omit when H′ = 1,2,3)

The cross-covariance and cross-spectra options should be in ((2M + 6)K) to ((K/2)(4M + K + 11) - 1). A 1 in store indicates that the option is required. The order in store is series 1,2 1,3...............1,K
2,3 2,4..............2,K
..................(K - 1),K

In D′ onwards there should be the N K (or (N - 1)K if pre-whitened) terms of the K series arranged sequentially (i.e. the first term of each series followed by the second term of each series, etc.), if H′ $\neq$ 4,8,12.

If H′ = 4,8,12 in D′ onwards there should be auto- and cross-covariances in the order:

$R_{11}(O)$ $R_{21}(O)$.....$R_{K1}(O)$ $R_{12}(O)$.....$R_{K2}(O)$.....$R_{KK}(O)$
....
$R_{11}(L)$..$R_{KK}(L)$

Indices and variables used by the programme

All unprimed indices except O.

All primed indices except Q′, R′, S′ and T′ .

All unprimed special variables.

All primed special variables except E′,F′,G′,U′,V′,W′ and Π′ .

Entry Entry is at label 8), the first instruction in chapter 1.

Store and computing instructions

The store required for compilation = 65 blocks.

The store required for execution = 17 + main store + auxiliary store (see below).

The computing instructions used will be less than 2000 for compilation and ($3000 + KNL/80 + P^2 K^2NL/80$) for execution of each set of data where P = fraction of total possible options required (e.g. 8/15 for 8 out of 15 possible cross-covariance pairs). For H′ = 4,8,12 the formula becomes ($3000 + 10K^2 L$).

Different versions of the programme will be needed for varying data requirements. The directives (and hence the total main store requirement) should be set as follows:-

Chapter 1: $X \geq K(L + 1)$, $8(2L + 1)$, 1000
$Y \geq K^2(L + 1)$
$A,C,U,V,W,\Pi \geq K$
$B \geq K(K - 1)/2$

Chapter 2: $Y \geq K^2(L + 1)$
$Z \geq K^2(M + 1)$
$B \geq K(K - 1)/2$
$F,G \geq K(M + 1)$
$U,V \geq K$

4 main stores are required in chapters 1 and 2 for index stores.

The total auxiliary store should be greater than

D′ + (L + M + 2)KK or D′ + NK, whichever is the greater;
D′ must be at least ((K/2)(4M + K + 11)).

Output 200 +

$L(1 + INTPT(K - 1)/7)$ lines for auto-covariances

$M(1 + INTPT(K - 1)/7)$ lines for auto-spectra

$2L(1 + INTPT(KP(KP - 1)/14)$ lines for cross-covariances

and $(M + 6)(KP(KP - 1)/2)$ lines for cross-spectra for each set of data. The spectral outputs will be doubled if pre-whitening has been used. (P = fraction of total possible options required).

	Restrictions
The output consists of:-	
N, the number of terms in each series.	$H' \neq 4,8,12$
The time lags of each series.	$H' \neq 4,8,12$
X, the means of each series.	$H' \neq 4,8,12$
X', the means converted to physical units.	$H' \neq 4,8,12$
V_x, the variances of the series.	$H' \neq 4,8,12$
V_x', the variances converted to physical units.	$H' \neq 4,8,12$
ψ_{xx}, the lagged normalised auto-covariances for $r = 0(1)L$	H' odd
V_{xy}, V'_{xy}, the un-normalised cross-covariances at zero lag, followed by the values in physical units.	$H' = 2,3,6,7,10,11,14,15$
ψ_{xy}, the lagged normalised cross-covariances for each pair of series for $r = -L(1)L$.	$H' = 2,3,6,7,10,11,14,15$
The frequency increment between successive spectral estimates.	
E_{xx}, the auto-spectra, in physical units, for $s = 0(1)M$.	$H' = 4-7, 12-15$
E'_{xx}, the auto-spectra corrected for pre-whitening.	$H' = 4-7, 12-15$ $F \neq 0$
Cross-spectra as specified, each consisting of:-	$H' \geq 8$
E_{xy}, E^*_{xy}, co- and quad-spectra in physical units.	
E'_{xy}, E'^*_{xy}, corrected co- and quad-spectra.	$F \neq 0$
I, intensity.	
I', intensity corrected for pre-whitening.	$F \neq 0$
ϕ, phase (in degrees).	
γ^2, coherence.	

The corrected values will not be printed for any frequency where the reciprocal of the attenuation factor due to pre-whitening is greater than 10^{10}.

On exit from the programme, the auxiliary store contains:-

means in locations	$0.......(K - 1)$	$H' \neq 4,8,12$
conversion factors in	$K.......(2K - 1)$	$H' \neq 4,8,12$
standard deviations in	$2K.....(3K - 1)$	
time lags in	$3K.....(4K - 1)$	
covariances in	$D'.....(D' + (K^2(L + 1) - 1))$	

in the form

$$R_{11}(0).....R_{K1}(0).....R_{1K}(0).....R_{KK}(0)$$

$$.......................................$$

$$R_{11}(L).....R_{K1}(L).....R_{1K}(L).....R_{KK}(L)$$

and auto-, co- and quad-spectra in

$(D' + K^2(L + 1))\ldots\ldots(D' + (K^2(L + M + 2) - 1))$

in the form

$S_{11}(O)\ C_{12}(O)\ldots\ldots\ldots\ldots\ldots\ldots C_{1K}(O)$

$Q_{12}(O)\ \ S_{22}(O)\ \ C_{23}(O)\ldots\ldots\ldots\ldots C_{2K}(O)$

$\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots$

$Q_{1K}(O)\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots S_{KK}(O)$

$S_{11}(1)\ \ C_{12}(1)\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots$

$\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots S_{KK}(M)$

These values will all be corrected for pre-whitening unless the reciprocal of the attenuation factor is greater than 10^{10}.

Restrictions

$K < 100$

$M \leqslant L < 1000$

$M = O$ when $H' = 1, 2$ or 3

$N \leqslant 100{,}000$

$-1 \leqslant F \leqslant 1$

$H' \leqslant 15$

$-1 \leqslant P \leqslant K(K - 1)/2$

$K^2(M + 1) \leqslant 40{,}000$

Method

Let the first series of data be x_i $[i = O(1)N-1]$ and the second series of data be y_i, etc.

Let G_x, G_y, etc. be the respective conversion factors from digital to physical units.

Let the mean of any one series be represented by X digital units and hence $G_x X$ in physical units.

Variances:

$$V_x = \frac{1}{N-1}\sum_{i=O}^{N-1} x_i'^2$$

where $x_i' = x_i - X$.

Then $V_x' = G_x^2 V_x$

Lagged normalised auto-covariances:

$$\psi_{xx}(r) = \left[(N-r-1)V_x\right]^{-1}\sum_{i=O}^{N-r-1} x'_{i+r}\ x_i'$$

Cross-covariances:

$$V_{xy} = \frac{1}{N-1}\sum_{i=O}^{N-1} x_i'\ y_i'$$

$$V'_{xy} = G_x G_y V_{xy}$$

Lagged normalised cross-covariances:

Negative lags:-

$$\psi_{xy}(r) = \left[(N-r-1)\sqrt{V_x V_y} \right]^{-1} \sum_{i=0}^{N-r-1} x'_{i+r}\, y'_i$$

Positive lags:-

$$\psi_{yx}(r) = \left[(N-r-1)\sqrt{V_x V_y} \right]^{-1} \sum_{i=0}^{N-r-1} x'_i\, y'_{i+r}$$

Frequency increment between successive spectral estimates is 0·5/LE cycles/time unit (e.g. c/sec.)

Auto-spectra:

$$E_{xx}(s) = 4EV_x' \sum_{r=0}^{L}{}'' \phi_{xx}(r) \cos\left(\frac{rs\pi}{L}\right)$$

where $\sum''$ means the sum with the first and last terms halved,

and $\phi_{xx}(r) = \psi_{xx}(r) \cos^2\left(\frac{\pi r}{2L}\right)$.

If requested, the auto-spectra, corrected for pre-whitening, is

$$E'_{xx}(s) = E_{xx}(s) \Big/ \left[1+F^2+2F \cos\left(\frac{\pi s}{M}\right) \right].$$

Cross-spectra:

$$\text{Co-spectra, } E_{xy}(s) = 4EG_x G_y \sqrt{V_x V_y} \sum_{r=0}^{L}{}'' \tfrac{1}{2}\left[\phi_{yx}(r)+\phi_{xy}(r) \right] \times \cos\left(\frac{rs\pi}{L}\right)$$

where $\phi_{yx}(r) = \psi_{yx}(r) \cos^2\left(\frac{\pi r}{2L}\right)$

and $\phi_{xy}(r) = \psi_{xy}(r) \cos^2\left(\frac{\pi r}{2L}\right)$.

$$E'_{xy}(s) = E_{xy}(s) \Big/ \left[1+F^2+2F \cos\left(\frac{\pi s}{M}\right) \right]$$

$$\text{Quad-spectra, } E^*_{xy}(s) = 4EG_x G_y \sqrt{V_x V_y} \sum_{r=0}^{L}{}'' \tfrac{1}{2}\left[\phi_{yx}(r)-\phi_{xy}(r) \right] \times \sin\left(\frac{rs\pi}{L}\right)$$

$$E^{*\prime}_{xy}(s) = E^{*}_{xy}(s) \Big/ \left[1+F^2+2F \cos\left(\frac{\pi s}{M}\right) \right]$$

The cross-spectrum is defined as $(E_{xy}+iE^{*}_{xy})$ or $(E'_{xy}+iE^{*\prime}_{xy})$.

Intensity, $I = \sqrt{E^{2}_{xy}+E^{*2}_{xy}}$; $I' = I \Big/ \left[1+F^2+2F \cos\left(\frac{\pi s}{M}\right) \right]$

Phase lag, $\phi = \tan^{-1} \dfrac{E^{*}_{xy}}{E_{xy}}$; if ϕ is positive, series x

leads series y in time.

Coherence, $\gamma^2 = I^2 \Big/ \left(E_{xx} \; E_{yy}\right)$

Corrections for time lags between series

If there exists a time lag xE between corresponding terms of series n and series 1, where x is a decimal fraction having values between 0 and 1, then the cross-spectral estimates $E_{xy}(s)$ and $E^{*}_{xy}(s)$ are modified as follows to $F_{xy}(s)$ and $F^{*}_{xy}(s)$ where:-

$$F_{xy}(s) = E_{xy}(s) \cos\left(\frac{\pi sx}{L}\right) - E^{*}_{xy}(s) \sin\left(\frac{\pi sx}{L}\right)$$

and $F^{*}_{xy}(s) = -E^{*}_{xy}(s) \cos\left(\frac{\pi sx}{L}\right) - E_{xy}(s) \sin\left(\frac{\pi sx}{L}\right)$.

The computations proceed using $F_{xy}(s)$ and $F^{*}_{xy}(s)$ in place of $E_{xy}(s)$ and $E^{*}_{xy}(s)$.

Corrections for amplitude and phase response of recording system

If the recording system of series x has a response to unit input at s/2LE cycles/time unit of

$$a^{x}_{s} \exp\left(\frac{i\pi\theta^{y}_{s}}{180}\right)$$

and series y has a response

$$a^{y}_{s} \exp\left(\frac{i\pi\theta^{y}_{s}}{180}\right) \quad [\theta \text{ in degrees}]$$

then the auto-spectral estimates $E_{xx}(s)$ will be divided by $(a^{x}_{s})^2$, the cross-spectral estimates $E_{xy}(s)$ and $E^{*}_{xy}(s)$ will be divided by $a^{x}_{s} \; a^{y}_{s}$ and the phase lag ϕ will be replaced by $\phi + (\theta^{x}_{s} - \theta^{y}_{s})$.

Notes: This programme combines programmes -1 and -2 and extends them to cover larger values of K,L,M and N, and also includes optional selection of only certain pairs of series for cross-correlation. In addition, the facility to skip certain parts of the programme is available.

Programmers: James Crease
Brian Hinde