Idpac Commands

User's Guide

Wieslander, Johan

1980

Document Version:
Publisher's PDF, also known as Version of record

CODEN: LUTFD2/(TFRT-3157)/1-108/(1980)

Idpac Commands

- User´s Guide

JOHAN WIESLANDER

INSTITUTIONEN FÖR REGLERTEKNIK
LUNDS TEKNISKA HÖGSKOLA
1980

IDPAC COMMANDS
-
User's Guide

Johan Wieslander

Organization
LUND INSTITUTE OF TECHNOLOGY
Department of Automatic Control
Box 725
S-220 07 Lund 7 SWEDEN

Document name
Project report STU

Date of issue
February 1980

CODEN: LUTFD2/(TFRT-3157)/1-108/(1980)

Author(s)
Johan Wieslander

Sponsoring organization
Swedish Board for Technical Development (STU), contract No 77-3548

Title and subtitle

Idpac Commands - User's Guide

A4 A5

Abstract

Idpac is an interactive program, command oriented with a powerful macro facility. The program is aimed at data analysis, mainly using spectral analysis and identification techniques. The identification commands allow parametric model building using the Least Squares or the Maximum Likelihood methods. Model analysis and simulation is used to test the results.

Key words

A4 A5

Classification system and/or index terms (if any)

Supplementary bibliographical information

Language
English

ISSN and key title

ISBN

Recipient's notes

Number of pages
108

Price

Security classification

Distribution by (name and address)

DOKUMENTDATABLAD enl SIS 61 41 21

Idpac Commands - User's Guide

This is the guide to the commands in the program Idpac. It will give information on how to call the different commands, what their functions are and what method is used. In many cases hints and examples are included.

Idpac is an interactive command oriented program package. It is aimed at problems encountered in analysis of time series. A time series is the consecutive values taken by a certain signal as a function of time. Typical sources of the time series would be recorded measurements and control signals from industrial processes. Examples where Idpac has been applied includes paper and pulp industy, power systems, chemical industry and ship building. Non-industrial applications are found e.g. in biomedicine and econometrics.

The types of questions Idpac may answer are:
a) What do the signals look like?
b) What is the size of the disturbancies?
c) Where can we find them?
d) What are their properties?
e) How can the process be described?
f) How does the process react to specific inputs?

In order to answer such questions, Idpac contains modules for data conversion, scaling and calibration. A comprehensive plotting facility is also included in order to visualize the data.

The analysis tools available in Idpac are:
a) Correlation analysis.
b) Spectral analysis (Fourier transform of correlation function as well as DFT).
c) Parametric model fitting (maximum likelihood and least squares method).
d) Normality and independence tests.
e) Simulation

Idpac is a member of a family of program packages with identical data structures and user interaction. The interaction is primarily intended to satisfy the needs of the knowledgable user, but mechanisms exist to adapt the interaction to other demands. Further reading on the interaction and the general use of these programs is found in the two reports:

1. J. Wieslander, H. Elmqvist: INTRAC - A Communication Module for Interactive Programs - Language Manual, TFRT-7132, Dept. of Automatic Control, Lund Institute of Technology, Lund, Sweden.

2. J. Wieslander: Interactive Programs for Computer Aided Analysis and Design of Control Systems - General Guide, TFRT-3156, Dept. of Automatic Control, Lund Institute of Technology, Lund, Sweden.

These programs were developed with financial support from the Swedish Board for Technical Development, contracts 73-3553, 75-3776 and 77-3548. They represent the combined effort of many people at the depertment over many years.

Commands Available in Idpac

The following is a structured list of the commands available in Idpac, together with a short indication of their use.

1. Input & Output

CONV - Conversion of data into internal standard form
EDIT - Symbolic text editor
FORMAT - Conversion of data into symbolic (external) form
LIST - Output of data on user readable form
MOVE - Moving data in the data base

2. Graphic Output

BODE - Draw curves in a diagram with logarithmic scales
HCOPY - Take a hard copy of the last graphic output
PLMAG - Draw a magnified plot and allow changes
PLOT - Draw curves with linear scales

3. Time Series Operations

ACOF - Compute autocorrelation functions
CCOF - Compute crosscorrelation functions
CONC - Concatenate time series
CUT - Extract a part of a time series
INSI - Generate time series
PICK - Pick equidistant time points
SCLOP - Do scalar operations on a time series
SLIDE - Introduce relative delays between time series
STAT - Compute some statistical numbers
TREND - Remove a trend
VECOP - Do vector operations on a time series

4. Frequency Response Operations

ASPEC - Compute an auto spectrum
CSPEC - Compute a cross spectrum
DFT - Discrete Fourier Transform
FROP - Operate on frequency responses
IDFT - Inverse Discrete Fourier Transform

5. Simulation & Model Analysis

DETER - Deterministic Simulation
DSIM - Simulation with noise
FILT - Compute a filter system
RANPA - Compute a system with random coefficients
RESID - Compute residuals with statistical tests
SPTRF - Compute the frequency response of a transfer function

6. Identification

LS - Least Squares identification
ML - Maximum Likelihood identification
SQR - Least Squares data reduction
STRUC - Least Squares structure definition

7. Miscellaneous

DELET - Delete a file
FHEAD - Inspect and change file parameters
FTEST - Check existence of a file
TURN - Change program switch settings

8. Alphabetical Command List

CONV

Purpose

To read a source file (ASCII-code) on mass storage in free format and convert it to an IDPAC - standard binary data-file.

Command

CONV DNAME < FNAM[(C1..)] NCOLX [TSAMP]

DNAME - resulting data file (Time Series)
FNAM - input free format file
C1.. - column numbers specifying which columns in FNAM that will be converted
(default: C1.. = 1, 2, .., NCOLX)
NCOLX - total number of columns in FNAM
TSAMP - sample interval in seconds (default: DELTA.)

Function

The source (symbolic) file is read in free format, i.e. strings of characters from the set

0 1 2 3 4 5 6 7 8 9 + - . E

are converted to real numbers following normal rules. All other characters including carriage return, tab, etc., are treated as delimiters. Thus, the use of formatting characters (i.e. space, tab, CR-LF, etc.) is unrestricted. However, the number of data items corresponding to a sampling instant must be constant, specified by NCOLX.

Conversion errors detected cause the output of the result (DNAME) to be inhibited. The lines causing the errors are written onto a previously unused file CONVXX (XX = 1..99). A message of this effect is given.

Examples

If FNAME is:

```
0 1 2 3 4 5 6
7 8 9 10
1.1G+1
12
13X+0
14
15
```

the command

```
>CONV DATA < FNAME(1) 2
```

will give an error message and generates the file CONV1:

```
"CONV DATA < FNAME(1) 2
"
    3.  1.1G+1
    5.  13X+O
```

If lines 3 and 5 are corrected (an E should be used)

```
>CONV DATA < FNAME(1 2) 2
>LIST DATA
```

will produce the following output:

```
0.000     1.000
2.000     3.000
4.000     5.000
6.000     7.000
8.000     9.000
10.000    11.000
12.000    13.000
14.000    15.000
```

EDIT

Purpose

To edit, i.e. create or make changes to a symbolic (text) file. Examples are MACRO-files, system files and symbolic data files from outside.

Command

EDIT TFILE

TFILE	name of symbolic (text) file

Subcommands

The following general notation is used:

n	denotes a positive integer, default 1.
/	denotes any character not included in 'string'.
string	denotes any sequence of printing characters including space.

A string	the string is appended to the current line.
B	the bottom line of the file is made the new current line.
C /string1/string2/	string1 in the current line is changed to string2.
D [n]	n lines are deleted starting with the current line.
E	exit, i.e. close the file and return.
F string	find the first line after the current line starting with string and make it current.
I string	insert string as the new current line after the now current line.
L string	locate the first line after the current line containing string and make it current.
N [n]	make the n:th next line current.

O [n]	overlay the n next lines including the current with keyboard INPUT.
P [n]	print n lines starting with the current line. The last line printed is the new current line.
R string	replace the current line with string.
T	go to the top of the file.
DIS ON DIS OFF	enable/disable output on display.

Function

The editor works in one of two modes, EDIT-mode and INPUT-mode. In EDIT-mode, the editor will read the text-file line by line. At any time, one line is the 'current line'. The subcommands control the position of the 'current line' within the text-file, or modify the 'current line'.

In INPUT-mode, a line typed on the keyboard is made the new 'current line', thus forcing the old one to be written to the output file.

The initial mode of the editor is INPUT if the specified file is not found, otherwise EDIT. An empty line is used to switch the mode.

Cautions, Restrictions

EDIT does not allow subcommands or input from a macro.

Hints

System files are normally generated by the command SYST. Exception: the polynomial image system files in Idpac.

FORMAT

Purpose

To convert a binary data file into a text (symbolic) file.

Command

FORMAT [FFILE] < BFILE[(C1 C2 ..)] [BEGIN COUNT]

FFILE	- name of formatted data file (default: BFILE)
BFILE	- name of binary data file
C1 ..	- column numbers in BFILE (default: 1 2 ..)
BEGIN	- number of first record in BFILE to be formatted (default: 1)
COUNT	- number of records to be formatted (default: all)

Function

The selected columns of BFILE are written rowwise onto the output file, in a format compatible with the command CONV.

The command line and the file head parameters are included as comments.

Cautions, Restrictions

The output format and the number of columns allowed is installation dependent.

LIST

Purpose

To output a file on lineprinter, teleprinter, or display.

Command

```
LIST [(DEV)][(FEED)][(DMODE)][AGGREG:]FNAME[(A1 A2..)][IF NUM]

DEV     - device = 'DIS'/'LP'/'TP'
          DIS   - display
          LP    - line printer
          TP    - teleprinter
          (by default DEV = 'DIS')
FEED    - form feed parameter = 'FF'/'LF'
          FF    - a form feed will precede output
          LF    - a line feed will precede output
          (by default FEED = 'LF')
DMODE   - data mode indicator = 'D'/'T'/'DS'/'TS'/'FT'/'FTS'
          D     - FNAME is assumed to contain binary data
          T     - FNAME is assumed to contain text
          DS    - same as 'D', but sequence numbers written
          TS    - same as 'T', but sequence numbers written
                  and the text will be truncated after
                  72 characters
          FT    - same as 'T' but 'BEGIN', 'END' not written
                  provided that section names
                  have been given explicitly
          FTS   - same as 'FT' with sequence numbers
          (by default DMODE = 'D')
AGGREG  - aggregate file, invalid in connection with
          DMODE = 'T'
FNAME   - name of file to be listed
A..     - attributes associated with FNAME, if
          DMODE = 'D'/'DS', then A.. denotes column numbers,
          otherwise names of sections within FNAME
IF      - number of 1st record to be listed
          (valid only in connection with DMODE = 'D'/'DS')
NUM     - number of records to be output
          (valid only in connection with DMODE = 'D'/'DS')
```

Function

The data is printed as matrix blocks with NUM lines containing the first few columns, a blank line, NUM lines containing the next few columns etc.

Note, frequency response files are special cases of data files.

Input & Output
LIST

Text files: The file is directly copied onto the output medium. Text files are:

a) any file created or manipulated by the EDIT command,
b) MACRO files,
c) system files,
d) structure files.

If a section name is given for a system file, only that section is output.

Cautions, Restrictions

The available output devices may be installation dependent.

Hints

The mechanism that allows listing of selected sections of a system file uses the keywords BEGIN and END. These keywords may be used to produce the same effect in any text file, e.g. to output descriptive text from a macro.

Examples

```
>LIST DATA
>LIST (LP) DATA(3 4 6) 20 10
>LIST (T) MAC
>LIST (LP)(T) SYST(NAME)
```

MOVE

Purpose

To move files in the data base. The columns of a data file may be rearranged.

Command

```
MOVE [(DMODE)] [[AGOUT:]FILOUT[(C11..)]] <
                                [AGIN:]FILIN[(C21..)]

DMODE    - data mode = 'D'/'T'/'ND' (default: DMODE = 'D')
           D     - the file is assumed to contain binary data
           T     - the file is assumed to contain text
           ND    - same as 'D' but the columns C11.. will,
                   if previously defined, not be overwritten,
                   but placed rightmost in FILOUT, in the
AGOUT    - output aggregate file
FILOUT..- output file name [with column numbers]
AGIN     - input aggregate file
FILIN.. - input file name [with column numbers]
```

Function

The columns C21,.. in the data file FILIN are moved to the columns C11,.. in the data file FILOUT.

Copying is the only function available for symbolic (text) files or for files within aggregates.

Cautions, Restrictions

- Column numbers cannot be used for system- and macro-files.
- Data files may contain up to 20 columns as input files and up to 15 columns as output files.

Examples

```
>MOVE WORK < DATA(2 5 3)
>MOVE (ND) WORK(1 3) < DK DATA(4 1)
```

The results are shown below.

Input & Output
MOVE

DATA

11.0000	21.0000	31.0000	41.0000	51.0000
12.0000	22.0000	32.0000	42.0000	52.0000
13.0000	23.0000	33.0000	43.0000	53.0000
14.0000	24.0000	34.0000	44.0000	54.0000
15.0000	25.0000	35.0000	45.0000	55.0000
16.0000	26.0000	36.0000	46.0000	56.0000
17.0000	27.0000	37.0000	47.0000	57.0000
18.0000	28.0000	38.0000	48.0000	58.0000
19.0000	29.0000	39.0000	49.0000	59.0000
20.0000	30.0000	40.0000	50.0000	60.0000

WORK

21.0000	51.0000	31.0000
22.0000	52.0000	32.0000
23.0000	53.0000	33.0000
24.0000	54.0000	34.0000
25.0000	55.0000	35.0000
26.0000	56.0000	36.0000
27.0000	57.0000	37.0000
28.0000	58.0000	38.0000
29.0000	59.0000	39.0000
30.0000	60.0000	40.0000

WORK

41.0000	21.0000	11.0000	51.0000	31.0000
42.0000	22.0000	12.0000	52.0000	32.0000
43.0000	23.0000	13.0000	53.0000	33.0000
44.0000	24.0000	14.0000	54.0000	34.0000
45.0000	25.0000	15.0000	55.0000	35.0000
46.0000	26.0000	16.0000	56.0000	36.0000
47.0000	27.0000	17.0000	57.0000	37.0000
48.0000	28.0000	18.0000	58.0000	38.0000
49.0000	29.0000	19.0000	59.0000	39.0000
50.0000	30.0000	20.0000	60.0000	40.0000

BODE

Purpose

To plot frequency response files in Bode diagram format.

Command

BODE [(SW)] FRF1[(F11 F12 ..)] [FRF2[(F21 ..)] ..

SW — page switch = 'A'/'P'/'AP'/'AO'
(default: 'AP')
A : plot amplitude only,
then read a sub-command
P : plot phase only, then exit
AP: plot amplitude and phase together,
then exit
AO: plot amplitude only, then exit

FRF.. — frequency response file name(s)
F11.. — frequency response number(s) (default all)

Subcommands

PAGE — request the phase plot (relevant only if SW = 'A')
KILL — skips the phase plot (relevant only if amplitude and phase are to be plotted separately)

Function

The indicated (default all) curves of the frequency response file(s) FRF1 etc., are plotted versus frequency. The abscissa is a logarithmic axis, while the ordinate is logarithmic for the amplitude and linear for the phase. If the phase information is identically zero, as for auto spectra, the phase plot is omitted.

Amplitude values smaller than 1.E-5*(largest value) are replaced by the lower limit.

If more than one set of curves are requested, they are marked with integers representing the order of the corresponding response in the command.

Hints

a) Cf. the command PLOT for methods to include text in the diagram.

Graphical Output
BODE

b) The command treats frequency response files; see the general guide. Generally, they include frequency information scaled in rad/s. If you want a bode plot in Hz, use the SCLOP command to divide the frequency by 2π = 6.2831853.

Example (cf. the commands NIC and NYQ)

The Bode plot for the system

$$G_D = \frac{3.25\ s + 1}{s^2\ (s + 1.75)}$$

is given through the command (the response is contained in the file FRF):

```
>BODE FRF
```

See Figure 1.

The Bode plot for the system

$$G_C = \frac{3.25\ s + 1}{s^3 + 1.75\ s^2 + 3.25\ s + 1}$$

is shown in Figure 2 (the response was contained in FRF2).

Graphical Output
BODE

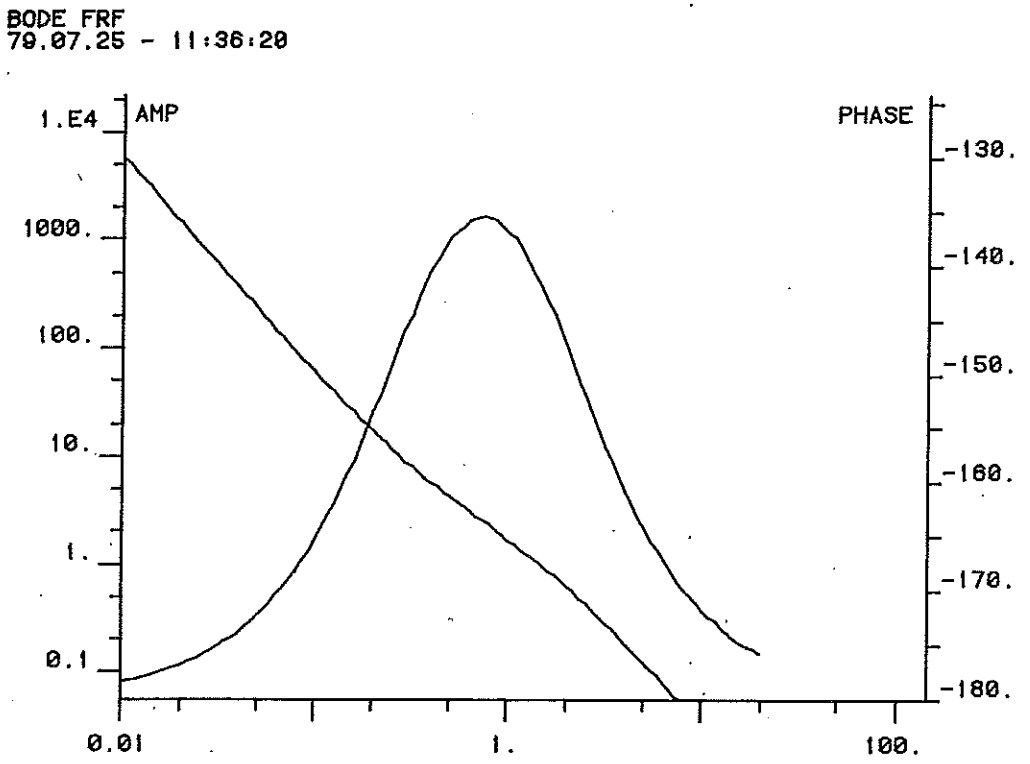


Figure 1. The amplitude and phase of G_O.

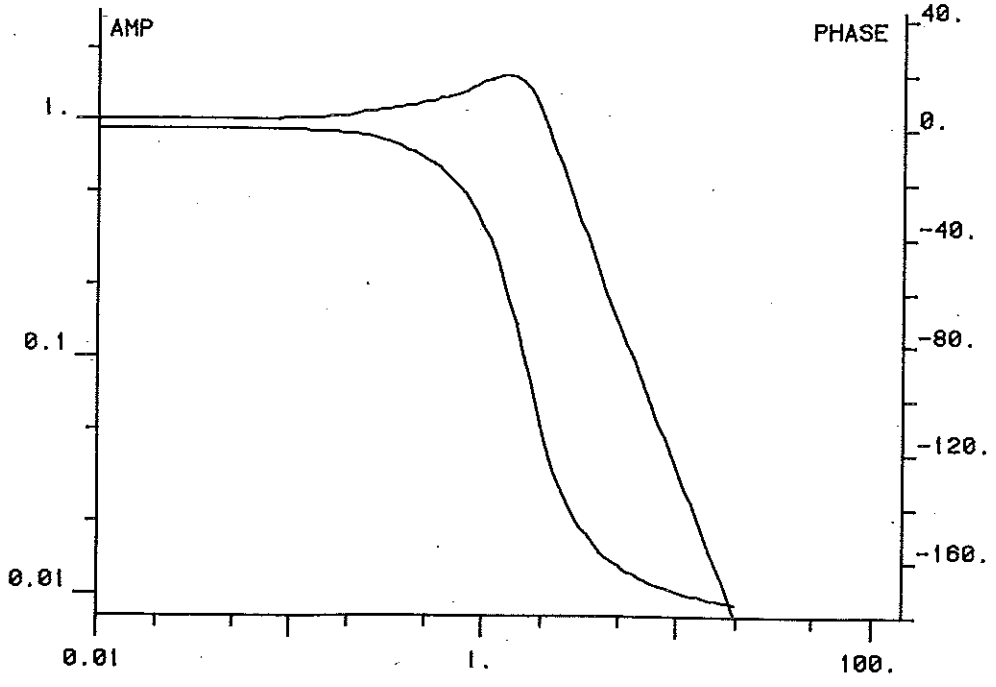


Figure 2. The amplitude and phase of G_C.

HCOPY

Purpose

To generate a hard copy of graphical output.

Command

```
HCOPY [DEV] [FACTOR]
   or
HCOPY SWITCH
```

DEV - hardcopy device = 'L'/'R' (default: 'L')
  L: local hardcopy (Tektronix 4662)
  R: remote hardcopy (Calcomp 1051)

FACTOR - scale factor (default: 1.)
  For L: .5 < FACTOR < 1.6
  For R: .5 < FACTOR < 4.

SWITCH - = 'ON'/'OFF'/'T'
  ON : enables hardcopy
  OFF: disables hardcopy
  T : output a free text string at the current joystick location;applies to TEKTRONIX 4662 only, the textstring in the command line being preceded by a double quote

Function

After that the command HCOPY has been used with the switch ON, all graphical output that is generated in any command is also saved temporarily. A subsequent use of the command HCOPY will cause the saved information from the last such command to be sent to the selected hard copy device.

Hints

Note that HCOPY actually is available as a subcommand for all graphic generating command.

PLMAG

Purpose

To plot parts of a time series file on display and enable the user to alter data values.

Command

PLMAG DATA [(C)]

DATA - data file name
C - column number for data (by default C=1)

Subcommands

B[LOCK] NB - defines the number of data points to be plotted per page (initially NB = 50, NB =< 175)
P[LBEG] NR - NB points are plotted from the NR:th on
A[LTER] NR [NUM] - alter the value of NUM (default 1) points, starting with the NR:th point if NUM is omitted, the old value is written on the tty and a new may be entered, if the new value is not accepted, just type '<' or '>' to access the previous or next data point resp. if NUM printed, all new data may be entered on the same line.
PA[GE] - next block is plotted
D[ELET] NR [NUM] - delet NUM (default 1) data points starting at the NR:th point
KILL - abort the subcommand sequence and leave DATA unaffected
X - effectuate the subcommand sequence and update DATA

Function

The Cth (default 1st) column in the time series file DNAME may be plotted and data values altered using the subcommands.

Cautions, Restrictions

A data point must be plotted before it can be altered.

Graphical Output
PLMAG

Example

The following few command lines are used to correct an erroneous data point in FILE. Figures 1 and 2 show part of the file before and after the change.

```
>PLMAG FILE
      >B 20
      >P 35
      >A 43
       43 OLD VALUE 4.85 #>
       44 OLD VALUE 0.0  #4.75
```

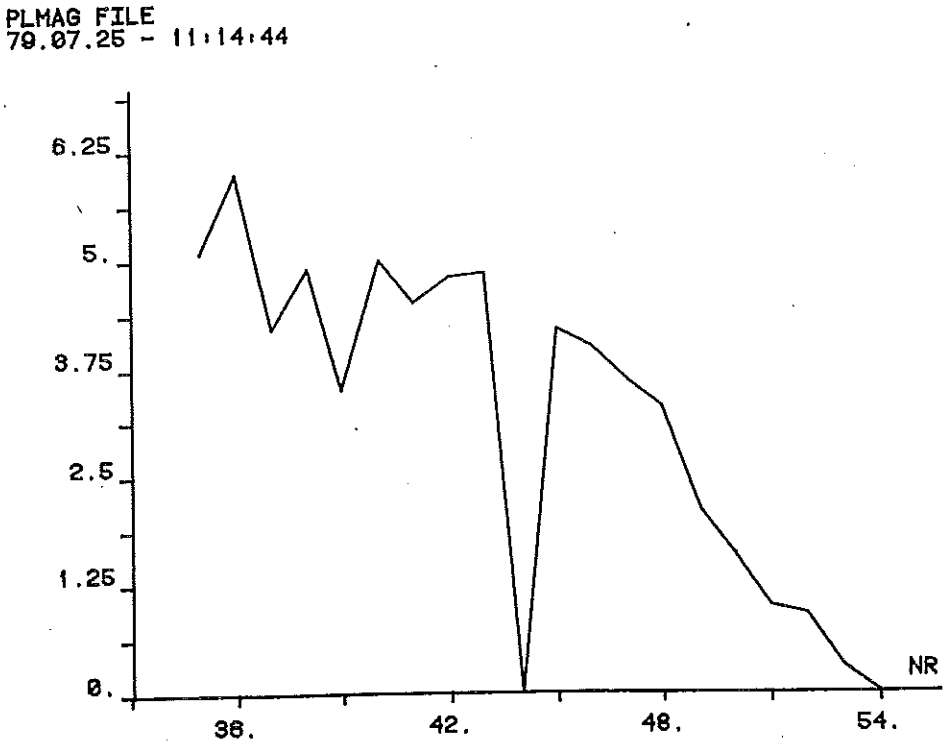


Figure 1. The data before the change.

Graphical Output
PLMAG

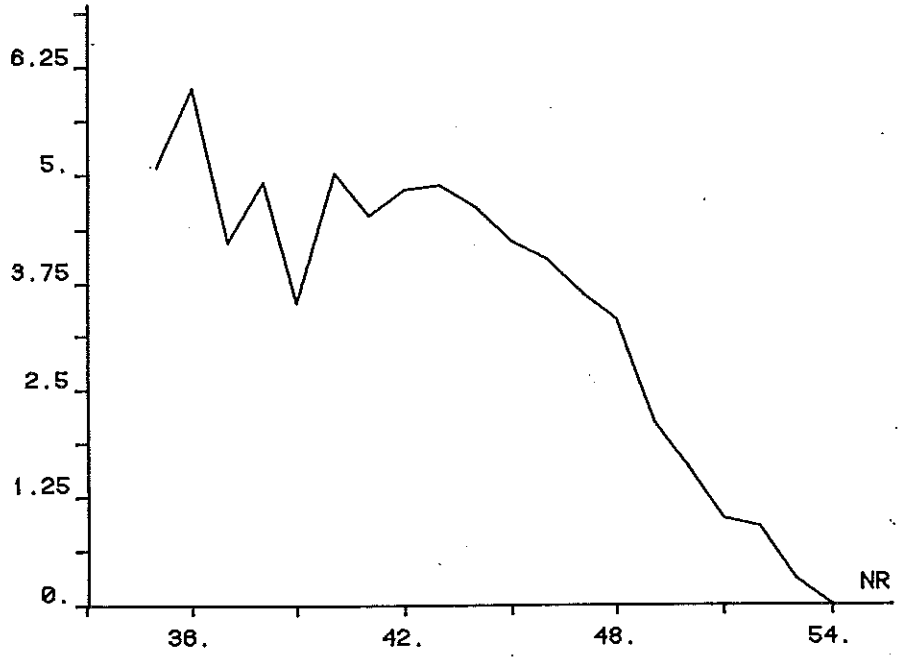


Figure 2. The data after the change.

PLOT

Purpose

To show data in diagrams with linear scales.

Command

```
PLOT [(M)] [(NP)] [FNAMX[(C1..)] < ] [(OPT1)] FNAM1[(C11..)]
     [[(OPT2)] [FNAM2[(C21..)]] .. ] [YMI YMA]
```

```
M        - mark option = 'M'/'NM' (default: 'M')
           M     - when more than one curve is plotted,
                   the curves are marked with integers
                   representing the order of the
                   corresponding column in the PLOT command
           NM    - no curve marks
NP       - nr of points per page (by default NP=NPLX.)
FNAMX    - optional file containing x-values if plotting
           versus time/sample number not wanted
C1       - column number for FNAMX (by default C1=1)
OPTi     - plot option='LI'/'HP'/'NL' (by default OPT='LI')
           LI: linear interpolation plot
           HP: histogram plot
           NL: no lines will connect the plotted points
FNAMi    - i:th data file containing the y-values
Cji..    - column numbers of FNAMj
YMI      - minimum value for this plot (by default YMI=YMIN.)
YMA      - maximum value for this plot (by default YMA=YMAX.)
```

Screen split operators inserted between groups of file names:

```
'/'      - divides the plotting area horizontally
':'      - divides the plotting area vertically
```

These operators express the mutual positioning of graph groups. It is allowed to divide the screen at most into three parts horizontally and two parts vertically.

Subcommands

```
KILL     - skip all the following plot pages
PAGE     - plot the next plot page in turn
SKIP [N]- skip the N (deafault: 1) following plot pages
```

Graphical Output
PLOT

Function

The indicated column(s) in the data file(s) are shown in graphic form, normally directly on the operators terminal.

In normal operation, the information is shown along the vertical axis versus a linearly increasing variable on the horizontal axis (time). The scaling of the time variable may, by use of the TIME switch be altered from sample number in the command TURN, to hour, minute or second. If the argument FNAMX is included, the variable shown along the horizontal axis is taken from a file with this name. The x-axis is marked with an H, M or S depending on the time unit, with NR if the sample number is used, otherwise it is left blank.

The number of values shown on a single page is determined from the argument NP if present, else from the global variable NPLX. The global variables referenced by the PLOT command are given default values at program start up. They may be inspected by the WRITE statement of Intrac and may be modified with the LET statement. The TIME switch determines the interpretation of NP (NPLX.).

When more than one variable is shown, they are identified by integers according to the order of the variables in the command line.

The scaling along the vertical axis is determined in the following way:

a) The arguments YMI and YMA are present or the global variables YMAX. and YMIN. satisfies YMAX.>YMIN.: Then YMI (YMIN.) is used as the minimum value on the axis and YMA (YMAX.) is used as the maximum value.

b) Otherwise automatic (i.e. data dependent) scaling is performed. Automatic scaling is always used if the screen-split operations are used.

Automatic scaling is influenced by the global variable SCALES.:

If SCALES.≠0 then scale marks are multiples of either 1.0, 1.5, 2.0, 2.5, or 5.0.

If SCALES.=0 then scale marks are multiples of either 1.0, 2.0, or 5.0.

Hints

a) Note that the command line is written on the display above the plot and that comments can be added (after a double quote).

b) If the users terminal is a display with a direct hard copy option, the command WRITE (DIS) ... may be used to add text to the plot.

c) The command HCOPY may be used to include text on an on-line plotter if such is available.

Examples

V1 and V2 are two given signals. They may be visualized in the following way:

```
>PLOT V1 V2
```

giving the result shown in Figure 1. Alternatively, they may be shown in the following way:

```
>PLOT V1 / V2
```

The result is shown in Figure 2.

In order to visually study their interrelation, a scatter diagram might be constructed using an x-y plot and the "No Lines" switch:

```
PLOT V1 < (NL) V2
```

It is shown in Figure 3.

A pulse with IFP.=3 and LENGTH=2 (cf. the command INSI) looks like this (Figure 4) with and without the "HP" switch:

```
>LET IFP.=3
>INSI PULSE 7
    >PULSE 2
    >X
>PLOT (7) PULSE / (HP) PULSE
```

Graphical Output
PLOT

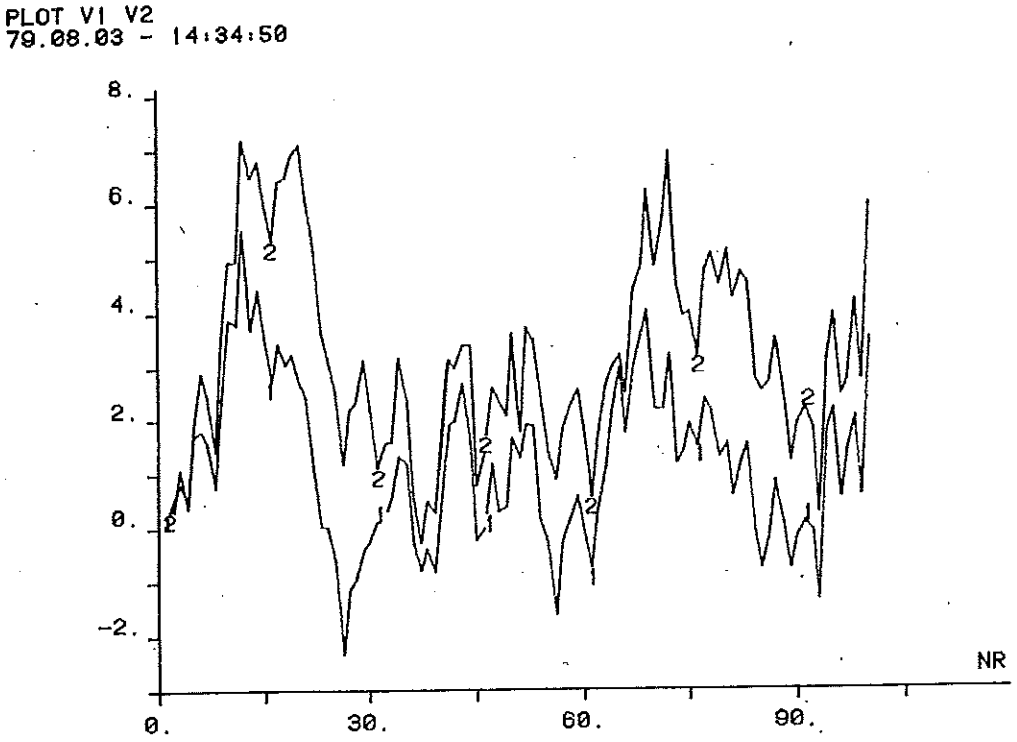


Figure_1. A plot with two curves in the same diagram.

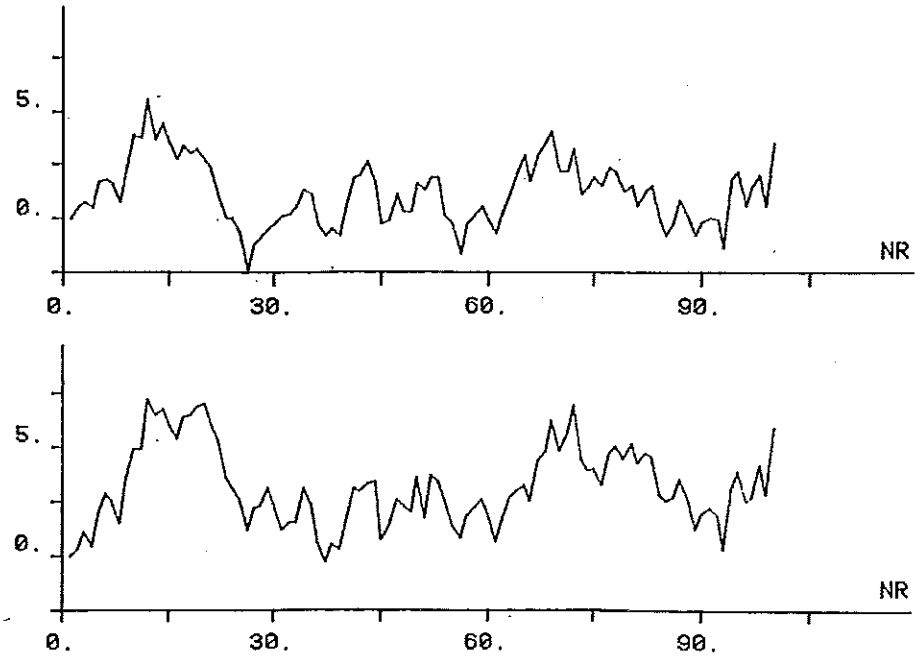


Figure_2. A plot with the screen split horizontally.

Graphical Output
PLOT

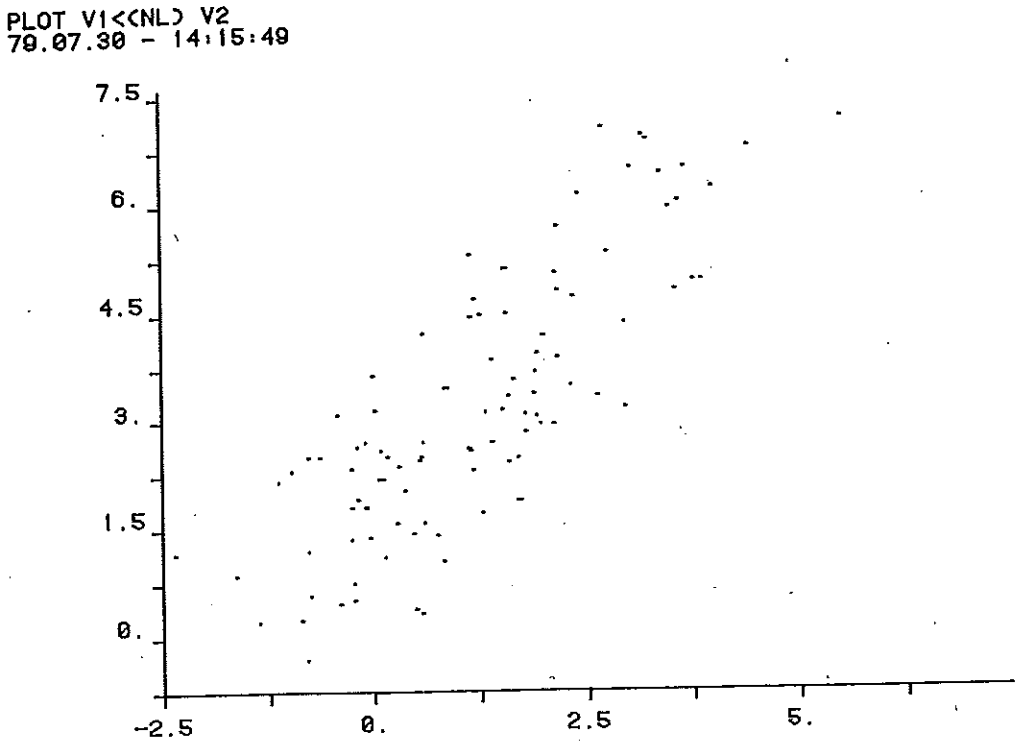


Figure 3. A plot in the x-y mode of operation.

Graphical Output
PLOT

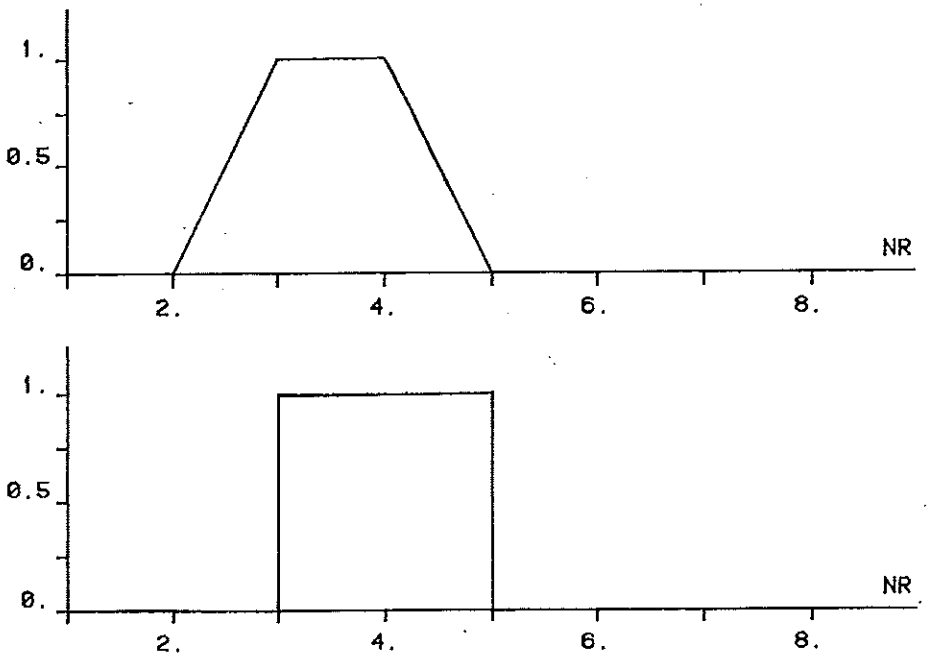


Figure 4. An illustration of linear interpolation vs. histrogram plot.

ACOF

Purpose

To compute the autocorrelation function of a time series.

Commands

```
ACOF FNAM1[(C1)] < FNAM2[(C2)] NOL [EXT]
```

FNAM1	- auto covariance file name
C1	- auto covariance index (default: 1)
FNAM2	- time series file name
C2	- time series index (default: 1)
NOL	- number of lags for computation of autocovariances
EXT	- name extension for the variance

Function

The autocorrelation of the specified time series is computed and output as a new time series containing NOL+1 points.

If EXT is included and if VAR.EXT is defined as a real global variable, it receives the variance of FNAME, i.e. $R_{xx}(0)$.

Method

The autocovariances are computed using

$$R_{xx}(\tau) = \frac{1}{N} \sum_{J=\tau+1}^{N} (x_J - m)(x_{J-\tau} - m) \qquad \tau = 0,1,..,NOL$$

where N is the number of data in FNAM2(C2), x_J is the J:th point, and m is the mean value of the data in FNAM2(C2). Then the autocorrelations are computed from

$$r_{xx}(\tau) = \frac{R_{xx}(\tau)}{R_{xx}(0)} \qquad \tau = 0,1,..,NOL$$

Cautions: Restrictions

The maximum number of lags is 500.

Hints

a) Note that if the switch TIME is OFF, the value $r_{xx}(0)$ will appear as sample point number 1, being the first value in the file. If you use TIME S/M/H, $r_{xx}(0)$ will appear on the y-axis.

b) If it is the autocovariance you want, use the possibility to obtain the variance as above and rescale using SCLOP, see the example.

c) Note that there is a heavy correlation between neighbouring points of the computed correlation function.

Example

The following command sequence will compute the auto covariance function.

```
>LET VAR.N = 0.0
>ACOF NCORR < NOISE 100 N
>SCLOP NCOVAR < NCORR * VAR.N
```

Time Series Operations
CCOF

CCOF

Purpose

To compute the cross correlation between two time series.

Commands

```
CCOF FNAM1[(C1)] < FNAM2(C21 C22) NOL
   or
CCOF FNAM1[(C1)] < FNAM2[(C2)] FNAM3[(C3)] NOL
```

```
FNAM1    - cross covariance file name
C1       - cross covariance index (default: 1)
FNAM2,3  - time series files
C2,3     - time series indices (default: 1)
NOL      - number of time lags
IALIGN   - the number of lags necessary to align the
           processes so that the largest cross covariance
           is centered at zero (default 0)
```

Function

The cross correlation of the two specified time series is computed for -NOL through 0 to NOL lags and output as a new time series.

Method

The cross covariances are computed using

$$R_{xy}(\tau) = \frac{1}{N} \sum_{j=\max(1,1-\tau)}^{\max(N-\tau,N)} (x_j - m_x)(y_{j+\tau} - m_y), \quad \tau=-NOL,..,NOL$$

where N is the number of data in each inut vector, x_j and y_j are the j:th data values, and m_x and m_y are the mean values of the data in the input vectors. Then the cross correlations are computed as

$$r_{xy}(\tau) = \frac{R_{xy}(\tau)}{\sqrt{R_{xx}(0)\, R_{yy}(0)}} .$$

The signal x above corresponds to the first time series in the right hand arguments and the signal y to the second one.

Cautions; Restrictions

The maximum number of lags is 500.

Hints

a) The cross covariance can be recomputed by multiplying the cross correlation with the product of the standard deviation of the two time series. The standard deviations are attainable from the command STAT.

b) If you want to plot the cross correlation with a properly scaled x-axis, use an x-y plot with the x-variable generated as a ramp with INSI, see the example below.

Example

If the input x to a system is white noise, the cross covariance R_{xy}, where y is the output, is the impulse response. In the following command sequence, the cross correlation between x and y is computed and scaled to represent the cross covariance. Finally a suitable x-axis variable is generated and the result is plotted. The file YPULSE contains the true impulse response of the system.

```
>CCOF CCF < X Y 30
>LET STDEV.X = 0.
>LET STDEV.Y = 0.
>STAT X X
>STAT Y Y
>LET A = STDEV.X * STDEV.Y
>SCLOP CCOVF < CCF * A
>INSI XAX 61
    >RAMP -30 1
    >X
>PLOT (61) XAX < CCOVF YPULSE
```

Time Series Operations
CCOF

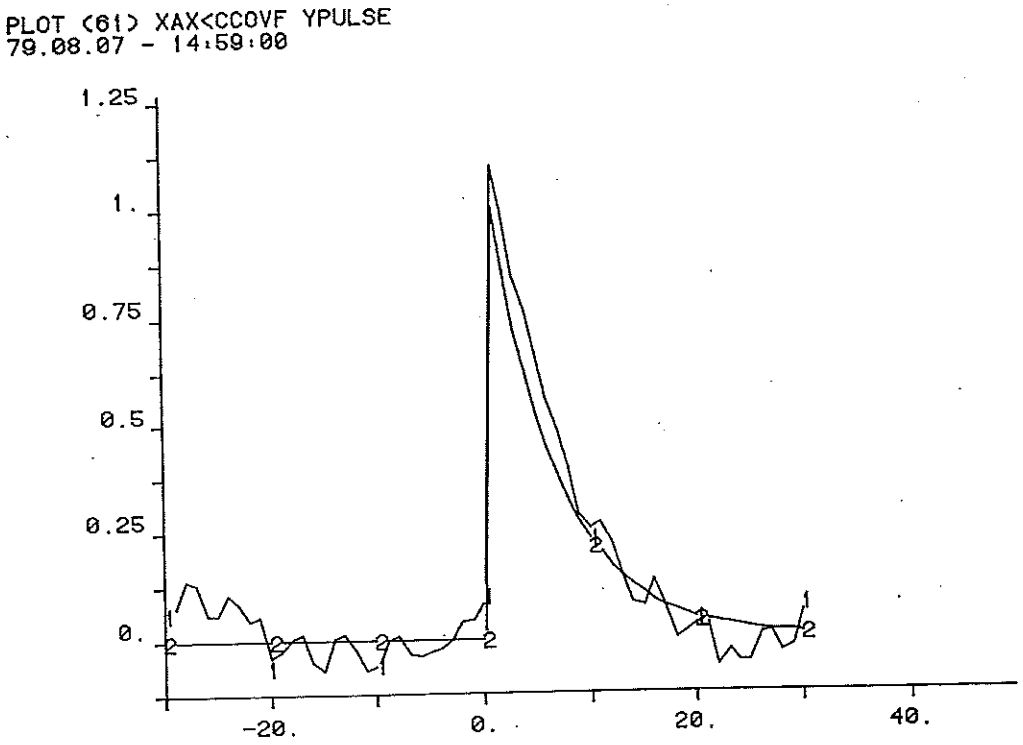


Figure 1. The estimated and true impulse response of a system.

CONC

Purpose

To concatenate two time series files.

Command

CONC [FNAM1] < FNAM2 FNAM3

FNAM - names of output and input files, resp.
(by default FNAM1 = FNAM2)

Function

The files FNAM2 and FNAM3 are concatenated giving the file FNAM1. They must contain the same number of columns.

CUT

Purpose

To cut out a part of a time series file.

Command

CUT [FNAM1] < FNAM2 IB NUM

FNAM1 - output file (default: FNAM2)
FNAM2 - input file
IB - first record to be saved
NUM - number of records to be saved

Function

The rows IB through IB+NUM-1 in the file FNAM2 are moved to the file FNAM1. Cut operates on all columns of the file.

Example

```
>CUT SHORT < FILE 2 7

FILE
      11.0000     21.0000     31.0000
      12.0000     22.0000     32.0000
      13.0000     23.0000     33.0000
      14.0000     24.0000     34.0000
      15.0000     25.0000     35.0000
      16.0000     26.0000     36.0000
      17.0000     27.0000     37.0000
      18.0000     28.0000     38.0000
      19.0000     29.0000     39.0000
      20.0000     30.0000     40.0000

SHORT
      12.0000     22.0000     32.0000
      13.0000     23.0000     33.0000
      14.0000     24.0000     34.0000
      15.0000     25.0000     35.0000
      16.0000     26.0000     36.0000
      17.0000     27.0000     37.0000
```

INSI

Purpose

Generates signals in columns of a time series file. The signals may be of the following types:

PRBS - a pseudorandom binary sequence
NORM - a pseudo random signal with normal (gaussian) distribution
RECT - a pseudo random signal with rectangular distribution
SINE - a sine wave
ZERO - a signal identically zero
STEP - a step signal
RAMP - a ramp signal
PULSE - a pulse signal
SRTW - a sequential random telegraph wave

Command

INSI FNAME [(C)] NP [TSAMP]

FNAME - output file name
C - column number (default 1)
NP - number of data points wanted
TSAMP - sample interval in seconds (default DELTA.)

Subcommands

PRBS [IBP [NBIT [ISTART [OPT]]]]

IBP - basic period (1)
NBIT - number of bits in shift register min 3, max 16 (7)
ISTART - specifies starting point in the sequence 1,2 or 3 (1)
OPT - trick option = 'KNEP'/'VOID' ('VOID')
KNEP - F.O.A.-trick is used
VOID - no trick

NORM [RMEAN SIGMA]
RMEAN - mean value (0.0)
SIGMA - std. dev. (1.0)

RECT [A B]
A - lower boundary (0.0)
B - upper boundary (1.0)

SINE [OMEGA FI]
OMEGA - frequency (1.0 radian/s)
FI - phase (0.0 degrees)

ZERO

STEP

RAMP [A B]

Time Series Operations
INSI

```
        A       - constant term (0.0)
        B       - linear term (1.0)
PULSE [LENGTH]
        LENGTH  - pulse length in samples (1)
SRTW [PS]
        PS      - change-of-sign probability (0.5)
LOOK    - display names of subcommands and reserved
          variables
KILL    - abort the operation of INSI and resume
          main command mode
X       - let the previous subcommands take effect,
          then resume main command mode
```

Function

Sequencies corresponding to the specifying subcommands are generated and upon the receipt of the command X they are stored in the specified file.

INSI makes implicit use of the reserved global variables IFP., NU., AMP., and DELTA.. IFP. specifies in all cases the sample point where the sequence should start, all previous values are set to zero. NU. is the state of the internal random number generator, and is automatically updated by INSI. It is sometimes desirable to save and/or initialize NU. using the LET command in order to obtain reproducible results. AMP. is used as the amplitude of some signals, see below. DELTA. is the value of the sampling interval to be recorded in the file.

Particulars:

PRBS
This signal is generated as the output of a shift register with feedback. The shift register is clocked at each IBP:th sampling point. NBIT specifies the length of the shift register. The output repeats itself after $2^{NBIT}-1$ clock instants. Four different initial values, i.e. the starting points in the periodic sequence, can be chosen by ISTART. The last argument, the "trick" parameter KNEP, specifies a slightly different form of the PRBS output; it is the output of a flip-flop complementing at the clock instants if the original sequence is negative. Thus the new sequence has asymptotically zero mean value, which the standard PRBS has not.

RECT
A rectangularly distributed pseudo random signal is generated by a mixed congruential method. NU. is used and altered.

NORM
An approximately normally distributed (i.e. gaussian) pseudorandom signal is generated as the sum of twelve rectangular values as above. NU. is used and altered.

SRTW
This is a two-valued (+a or -a) signal with the amplitude (a) specified by AMP. and with a given change-of-sign probability. Changes of sign occur when a rectangularly distributed (0,1) random number as above computed at each sample point falls below the parameter specified in the command. NU. is used and altered.

SINE
The phase refers to the phase at the starting point (i.e. IFP.). Thus FI = $\pi/2$ = 1.570796 will give a cosine signal. AMP. is the amplitude.

ZERO
The signal is constantly zero.

STEP
The signal changes from zero to AMP. at the sampling point IFP..

RAMP
Starting at sampling point IFP., a signal of the form B*(i-IFP)+A is generated, where i is the number of the sampling point.

PULSE
A pulse of amplitude AMP. starts at sampling point IFP. and lasts a specified number of sampling intervals.

Cautions, Restrictions

If the file specified in the command already exists, it will be changed or expanded, provided that the length and column number arguments are compatible with the old file.

Hints

a) The global variables IFP., NU. etc. are of course accessible by the commands LET, READ, and WRITE, also between subcommands.

b) More complicated signals can be constructed using commands like CUT, CONC, SCLOP, and VECOP on the results of INSI.

Time Series Operations
INSI

Example

The following operations yield the signal shown below (see Figure 1).

```
>INSI  T  30
     >LET IFP.=10
     >RAMP 0. 0.2
     >LET IFP.=20
     >RAMP -2. -0.2
     >X
>VECOP U < T(1) + T(2)
>SCLOP U < U + 0.5
>PLOT (30) U 0 2.5
```

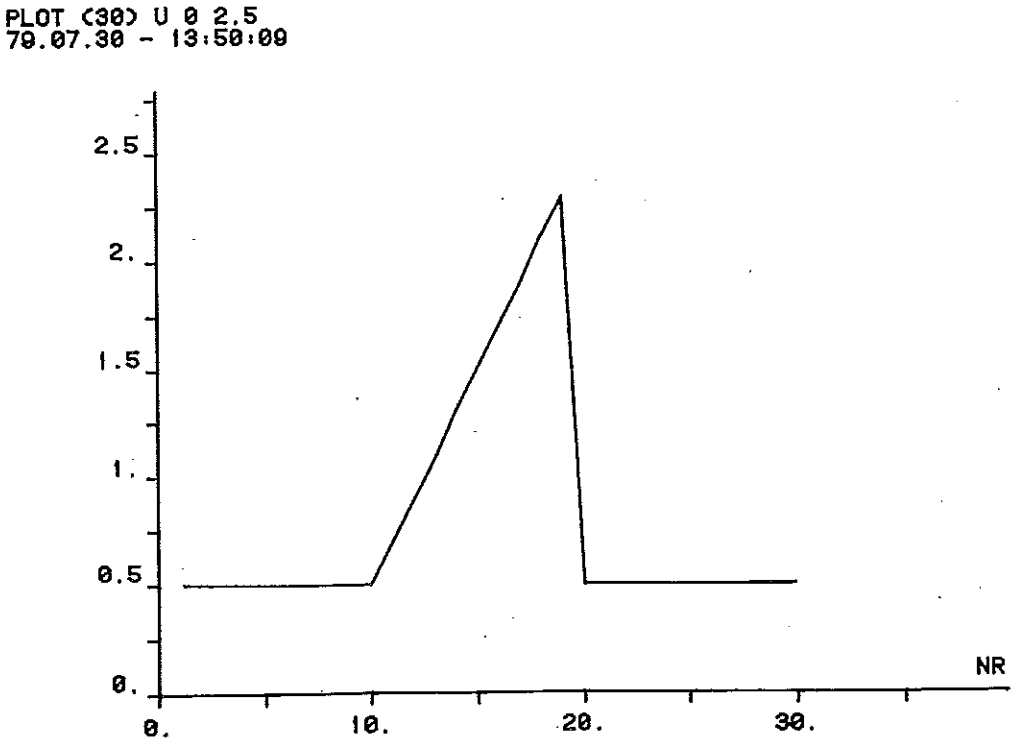


Figure 1

PICK

Purpose

To pick out equidistant samples from a time series file.

Command

PICK FNAM1 < FNAM2 N

FNAM1 - output file name
FNAM2 - input file name
N - each N:th record in FNAM2 is written into FNAM1

Function

Each N:th sample in the file FNAM2 is transferred to the file FNAM1. PICK operates on all columns of the file.

Cautions; Restrictions

Be cautious not to violate the sampling theorem conditions in using PICK.

SCLOP

Purpose

To perform scalar operations on a data vector such as amplitude scaling or offset correction.

Command

SCLOP [FNAM1[(C1)] < FNAM2[(C2)] OPER CONST

FNAM1	- name of output file (default FNAM2)
C1	- column number for output file (default 1)
FNAM2	- name of input file
C2	- column number for FNAM2 (default 1)
OPER	- operation to be performed = '+', '-', '*' or '/'
CONST	- variable or unsigned numerical constant preceded by a space

Function

Each element in column C2 of FNAM2 is added, subtracted, multiplied or divided by CONST. The resulting data vector is placed in column C1 of FNAM1.

Hints

To subtract the mean value of a data vector use STAT to compute the mean and then SCLOP to subtract it. This can also be done as a 0:th order trend correction by TREND.

SLIDE

Purpose

To move the columns of a data file relative to each other.

Command

SLIDE [FNAM1] < FNAM2 K1 K2 K3 ..

FNAM1 - output file name (default FNAM2)
FNAM2 - input file name
K1.. - time delays for columns 1, 2, 3, .. (signed)

Function

Row i in the output file will consist of element $i+k_1$, $i+k_2$, $i+k_3$ ... from respective column in the input file, where k_J is

$$K_J - \min_m(K_m).$$

Thus the operation

$$q^{k_J}$$

is performed on each column of the file, where q is the forward shift operator. A resulting positive value of k_i will thus result in an upwards shift of that column in a listing or to a left shift in a plot.

Cautions, Restrictions

The maximum difference between Ki and K_J must not exceed 175. There must be a K for each column in the file.

Time Series Operations
SLIDE

Hints

SLIDE is a means of introducing or removing delays in time series. If WRK contains data, the input in WRK(1) and the output in WRK(2), from measurements on a process with a known delay of 3 sample intervals, the following two commands are equivalent and will give a model without time delays:

```
>SLIDE  < WRK -3 0
>SLIDE  < WRK  0 3
```

Example

The command

```
>SLIDE < FILE -1 0 2
```

has the following effect on FILE:

```
FILE  (before)
     11.0000    21.0000    31.0000
     12.0000    22.0000    32.0000
     13.0000    23.0000    33.0000
     14.0000    24.0000    34.0000
     15.0000    25.0000    35.0000
     16.0000    26.0000    36.0000
     17.0000    27.0000    37.0000
     18.0000    28.0000    38.0000
     19.0000    29.0000    39.0000
     20.0000    30.0000    40.0000

FILE  (after)
     11.0000    22.0000    34.0000
     12.0000    23.0000    35.0000
     13.0000    24.0000    36.0000
     14.0000    25.0000    37.0000
     15.0000    26.0000    38.0000
     16.0000    27.0000    39.0000
     17.0000    28.0000    40.0000
```

STAT

Purpose

To compute the statistical properties sum, mean, variance, standard deviation, minimum, and maximum for a time series.

Command

```
STAT FNAME [(C)] [EXT]
```

```
FNAME    - name of data file
C        - column number (default: 1)
EXT      - name extension for global variables
```

Function

The sum, mean, variance, standard deviation, minimum, and maximum for the C:th (default 1st) column in the file FNAME are computed and displayed. The results will also be printed on line printer if the reserved variable PRINT. is nonzero. The output on the terminal will appear only if the switch 'TEXT' is 'ON'.

Those of the global variables SUM.EXT, MEAN.EXT, VAR.EXT, STDEV.EXT, MIN.EXT, and MAX.EXT that are previously defined as real variables will receive the appropriate value, provided that EXT is specified in the command.

Example

```
>STAT UT(2)

    UT(2)
    SUM      = -216.508
    MEAN     = -2.16508
    VARIANCE =  424.615
    ST.DEV.  =  20.6062
    MINIMUM  = -43.0363
    MAXIMUM  =  48.3345
    LENGTH   =  100
```

TREND

Purpose

To estimate and remove a polynomial trend from a part of a data vector.

Command

TREND [FNAM1[(C1)]] < FNAM2[(C2)] NO [IF NUM]

FNAM1	- output file name (default: FNAM2)
C1	- output column number (default 1)
FNAM2	- input file name
C2	- input column number (default 1)
NO	- order of the trend polynomial
IF	- first point to be corrected (default the 1st record)
NUM	- number of points to be corrected (default all)

Function

A polynomial trend of order NO is estimated for the C2:th column in the data file FNAM2 between the IF:th and (IF+NUM-1):th points. The trend is subtracted and the result is placed in the C1:th column of the data file FNAM1.

If the reserved variable PRINT. is nonzero, the parameters (in a coordinate system at the left end-point of the interval) will be printed on the line printer.

Method

A least squares technique is used where the parameters are estimated with reference to the midpoint of the interval.

Reference

Otnes & Enochson, Digital Time Series Analysis, Wiley, 1972.

Cautions, Restrictions

The order of the polynomial must be between 0 and 3.

Example

A time series stored in DATA is given, see Figure 1. The trend is removed and computed

```
>TREND Y < DATA 1 30 75
>VECOP TREND < DATA - Y
```

producing the signal with the trend removed (Y), Figure 2, and the estimated trend itself (TREND), Figure 3.

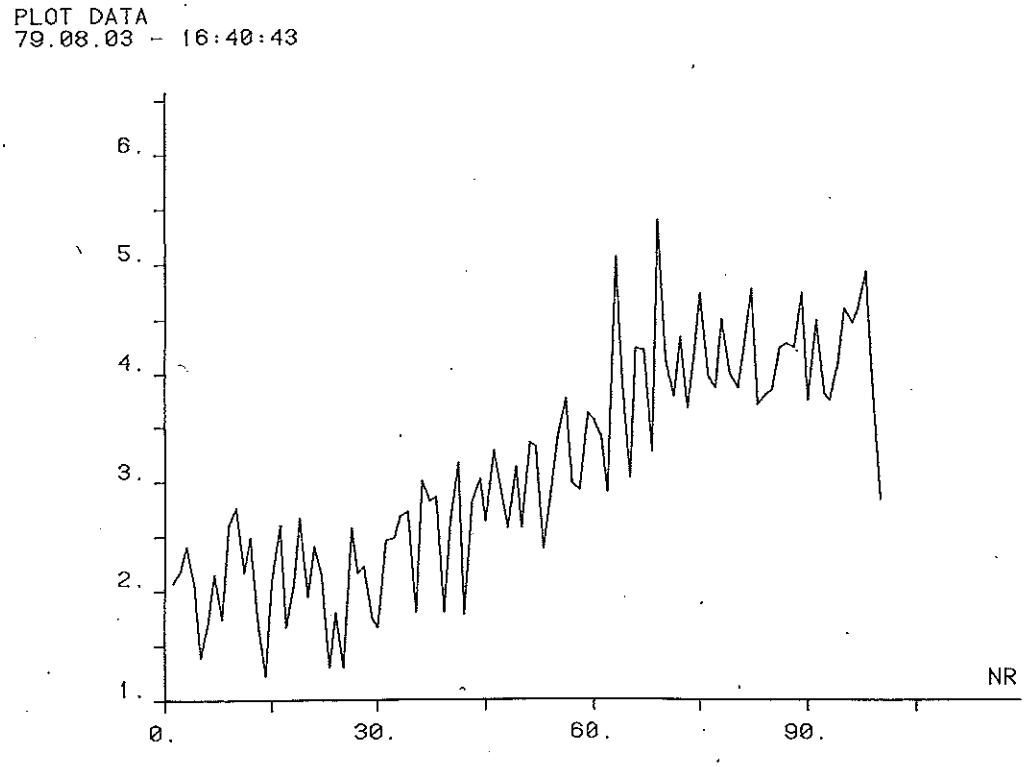


Figure_1. The original data including a trend.

Time Series Operations
TREND

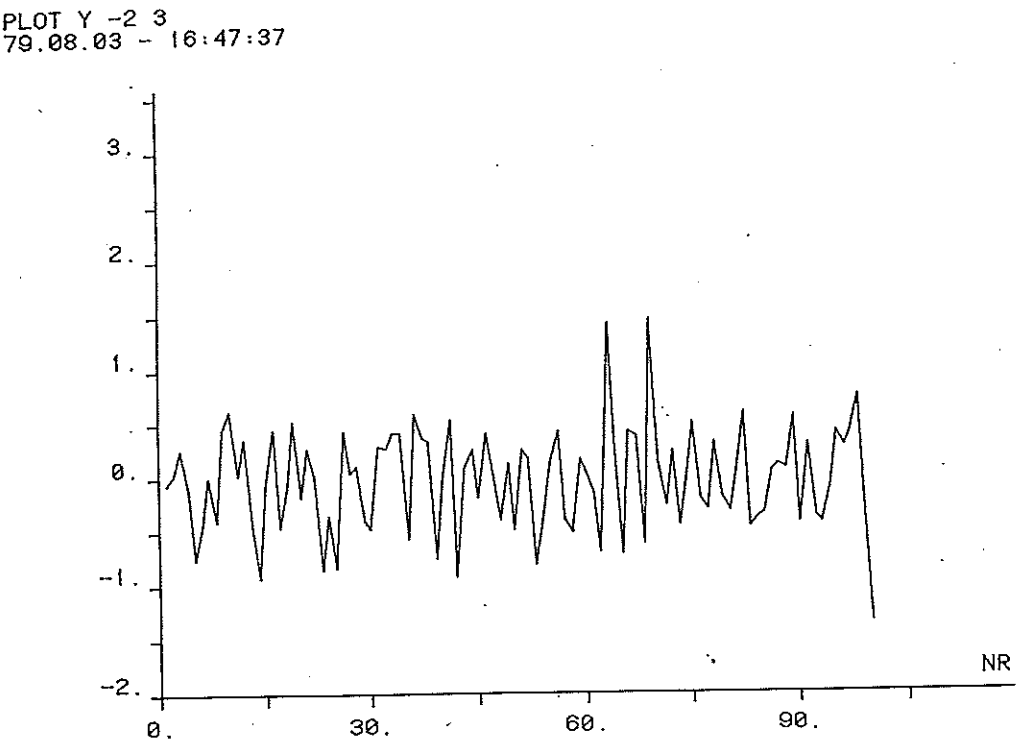


Figure 2. The trend has been removed.

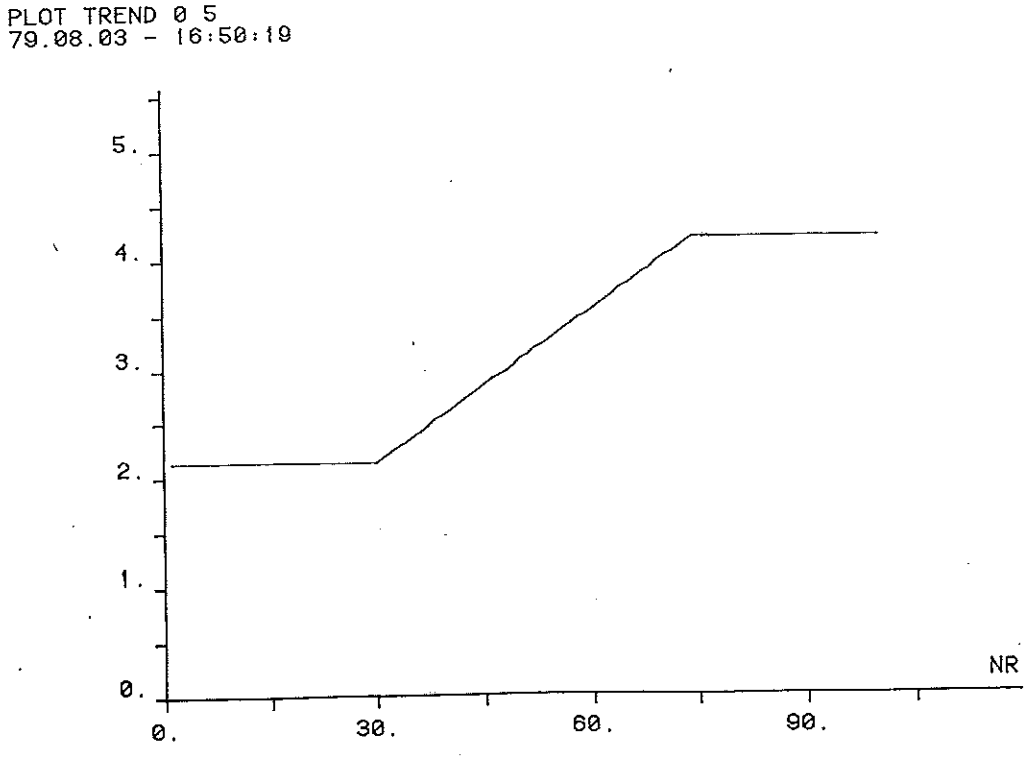


Figure 3. The estimated trend itself.

VECOP

Purpose

To add, subtract, multiply, or divide two data vectors element by element.

Command

VECOP [FNAM1[(C1)] < FNAM2[(C2)] OPER FNAM3[(C3)]

FNAM1 - output file name (default FNAM2)
C1 - column number within FNAM1 (default 1)
FNAM2,3 - input file names
C2,3 - column numbers within FNAM2,3 (default 1)
OPER - type of operation to perform = '+', '-', '*', or '/'

Function

Each element in column C2 of FNAM2 is added, subtracted, multiplied, or divided by the corresponding element in column C3 of FNAM3. The resulting data vector is placed in column C1 of FNAM1.

ASPEC

Purpose

To compute the autospectrum (powerspectrum) of a time series.

Commands

ASPEC FRF[(F)] < FNAM2[(C2)] NOL [FREQ]

FNAM2	- time series file name
C2	- time series index (default: 1)
FRF	- frequency response file name
F	- frequency response index (default: 1)
NOL	- number of lags for computation of autocovariances
FREQ	- file with frequency points in the first column

Function

The autospectrum of the input time series is computed using the autocovariances for up to NOL lags and written into FRF(F).

The frequency points are chosen in one of two ways. (NPOI is the number of data points in FNAM2 and T is the sample interval.)

a) NOF. points between Max(WMIN.,2π/NPOI*T) and Min(WMAX.,π/T). NOF., WMIN., and WMAX. are reserved global variables. The frequency points are distributed logarithmically in this interval (to achieve a uniformly dense BODE diagram).

b) If the file FREQ has been specified, the frequency points are read from (column 1 of) this file. Still there is a check for the 2π/NPOI*T and π/T limits.

Method

The autocovariances are computed as described for the command ACOF. Then the autospectrum is computed from

$$\Phi(\omega) = \frac{T}{\pi}\left\{ R_{xx}(0) + 2\sum_{\tau=1}^{NOL} R_{xx}(\tau)\cos(\omega T\tau)\, W_{BH}(\tau) \right\}$$

where ω is the frequency in rad/s, T is the sample period in s, $R_{xx}(\tau)$ is the autocovariance for lag τ, and $W_{BH}(\tau)$ is a Blackman-Harris window. The intent of the window function is to introduce an equalizing effect in the estimated autospectrum. Note that a wide time window (large NOL) corresponds to a narrow frequency window. The computed autospectrum is denoted $\Phi_x(\omega)$.

Reference

Jenkins, Watts: Spectral analysis and its applications. Holden-Day, 1968.

Cautions, Restrictions

Data files with zero sample period are not accepted as input files. Maximum number of frequencies are 500. The computed spectrum is one-sided.

Hints

a) ASPEC gives the resulting spectrum versus the angular frequency in rad/s. If you want, you can rescale the frequency information using SCLOP, cf. BODE.

b) Start with a frequency window corresponding to NOL in the range of 5% - 20% of the number of data points, then try opening and/or closing the window. A narrow window (large NOL) is likely to give too much and unjustifiable detail. A broad window (small NOL will give smooth curves where resonance peaks may be hard to detect or separate.

c) Note that the window has a constant bandwidth as demonstrated in the examples below. When plotting the computed frequency response in a logarithmic diagram (BODE), this fact shows as a low frequency region much smoother than the high frequency one.

d) Use the command DFT to analyse the frequency contents of periodic signals with little noise.

e) The autospectrum integrated over the angular frequency range (0 - π/T rad/s) will equal the variance of the signal.

Frequency Response Operations
ASPEC

Examples

The first example (Figure 1) shows the computed autospectrum of white gaussian noise that has passed through a third order filter.

The first curve with NOL = 20 % of the number of data points (N) is typical of a narrow window. Opening the window to NOL = 5 % of N produces a smooth result. The question is however: is it too smooth? Should there maybe be a weak resonance peak at ω = 0.17 as indicated by the first curve?

The second example serves to illustrate the effect of the window width. The frequency responses are computed with unusually high resolution; NOF.=400. The signal is the sum of two pure sinusoids with ω = 1. and ω = 5. respectively. Figure 2.1 shows the autospectrum with a relatively narrow window. The two peaks stand out clearly and the many side lobes of the window are seen to be very small. The effect of the logarithmic frequency axis is clearly seen and is much more accentuated when the window is made broader, Fig. 2.2. In Figure 2.3, the window is so broad that all frequency values below 1. rad/s fall well inside the main lobe. Note that this signal also is used as an example in the description of DFT.

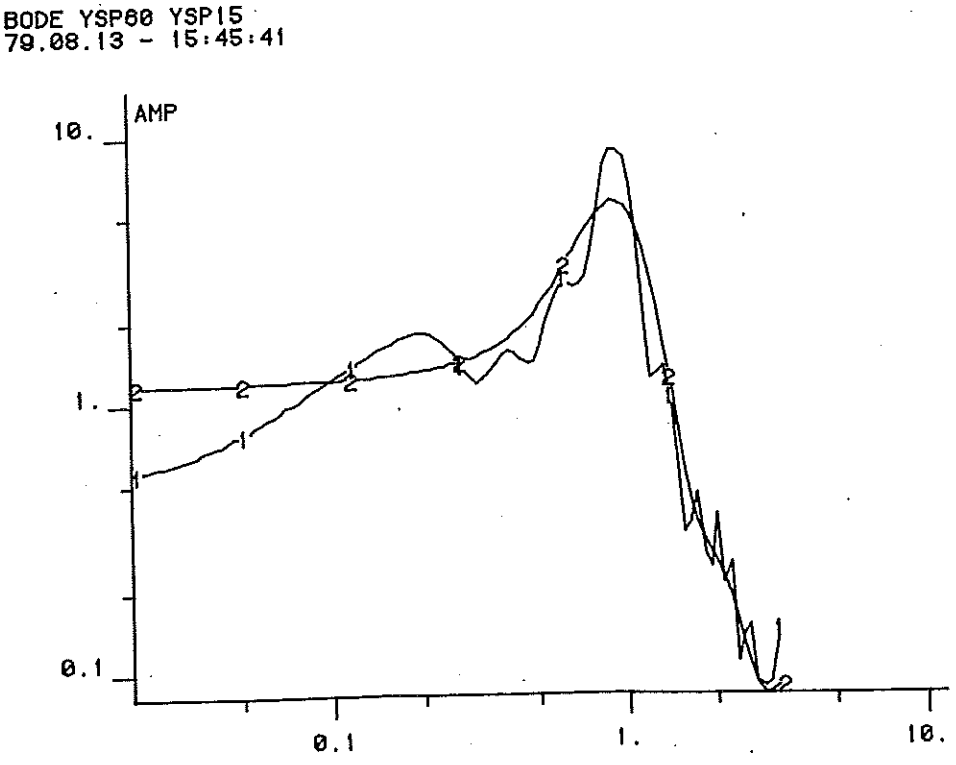


Figure 1. The autospectrum of coloured gaussian noise with a narrow (1) and a broader (2) window.

Frequency Response Operations
ASPEC

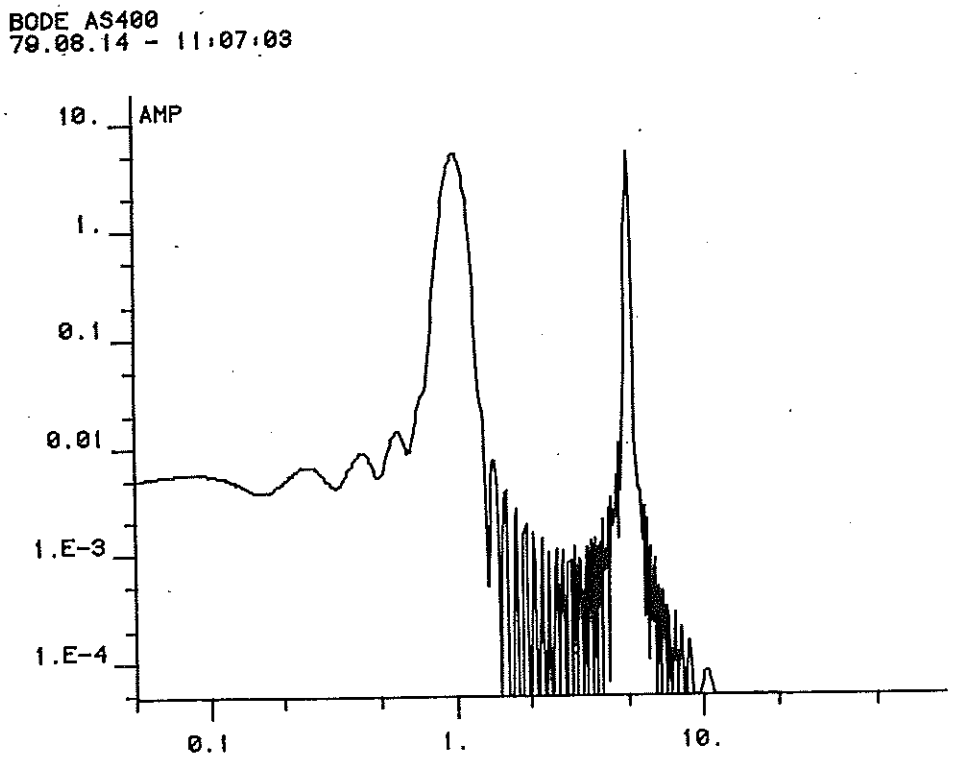


Figure 2.1. High resolution autospectrum of two sine signals with a very narrow window.

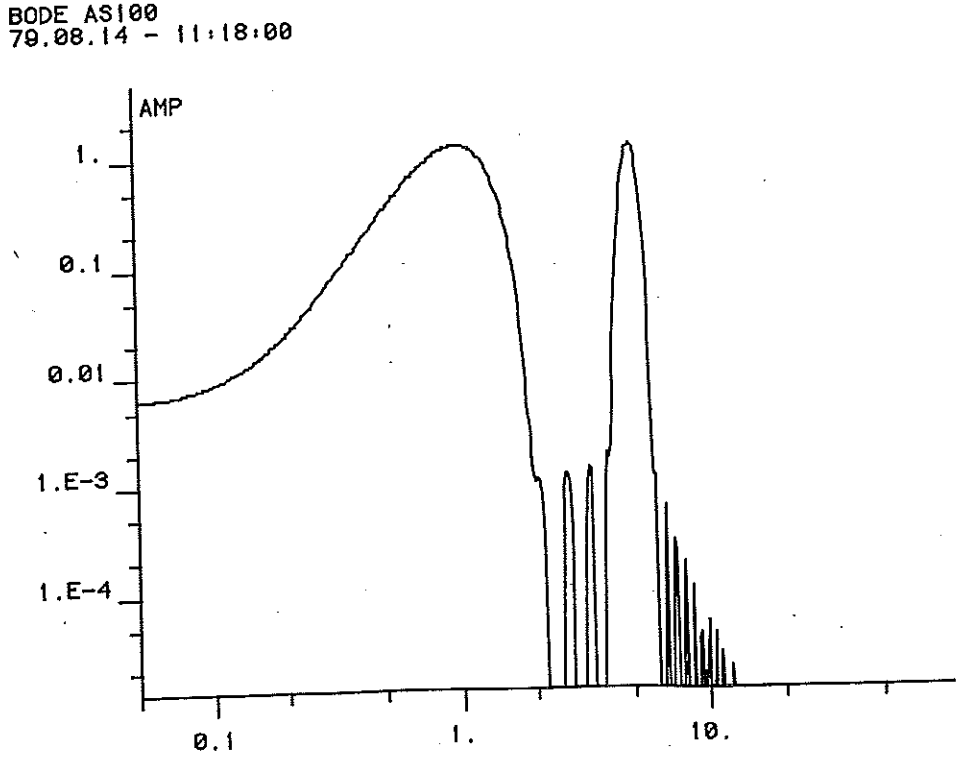


Figure 2.2. The effect of opening the window of Fig. 2.1.

Frequency Response Operations
ASPEC

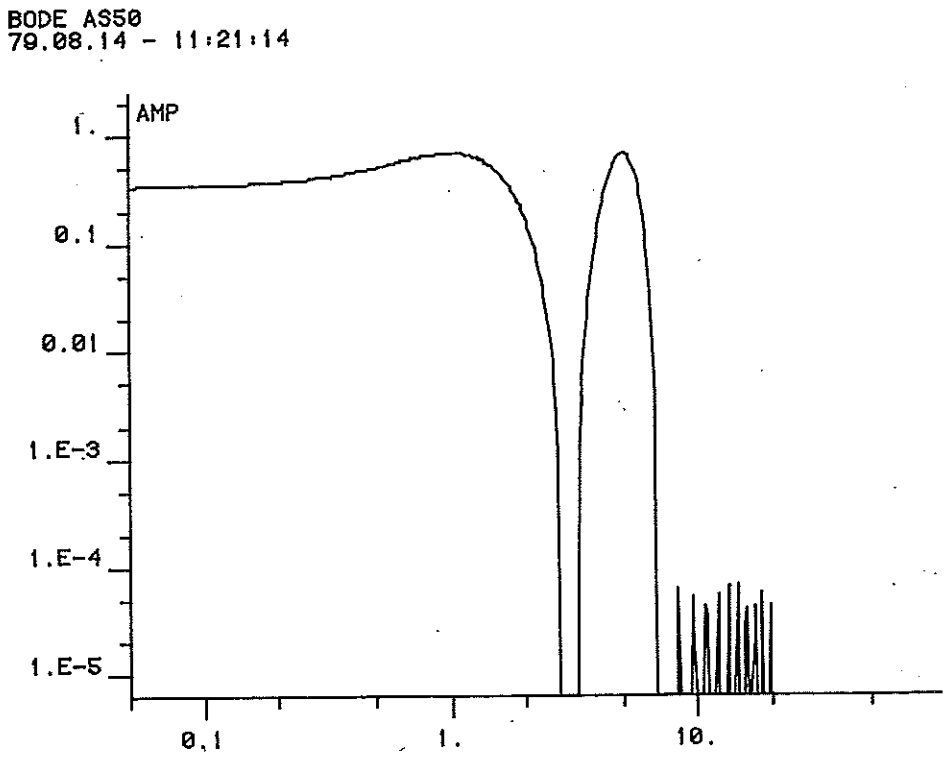


Figure 2.3. The window is now very broad.

CSPEC

Purpose

To compute the cross spectrum between two time series.

Commands

```
CSPEC FRF[(F)] < FNAM2(C21 C22) NOL [IALIGN][FREQ]
     or
CSPEC FRF[(F)] < FNAM2[(C2)] FNAM3[(C3)] NOL [IALIGN] [FREQ]
```

FNAM2,3 - time series files
C2,3 - time series indices (default: 1)
FRF - frequency response file
F - frequency response index (default: 1)
NOL - number of time lags
IALIGN - the number of lags necessary to align the processes so that the largest cross covariance is centered at zero (default 0)
FREQ - file of which the first column supplies frequencies for the computation

Function

The cross spectrum of the two input time series is computed using the cross covariances between -NOL and NOL lags. The frequency points are chosen as for ASPEC.

Method

The cross covariances are computed as described for CCOF. They are then divided into an even and an odd function:

$$EV(k) = 0.5\ [R_{xy}(k+IALIGN) + R_{xy}(-k+IALIGN)]$$

$$OD(k) = 0.5\ [R_{xy}(k+IALIGN) - R_{xy}(-k+IALIGN)]$$

$$k = 0, \ldots, NOL-/IALIGN/ = NOL'$$

These are then used to compute the real and imaginary part of the transform

Frequency Response Operations
CSPEC

$$RS(\omega) = \frac{T}{\pi}\left\{ EV(0) + 0.5 \sum_{\tau=1}^{NOL'} EV(\tau)\cos(\omega T\tau)\, W_{BH}(\tau) \right\}$$

$$IS(\omega) = \frac{2T}{\pi} \sum_{\tau=1}^{NOL'} OD(\tau)\sin(\omega T\tau)\, W_{BH}(\tau).$$

The spectrum is converted to magnitude and phase (in degrees) before it is output to FRF. ω is the frequency in rad/s, T is the sampling interval in s, $R_{xy}(\tau)$ is the cross covariance, and $W_{BH}(\tau)$ is a Blackman-Harris window.

The signal x above corresponds to the first time series in the command line, the signal y corresponds to the second one. The computed cross spectrum is denoted $\Phi_{xy}(\omega)$.

Reference

Jenkins, Watts: Spectral analysis and its applications. Holden-Day, 1968.

Cautions, Restrictions

Maximum number of time lags and maximum number of frequency intervals are 500. Data files with zero sample interval will not be accepted by the command. It is also impossible to compute the cross spectrum between two columns from data files with different sample intervals.

Hints

a) Cf. the hints for ASPEC.

b) If x is the input to a linear dynamic system and y is the output, the frequency response of its transfer function may be estimated as Φ_{xy}/Φ_{x}, an operation easily performed in FROP.

DFT

Purpose

To compute the Discrete Fourier Transform of a time series.

Command

```
DFT [(RES)] [(WND)] SPEC < DATA[(IND)] [START NSAMP]
```

RES	- result switch = 'AMP' or 'POW' (default: 'AMP')
	AMP - amplitude and phase spectrum
	POW - power spectrum
WND	- time window switch = 'BH' or 'BC' (default: 'BC')
	BH - DATA is premultiplied by Blackman-Harris's time window
	BC - the Box-Car time window is used, i.e. no premultiplication of DATA
SPEC	- name of resulting frequency response file
DATA	- name of time series file
IND	- time series index (default: 1)
START	- record number for the first sample in DATA to be included in the transform (default: 1)
NSAMP	- number of samples to transform (default: all)

Function

The discrete Fourier Transform (RES = AMP) or its absolute squared value (RES = POW) is computed using an FFT algorithm.

The arguments START and NSAMP allow the computation of the DFT based on a section only of the original time series.

If the length of the series is not an even power of 2, then zeros are appended up to the nearest such number, say 2^k.

The frequency response file will contain information for the frequencies 0 up to π/TSAMP, where TSAMP is the sampling interval of DATA. The frequency increment is such that $2^{k-1} + 1$ frequency values are obtained, linearly distributed. The complex values of the transform are expressed as magnitude and phase (in degrees).

Method

The Fourier transform of the time series x is approximated by the discrete counterpart.

Frequency Response Operations
DFT

$$X_{DFT}(\omega_k) = \sum_{n=0}^{N-1} W_{BH}(nT)\, x(nT)\, \exp(-j\omega_k nT)$$

where

$$\omega_k = \frac{2\pi}{NT} k, \qquad k = 0,1,2,\ldots,N-1,$$

N is the number of data points,
T is the sample interval,
W_{BH} is an optional Blackman-Harris window.
Note that the default Box-Car window is $W_{BC} = 1$.

The expression above can be especially efficiently evaluated if N is a power of 2, the Cooley-Tukey algorithm or the Fast Fourier Transform.

The intent of the window (if used) is to reduce the effect of approximating the infinite integral of the Fourier transform by a finitesum over a finite time interval.

References

F.J. Harris: On the Use of Windows for Harmonic Analysis with the Discrete Fourier Transform. Proc. of the IEEE, Vol. 66, No. 1, January 1978.

Cautions, Restrictions

Note that the function of the window is different from that in ASPEC and CSPEC. The form of the Blackman-Harris window is shown in Figure 1. Note that the window will severely influence the results in many cases if the time series includes transients or deterministic components.

The maximum length of the input file is 2048.

Hints

a) The command DFT may be used to compute the power spectrum (auto spectrum) of stationary signals with very high frequency resolution.

b) The command ASPEC is recommended for general use when an equalized power spectrum is desired.

c) DFT (POW) will for stochastic processes give a result similar to that obtained through ASPEC. Another way to obtain a power spectrum is with DFT (AMP) on the result of ACOF. In the latter case, do not forget to rescale to first obtain the autocovariance and do not use a window.

d) The frequency response of a linear transfer function can be computed by dividing the transform of the output (Y) with the transform of the input (X), i.e. by computing Y/X using FROP. Cf. an example below.

e) Cf. IDFT.

Examples

a) The power spectrum of a noise signal with a small sine component at $\omega = 0.8$ rad/s is shown in Figures 2.a and 2.b. The result in 2.a is obtained without windowing, i.e. the Box-Car window, while in 2.b the Blackman-Harris window was used. Note that it does not exhibit any equalizing effect.

b) The power spectrum of the sum of two sine signals with $\omega = 1$ rad/s and $\omega = 5$ rad/s is shown in Figure 3 with and without windowing. Note the marked effect of the window in decreasing spectral leakage caused by the finite length of the signal. Compare the second example in ASPEC where the same signal is used.

c) Figure 4 shows the output y and the input x of a linear dynamic system. In Figure 5 is shown the result obtained when the quotient of their respective transforms is computed (i.e. Y/X). The result is an accurate estimate of the true transfer function except at those points where very small values were encountered in X.

Frequency Response Operations
DFT

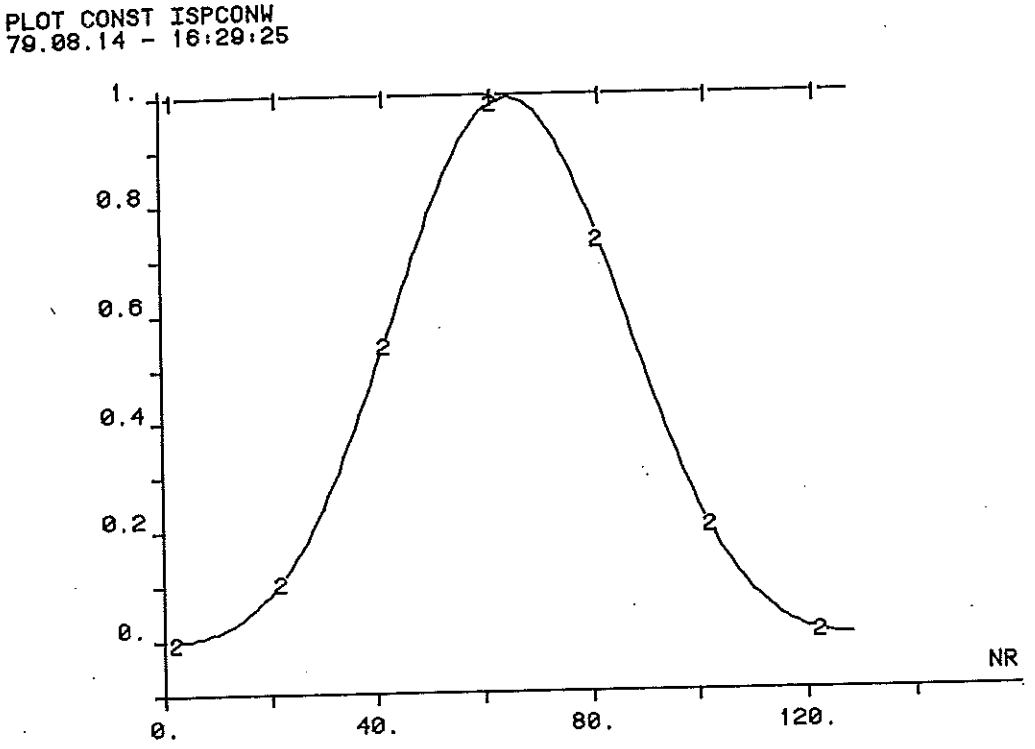


Figure 1. The shape of the Box-Car window (1) and the Blackman-Harris window (2).

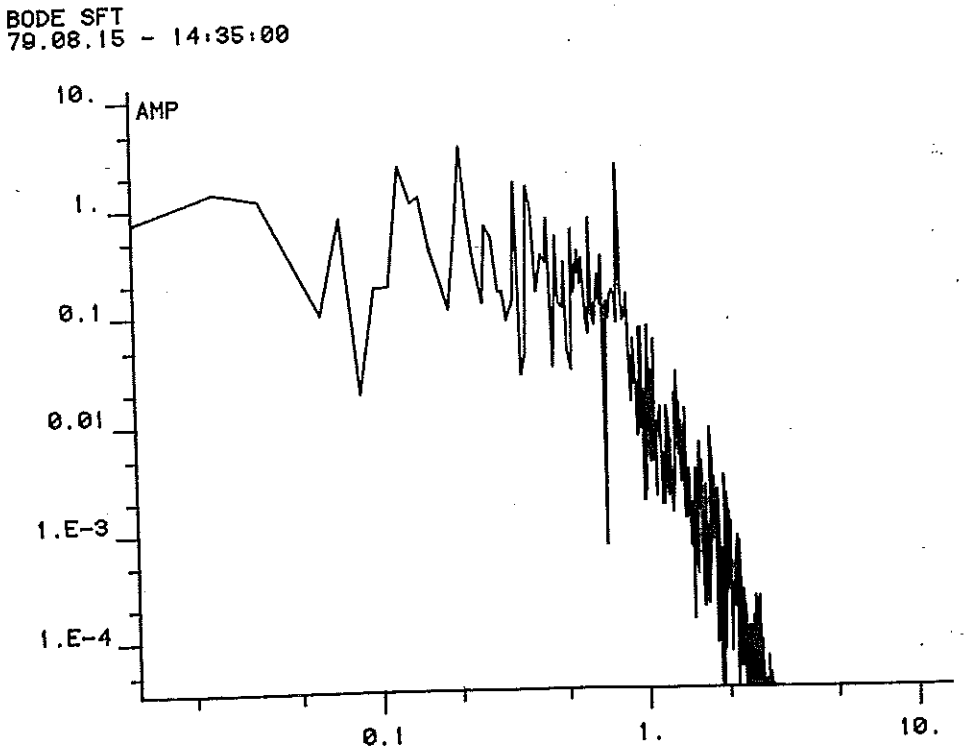


Figure 2.a. Coloured noise with a weak sine component. Box-Car window.

Frequency Response Operations
DFT

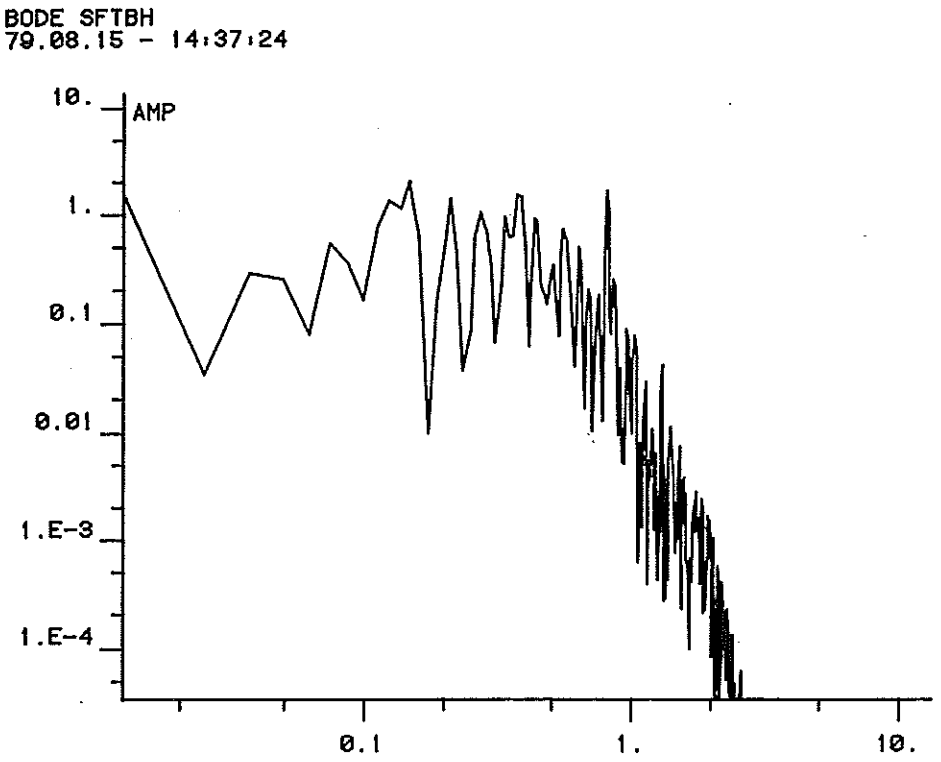


Figure 2.b. Same as in 2.a but with Blackman-Harris window.

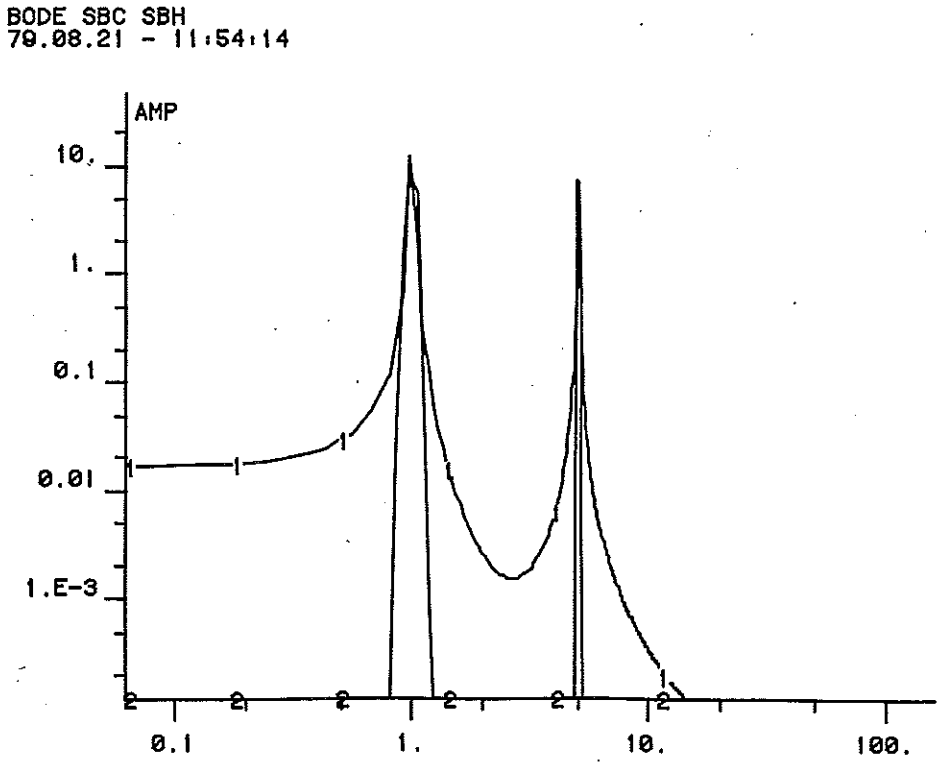


Figure 3. The effect of the Blackman-Harris window in reducing spectral leakage.

Frequency Response Operations
DFT

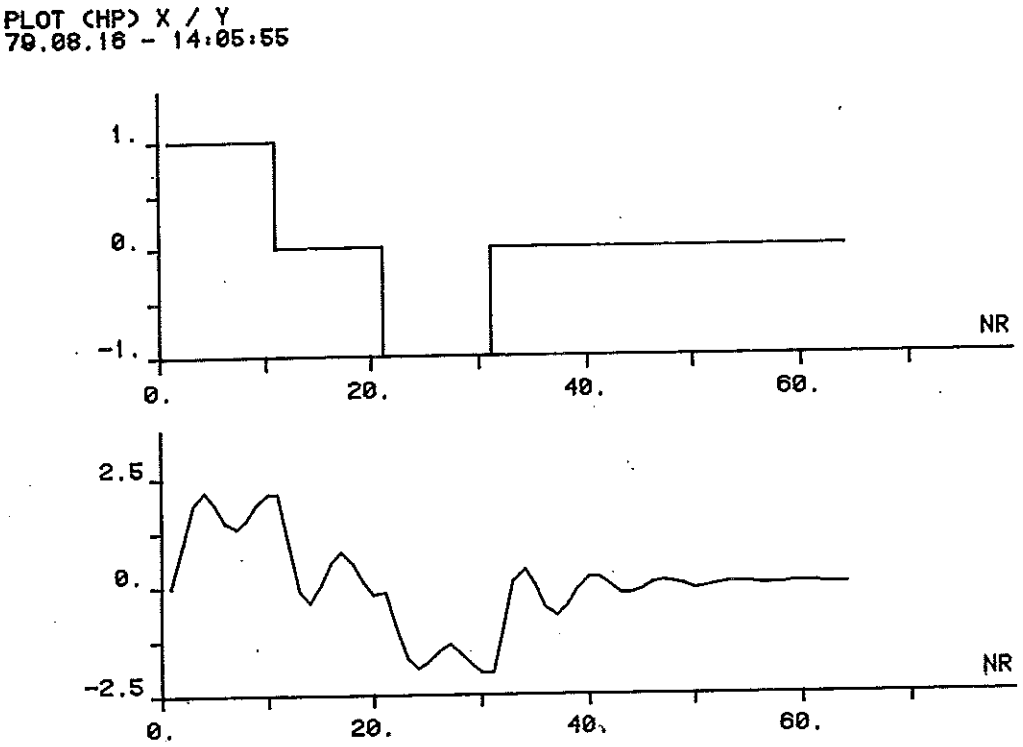


Figure 4. A test input (x) and the corresponding output (y) used to determine the transfer function of a dynamical system.

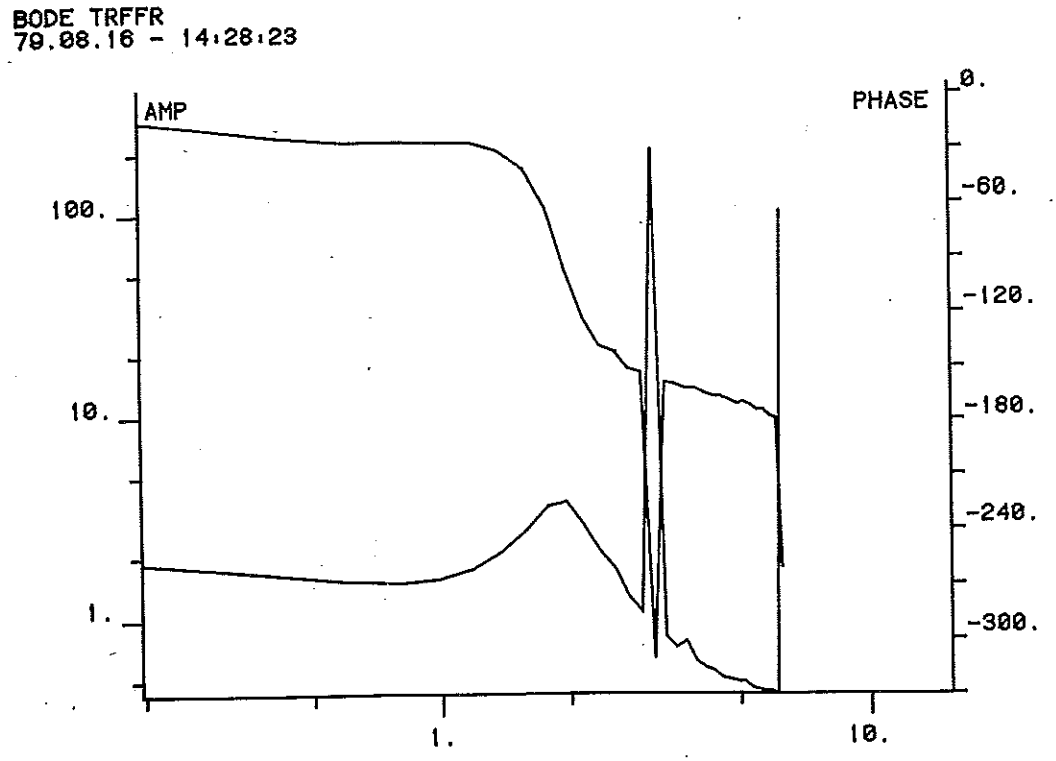


Figure 5. The result of dividing the fourier transform of the signals in Fig. 4.

FROP

Purpose

To add, subtract, multiply or divide two frequency response files element by element.

Command

FROP [FRF1[(F1)]] < FRF2[(F2)] OP FRF3[(F3)]

FRF1 - output frequency response file(default FRF2)
FRF2,3 - input frequency response files
F1,2,3 - response numbers (default 1)
OP - desired operation: '+', '-', '*', or '/'

Function

The two frequency response files FRF2 and FRF3 are read, and when two frequency points coincide, the desired operation is performed. The operands are taken as complex numbers and the result, still in amplitude and phase format (i.e. $z = r \cdot e^{i\varphi}$) is output to FRF1. An effort is made to keep the phase continuous across the 360^{o} boundaries.

Hints

See hints for CSPEC for a possible use of FROP.

IDFT

Purpose

To compute the Inverse Discrete Fourier Transform of a frequency response.

Command

IDFT DATA < SPEC[(IND)]

DATA - name of resulting time series file
SPEC - name of frequency response file
IND - frequency response index (default: 1)

Function

The inverse of the Discrete Fourier Transform is computed. The input frequency response SPEC should contain complex number in magnitude and phase form and the frequency information must be linearly increasing starting at zero frequency (such as generated by DFT). The length of SPEC should be of the form $2^k + 1$. If it is not, the input is implicitly augmented by zeros to the nearest such number.

The resulting time series DATA. will have a sampling interval π/WMAX where WMAX is the highest frequency in the input (after a possible augmentation with zeros) DATA will consist of 2^{k+1} points.

IDFT is the inverse of DFT in the sense that after

```
>DFT (AMP) (BC) SP < DATA1
>IDFT DATA2 < SP
```

the time series DATA1 and DATA2 will be identical.

Method

The expression

$$x(k) = \frac{1}{N} \sum_{n=0}^{N-1} X(\omega_n) \exp(j\omega_n kT)$$

where

Frequency Response Operations
IDFT

$$\omega_n = \frac{2\pi}{NT} n$$

is evaluated using the Fast Fourier Transform technique.

Cautions; Restrictions

Note that the inverse transform of a power spectrum gives the autocovariance function and not the signal itself and that no window should be used in computation (cf. DFT) of frequency responses later to be used in IDFT.

The maximum length of the input file is 1025.

The global variable TICK. must have been given a suitable value so that the sample interval of the resulting time series can be represented.

Hints

a) Signals may be filtered using any filter represented as amplitude and phase (i.e. not necessarily physically realizable; by computing the transform; operating upon it (e.g. using FROP) and inverse transforming the result. Be careful; however; not to violate the requirements of the sampling theorem.

b) It is possible to interpolate in a time series by extending its transform with zeros and inverse transforming the result. This is done in the example.

Example

In Figure 1 is a time series S1 shown with a sampling interval of 1 s. It is desired to construct a signal (S2) with sample interval 0.5 s; coinciding with S1 at the sample points of S1. This can be done as follows (in the example; S1 contains 8 sample points):

```
>DFT S1S < S1                "TRANSFORM THE SIGNAL
>INSI FREQ 9
    >RAMP 0. 0.78539         "3.14159/4 = 0.78539
    >X
>INSI T 4
    >ZERO
    >ZERO
    >ZERO
    >X
>CONC S2S < S1S T            "EXTEND FREQUENCY RESPONSE
>MOVE S2S(1) < FREQ          "INSERT FREQUENCY INFORMATION
```

Frequency Response Operations
IDFT

```
>FHEAD S2S                    "MAKE IT A FREQUENCY RESPONSE
    >9 1
    >X
>IDFT S2 < S2S                "BACK TRANSFORM
>PLOT S1 / S2                 "RESULT SHOWN IN FIG. 1
```

It may be worth noting that the original signal (S1) must have been properly sampled (i.e. contain no power at the Nyquist frequency) for this operation to give the indicated result.

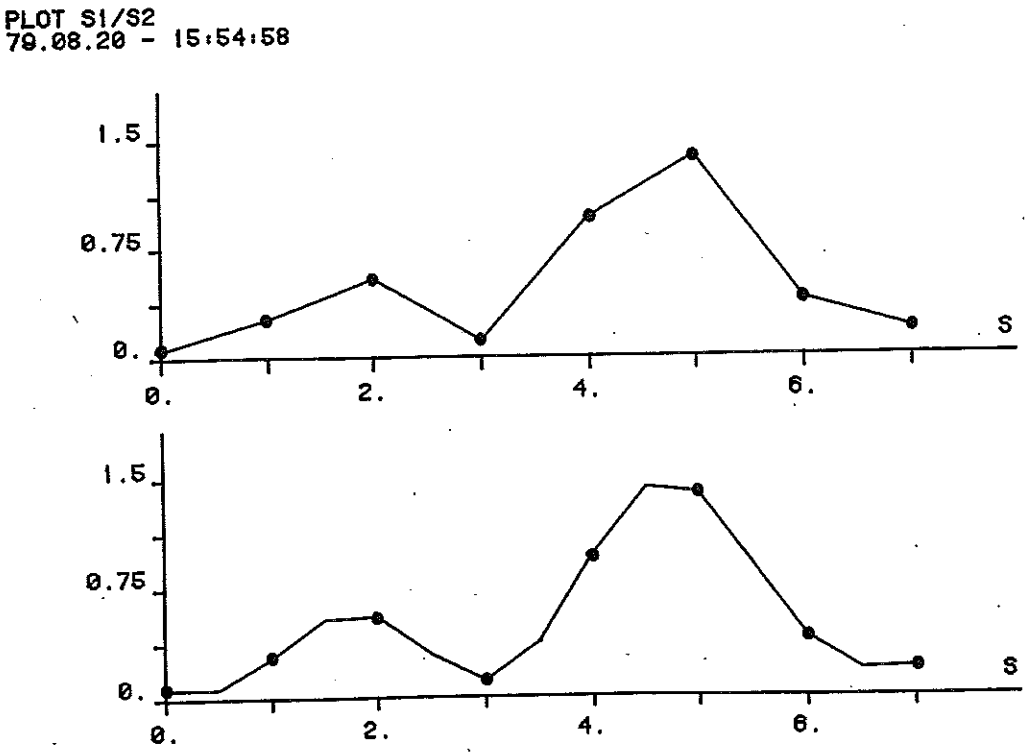


Figure 1. The indicated points coincide exactly.

DETER

Purpose

To simulate the Discrete MISO Transfer function using the deterministic part only.

Command

```
DETER DNAM1[(C1)] < SNAME[(NAME)] DNAM2[(C21 ..)]
      [DNAM3[(C31 ..)] [....]..] [NP]
```

DNAM.. - data file
C.. - column numbers within DNAME
(default: C.. = 1, 2, ..)
SNAME - system file name
NAME - name of section within PNAME
NP - number of points wanted (default the number of points in the shortest input file)

Function

The Discrete MISO Transfer function described in SYST (optionally section NAME) is simulated using the input U columns C21... . Only the deterministic part is simulated, even if the system contains a stochastic part. The single output is written to Y(C1).

Method

The simulated equation is

$$y(t) = \sum \frac{B_i(q^{-1})}{A_i(q^{-1})} u_i(t)$$

The A_i polynomials may be replaced by a single A-polynomial, in which case initial values for the output may be used (other than the default zeros).

Cautions: Restrictions

A maximum of 3000 points may be simulated.

Hints

a) Cf. the command DSIM.

b) Use DETER to compute the deterministic output from a model obtained in ML. Cf. the command ML.

Example

An often useful way of assessing the validity of a model is to compute the deterministic model output, i.e. to simulate the model with the recorded input signal but without (the unknown) noise input, or the error between the measured output Y and the model output YMOD.

This is done below for the model ML2 and the input signal PRBS, used in the example in the ML command.

```
>DETER YMOD < ML2 PRBS
>VECOP DETERR < Y-YMOD
>PLOT Y YMOD
>PLOT Y/DETERR
```

The first plot is shown in Figure 1, the second one in Figure 2.

Simulation & Model Analysis
DETER

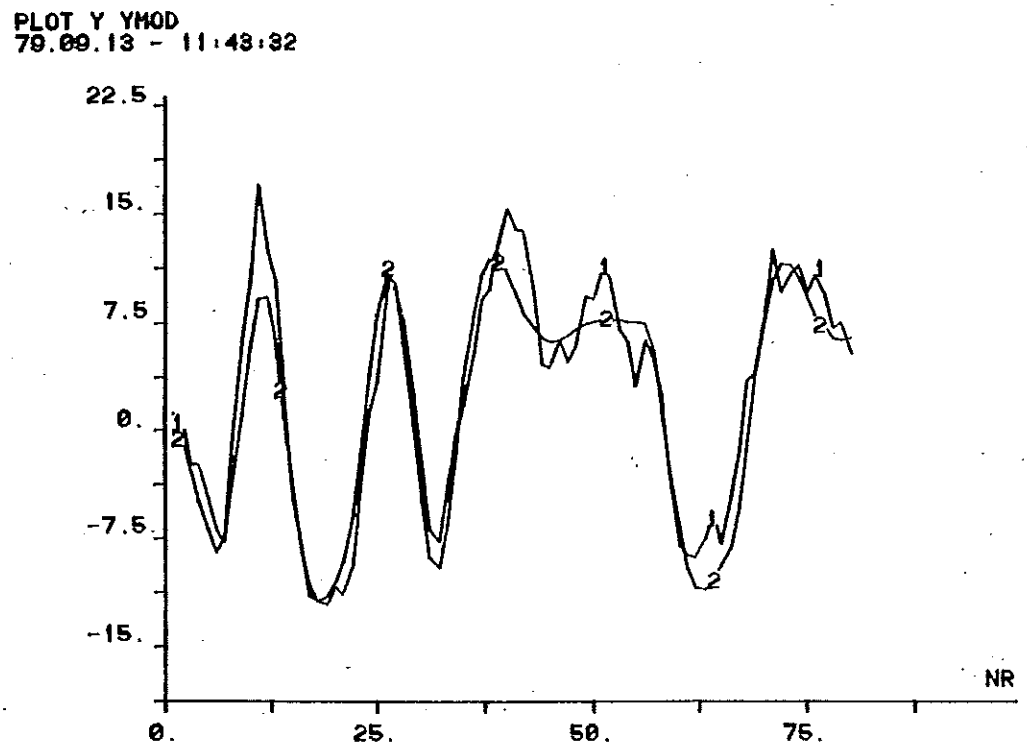


Figure 1. A plot of the recorded output Y and the model output YMOD.

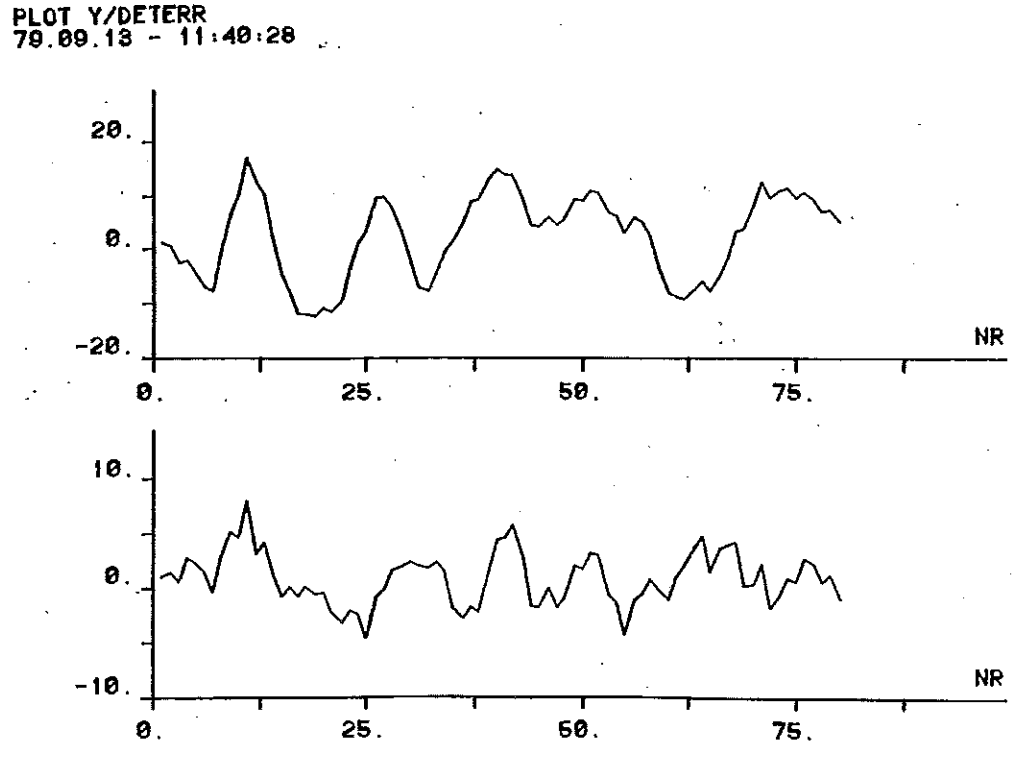


Figure 2. A comparison of Y and the model error.

DSIM

Purpose

To simulate the Discrete Miso Transfer function using both deterministic and stochastic inputs.

Command

```
DSIM DNAM1[(C1)] < SNAME[(NAME)] DNAM2[(C21 ..)]
     [DNAM3[(C31 ..)] [....]..] [NP]
```

DNAM..	- data file
C..	- column numbers within DNAME (default: C.. = 1, 2, ..)
SNAME	- system file name
NAME	- name of section within PNAME
NP	- number of points wanted (default the number of points in the shortest input file)

Function

The discrete MISO Transfer function described in SYST (optionally section NAME) is simulated using the two input files U1 columns C11... and U2 columns C21... . The first inputs are used for the deterministic part, the latter for the stochastic part. The single output is written to Y(C1).

Method

The simulated equation is

$$y(t) = \sum \frac{B_i(q^{-1})}{A_i(q^{-1})} u_i(t) + \sum \lambda_i \frac{C_i(q^{-1})}{D_i(q^{-1})} e_i(t).$$

In many cases the A_i and D_i polynomials are identical in which case a single A polynomial may be used in the system description. Initial values of the output may be included in the system description, if not zeroes are assumed.

Simulation & Model Analysis
DSIM

Cautions: Restrictions

A maximum of 3000 points may be simulated. Non-zero initial output values requires a single A-polynomial.

Hints

Use the command DETER when a simulation of the deterministic part only is required.

FILT

Purpose

To compute digital Butterworth filters of given order with given cut-off frequencies.

Command

FILT SNAME < FITYP NO DELTAT OMCO

SNAME - filter system file name
FITYP - filter type = 'LP'/'HP'
  LP - low-pass filter
  HP - high-pass filter
NO - order of the filter response (max 5)
DELTAT - sample period (s)
OMCO - cut-off frequency (rad/s)

Function

Parameters for a filter of type FITYP and order NO with cut off frequency OMCO are computed and written into the system file SYST as A- and B-polynomials.

Method

The continuous time transfer function of a Butterworth filter of the desired type is computed with the cut off frequency ω_d. Then a discrete time counterpart is computed using the bilinear transform

$$s = \frac{2\,(1 - z^{-1})}{T\,(1 + z^{-1})}$$

Due to the frequency warping of the bilinear transform, ω_d is computed as

$$\omega_d = \frac{2}{T} \tan\left(\frac{\omega T}{2}\right),$$

where ω is the desired cut off frequency (OMCO).

Simulation & Model Analysis
FILT

Reference

R.M. Golden, J.F. Kaiser: 'Design of wideband sampled data filters', in Bede Liu (editor): Digital filters and the fast Fourier Transform, Halsted press, 1975.

Cautions, Restrictions

Maximum filter order is 5. A cut off frequency higher than π/T would violate the sampling theorem and is thus rejected.

Hints

a) Cf. the commands SPTRF and BODE which may be used to visualize the filter transfer function.

b) To actually filter a time series, the filter obtained in FILT is used in the command DETER to actually operate upon the signal.

c) To obtain the effect of a band pass filter, the signal is filtered twice; in an LP and an HP filter.

d) Note the possibility to use DFT and IDFT to perform filtering operations.

RANPA

Purpose

To generate a new Discrete MISO Transfer function with random parameters (coefficients). The mean values and covariance matrix of the new parameters are taken from a given system description.

Command

RANPA PNAM1 < PNAM2[(NAME)]

PNAM1 - name of output system file
PNAM2 - name of input system file
NAME - name of section within PNAM2

Function

A new system description file with name SNAM1 is generated. The original system description must include a covariance matrix. Initial values for the output, if included, are not changed.

Method

The given covariance matrix R is factored

$$R = SS^T.$$

Then a vector of normally distributed random variables e is generated giving

$$v = S^T e. \qquad \text{(Note: } Ev^T v = Ee^T SS^T e = R\text{)}$$

Then the new parameters Θ_N are generated from the old Θ_O (i.e. the mean values) through

$$\Theta_N = \Theta_O + v.$$

Cautions, Restrictions

The global variable NU. is used and altered by the random number generator.

Hints

a) The covariance matrix is usually generated by using the subcommand SAVE COMAT in ML or LS.

b) RANPA can be used to determine the effect of uncertainties in an identification result. For example step or impulse responses for different sets of parameters give an indication of which of the DC gain and the high frequency properties of the model is the most accurate.

Example

In the example found in the description of the ML command, the covariance matrix was included in the second order model. The following macro definition and command sequence utilizes RANPA to obtain a number of step responses illustrating the amount and character of uncertainty in the model.

```
>MACRO RAND Y < SYS U NL
  >FOR I = 1 TO NL
  >RANPA ZP < SYS
  >DETER Y(I) < ZP U
  >DELET (T) ZP
  >NEXT I
  >PLOT (NM) Y
  >END
>LET NPLX. = 30
>INSI STEP NPLX.
  >STEP
  >X
>RAND YSTEP < ML2 STEP 5
```

This command sequence produces Figure 1.

Simulation & Model Analysis
RANPA

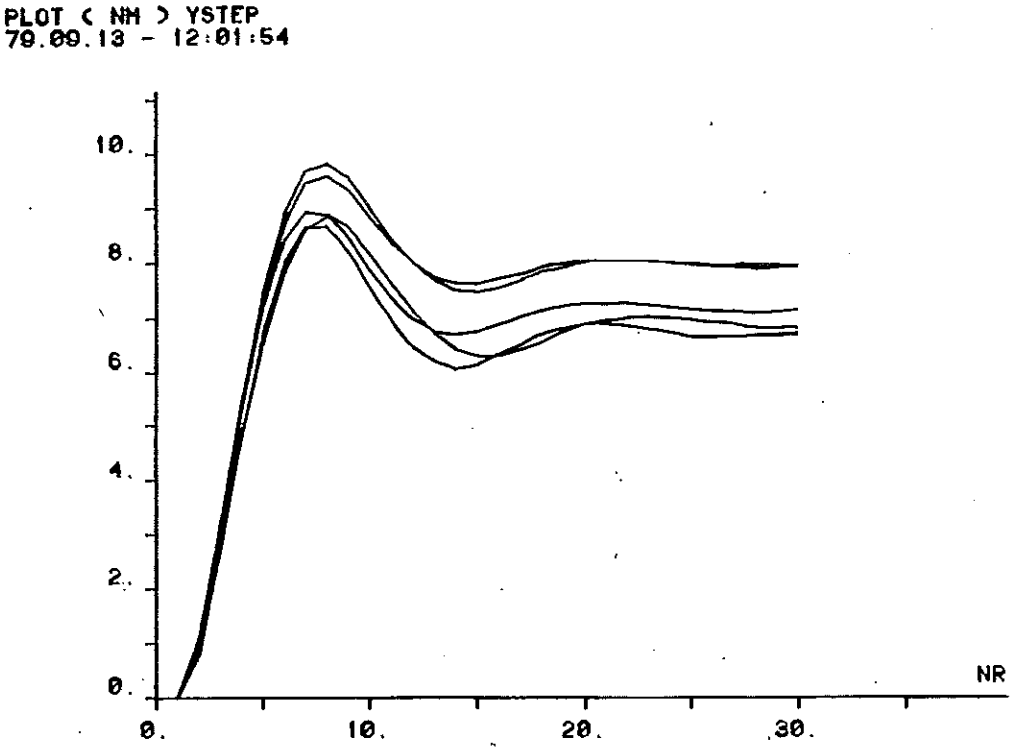


Figure 1. Step responses illustrating the uncertainty in the model in the example.

RESID

Purpose

To compute the residuals of a system for a given input and output, and to perform various statistic tests on the results.

Command

```
RESID RES[(C1)] < SYST[(NAME)] DATA[(C11 C12 .. )]
                                [NOL [NFREE]] [EXT]
```

RES	- residual file name
C1	- column number for RES (by default C1=1)
SYST	- system file name
NAME	- name of section within SYST
DATA	- data file name
C11..	- column numbers for DATA (by default C11=1, 2, ..)
NOL	- number of lags for correlation computations (by default NOL = 10)
NFREE	- number of degrees of freedom for the independence tests (by default NFREE = 10)
EXT	- name extension for the test quantities

Subcommands

KILL	- return immediately to the main program
PAGE	- view the next page on the display
TABLE	- write a table of the correlation function associated with the last 'PAGE' on the display

Function

The residuals are computed using the specified data as inputs and outputs of the system SYST. Note that the output must be the last column.

Apart from the file RES that receives the computed residuals, the results of RESID are displayed on the computer terminal in the form of test quantities and plotted correlation functions. This information is described below.

The displayed information is divided into pages that are displayed according to subcommands. This output can be inhibited using the switches GRAPH and TEXT in the TURN command.

The values of some test quantities are transferred to the global variables

ARES.EXT for the independence of the residuals;

NCRESi.EXT for the independence of residuals and input i for negative lags;

PCRESi.EXT for the independence of residuals and input i for positive lags;

if they exist. EXT is the extension name of the global variables and should be given in the argument string. The displayed information is output on line printer if the global variable PRINT. = 1. If PRINT. = 2, the absolute and cumulative frequencies from the normality test are printed as well.

Method

a) The residuals are computed using the formula

$$\varepsilon(t) = \frac{A(q^{-1})}{C(q^{-1})} y(t) - \sum \frac{B_i(q^{-1})}{C(q^{-1})} u_i(t)$$

b) The normality of the residuals is tested by a chi-square goodness-of-fit test. The observations are grouped into K intervals forming a frequency histogram. The observed frequency in the i:th class interval is called f_i and the expected frequency for a normal distribution is called F_i. The quantity

$$\chi^2 = \sum_{i=1}^{K} \frac{(f_i - F_i)^2}{F_i}$$

is approximately $\chi^2(K-3)$. The number of class intervals K is chosen automatically depending on the number of observations.

c) The number of changes of sign of the residuals is computed. This number is compared with the expected one, knowing the changes of sign to be normally distributed

$$N\left\{\frac{M-1}{2}, \frac{\sqrt{M-1}}{2}\right\},$$

where M is the number of observations.

d) To test the independence of the residuals

$$\sum_{1}^{NFREE} r_{\varepsilon}^{2}(\tau)$$

is computed. This quantity is χ^2(NFREE).

e) To test the independence between the residuals and the input(s) $Q = x^T P^{-1} x$ is computed from the cross correlations between u and ε and from the autocorrelations of u.

$$x = \begin{bmatrix} r_{u\varepsilon}(j) \\ \vdots \\ r_{u\varepsilon}(j+m-1) \end{bmatrix}$$

$$P = \frac{1}{N} \begin{bmatrix} r_{uu}(0) & \cdots & r_{uu}(m-1) \\ r_{uu}(1) & r_{uu}(0) & \cdots & r_{uu}(m-2) \\ r_{uu}(m-1) & \cdots & r_{uu}(0) \end{bmatrix}$$

$$r_{u\varepsilon}(\tau) = E\,[u(t)\,\varepsilon(t+\tau)]$$

Q is asymptotically χ^2(m). If P is singular, u is not persistently exciting, and a message to that effect is output.

For positive τ, j is chosen equal to n+1, where n is the order of the system description in SYST[(NAME)]. m is chosen to NFREE. The test quantity is also computed for negative lags, τ = -NFREE + 1, ..., 0.

f) The autocorrelation and cross correlation functions are plotted up to NOL lags together with the 95 % confidence limits [= ±1.96(1/ M)] indicating the region inside which the estimates of the correlations should lie if the residuals (and the input(s)) are independent.

Reference

J.S. Bendat, A.G. Piersol: Measurement and analysis of random data, Wiley, New York, 1966.

Cautions, Restrictions

The length of the input file must not exceed 3000 samples.

Hints

a) The test quantity for negative τ can indicate whether the original system is under feedback or not. A large test quantity may indicate feedback.

b) If the system file describes a unit system, i.e. contains A- and C-polynomials of order zero, the independence and normality of a time series may be tested.

c) Some (approximate) values for the CHI-SQUARE statistic follows: (Definition $P(\chi^2 > \ell) = 0.05$.)

Degrees of freedom	ℓ
5	11
10	18
15	25
20	31
25	38
30	44

Example

Figures 1 and 2 show the displayed output from using RESID on the 1:st respectively the 2:nd order model obtained in the example of the use of the ML command. Note in Figure 1 that the 1:st order model fails to make the residuals white and independent of the input. This is apparent both from the graphs and from the test quantities supplied. The 2:nd order model of Figure 2, however, produces residuals that passes all tests, hence we conclude that this is the appropriate model order.

Simulation & Model Analysis
RESID

VARIANCE OF THE RESIDUALS:
6.44606

NUMBER OF CHANGES OF SIGN
OF THE RESIDUALS: 98

5 PERCENT TOLERANCE LIMITS:
85 113

TEST OF INDEPENDENCE OF THE
RESIDUALS

E(RES(T)*RES(T+TAU))
FOR: 0 < TAU < 11

TEST QUANTITY: 31.8239
DEGREES OF FREEDOM: 10

TEST OF NORMALITY

TEST QUANTITY: 6.31177
DEGREES OF FREEDOM: 17

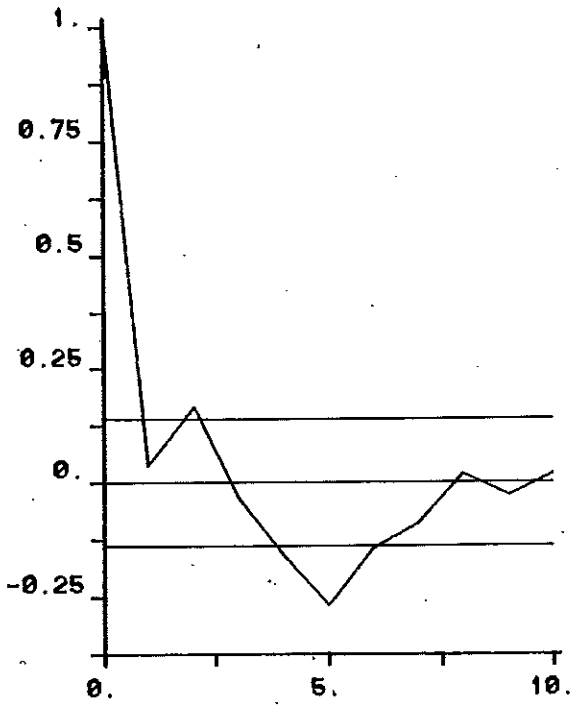


Figure 1.a. Test page 1 on the model ML1.

Simulation & Model Analysis
RESID

TEST OF INDEPENDENCE BETWEEN
RESIDUALS AND INPUT: 1

E(RES(T):U(T+TAU))
FOR: 1 < TAU < 12

TEST QUANTITY: 54.2076
DEGREES OF FREEDOM: 10

E(RES(T)*U(T+TAU))
FOR: -10 < TAU < 1

TEST QUANTITY: 24.5215
DEGREES OF FREEDOM: 10

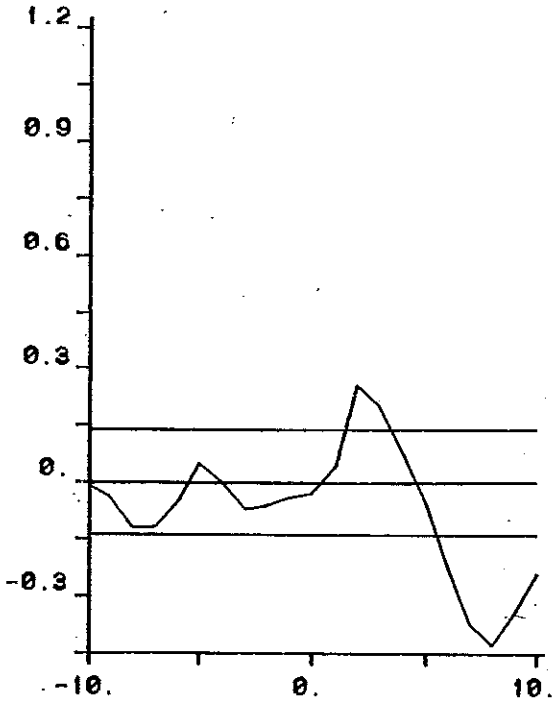


Figure 1.b. Test page 2 on the model ML1.

Simulation & Model Analysis
RESID

VARIANCE OF THE RESIDUALS:
3.44872

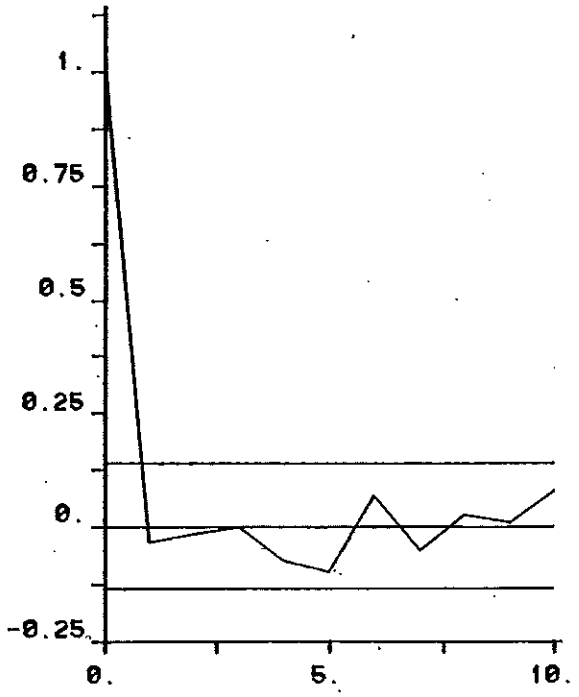
NUMBER OF CHANGES OF SIGN
OF THE RESIDUALS: 100

5 PERCENT TOLERANCE LIMITS:
85 113

TEST OF INDEPENDENCE OF THE
RESIDUALS

E(RES(T)*RES(T+TAU))
FOR: 0 < TAU < 11

TEST QUANTITY: 6.11162
DEGREES OF FREEDOM: 10



TEST OF NORMALITY

TEST QUANTITY: 10.4630
DEGREES OF FREEDOM 17

Figure_2.a. Test page 1 on the model ML2.

Simulation & Model Analysis
RESID

TEST OF INDEPENDENCE BETWEEN
RESIDUALS AND INPUT: 1

E(RES(T)*U(T+TAU))
FOR: 2 < TAU < 13

TEST QUANTITY: 2.68452
DEGREES OF FREEDOM: 10

E(RES(T)*U(T+TAU))
FOR: -10 < TAU < 1

TEST QUANTITY: 16.0164
DEGREES OF FREEDOM: 10

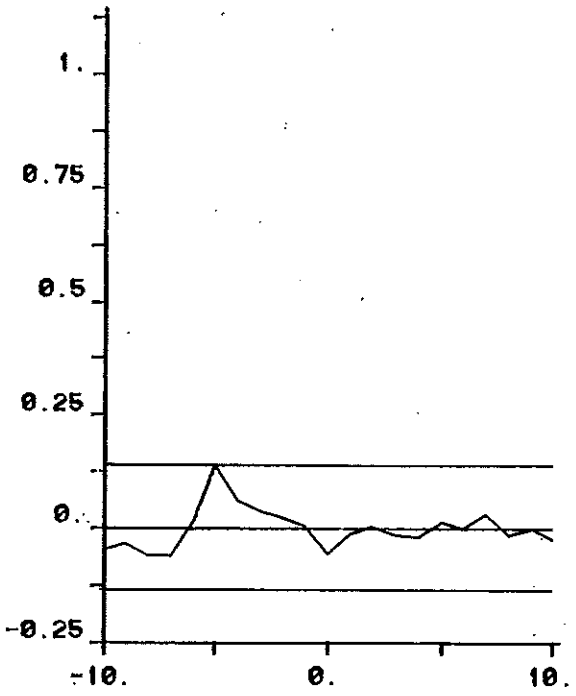


Figure 2.b. Test page 2 on the model ML2.

SPTRF

Purpose

To compute the frequency response of a discrete time transfer function. Either the amplitude and phase or the absolute squared values are computed.

Command

```
SPTRF [(SW)] FRF[(F1)] < SYST[(NAME)]
      TPN[(NRN)] / TPD[(NRD)] [FREQ[(F2)]]
```

SW	- = 'POW'/'AMP', switch choosing a power spectrum or an amplitude and phase computation (default: 'AMP')
FRF	- frequency response file
F1	- frequency response number (default value 1)
SYST	- system file name
NAME	- section name of system file
TPN	- numerator polynomial type (A, B, C or D)
NRN	- numerator polynomial number (default 1)
TPD	- denominator polynomial type (A, B, C or D)
NRD	- denominator polynomial number (default 1)
FREQ	- file with frequency values
F2	- column number (default value 1)

Function

The transfer function of interest is specified as the quotient of two polynomials of the system file in the argument list.

The computation is performed for NOF. logarithmically distributed frequency values in the interval WMIN. to min(WMAX.,π/T), where T is the sample interval of the system. NOF., WMIN., and WMAX. are global variables.

If the file FREQ is included, the frequency values are taken from column 1 of this file, but they are still subject to the constraints given above.

Method

The amplitude and phase of

$$H(e^{j\omega T})$$

or

$$| H(e^{J\omega T}) |^2$$

are computed. The transfer function H is

$$H = \frac{TPN(NRN)}{TPD(NRD)}$$

with the notation used in the argument list.

Cautions: Restrictions

Note that if H represents the factorized spectral density of a stochastic process x, i.e.

$$\Phi_x = | H |^2$$

then the result of SPTRF should be multiplied by T/π, where T is the sampling interval. Cf. Hint e) in ASPEC.

LS

Purpose

To compute a least-squares model using the square-root algorithm. The model is a multiple input - single output model of the form:

$$A(q^{-1})\, y(t) = \sum_{i}^{NU} q^{-K_i} B_i(q^{-1})\, u_i(t) + e(t)$$

with deg A = NA, deg B_i = NB(i), and K_i = KB(i). The model structure should be given in the command STRUC. The initial computations should have been done using the command SQR.

Command

LS [(SW)] SYST[(SECT)] < SFIL [EXT]

SW - switch for subcommands:
SW= 'SC' if subcommands wanted
SW= 'VOID' or else omitted
SYST - system file for result
SECT - optional section name within SYST
SFIL - input structure file name
EXT - those of the global variables AIC.EXT and V.EXT that exist as reals will receive the values of Akaike's test quantity and the loss function, respectively.

Subcommands

SAVE STDEV - uncertainties are saved in SYST file
SAVE COMAT - covariance matrix is saved in SYST file
KILL - resume main command mode, produce no results
X - produce results, then resume main command mode

Function

The input structure file contains:

a) The name of the file containing the R-matrix computed by SQR. See also equation (3) below.

b) The structure variables NA(MAX), NU(MAX), NBi(MAX), KBi(MAX), and KBi(MIN), defining the meaning of the elements in the R-matrix.

c) Information in the form of actual polynomial degrees or indices of fixed parameters telling which parameters that are not to be estimated.

Substituting backwards in equation (4) below, it is then possible to compute the desired parameters, with the additional precaution described.

The estimated parameter values are output to the system file together with some extra information. Some is optional, controlled by subcommands, while other is always given, e.g. the value of the loss function and Akaike's test quantity (AIC). On the user's terminal, a table is given with the value of the loss function and the AIC for each additional discarded parameter. The displayed information is also printed depending on the reserved global variable PRINT.

Method

For each measurement point (less some points at the beginning of the series) the quantity e(t) is observed:

$$y(t) + \sum_i a_i y(t-i) - \sum_{ij} b_{ij} u_i (t-j-k_i) = e(t)$$

Using normal symbols, the complete set of observations can be expressed in the equation (see the reference below)

$$Y - \varphi\Theta = E \qquad (1)$$

where Θ contains the unknown parameters while Y and φ contains the observations. The parameters Θ are now to be computed so as to minimize the loss function $V = 1/2\ E^T E$. If Q is an orthogonal matrix, i.e. $Q^T = Q^{-1}$, we have

$$E'^T E' = (QE)^T QE = E^T Q^{-1} QE = E^T E.$$

Now, rewriting (1) using partitioned matrices and multiplying by Q from the left, we have:

$$Q\ [\varphi\ Y] \begin{bmatrix} -\Theta \\ 1 \end{bmatrix} = QE = E' \qquad (2)$$

Q is then chosen so that

$$Q\,[\varphi\ Y] = \begin{bmatrix} R \\ 0 \end{bmatrix}. \qquad (3)$$

The upper-right triangular matrix R of order n+1, n being the number of parameters, is computed using Householder transformations. The steps between 1 and 3 are performed by the command SQR.

Using (3) Equation (2) can now be rewritten omitting the zeros (i.e. rows n+2, n+3, ...):

$$\left[\begin{array}{c|c} & r_{1\ n+1} \\ & r_{2\ n+1} \\ R_n & \vdots \\ & r_{n\ n+1} \\ & r_{n+1\ n+1} \end{array}\right] \left[\begin{array}{c} -\Theta_1 \\ -\Theta_2 \\ \vdots \\ -\Theta_n \\ 1 \end{array}\right] = \left[\begin{array}{c} e'_1 \\ e'_2 \\ \vdots \\ e'_n \\ e'_{n+1} \end{array}\right] \qquad (4)$$

Now, by direct backwards substitution Θ_i can be computed yielding $e'_1 = e'_2 = \ldots = e'_n = 0$. Thus the minimal value of the loss function is given by

$$V_n = \frac{1}{2} E^T E = \frac{1}{2} E'^T E' = \frac{1}{2} e'^2_{n+1} = \frac{1}{2} r^2_{n+1\ n+1}.$$

The heavy part of the computations, viz. the triangularization (3) is done in the command SQR allowing for the maximum number of parameters. If, however, actual values in the structure file indicates that some parameters are to be discarded or fixed, the corresponding columns of R_n are eliminated followed by a new triangularization giving a new R-matrix, R_m, $m<n$. For each discarded parameter, the corresponding loss function $V_m = 1/2\ r^2_{m+1\ m+1}$ and the value of AIC is computed and displayed.

Parameters are discarded beginning with the first B-parameter, corresponding to increasing delay (K-value). Then the last B-parameters and A-parameters are discarded, decreasing the order of the respective polynomial.

Then, for the required model, the noise intensity, the covariance matrix and the parameter accuracies are computed as well as Akaike's test quantity (AIC). Cf. the command ML.

$$\lambda^2 = \frac{2}{N} V_m$$

$$\sigma^2_{B_i} = \lambda^2 \, [R_m^T \, R_m]^{-1}_{ii}$$

$$AIC = N(\ln 2\pi + 2 \ln \lambda) + 2m$$

Reference

Astrom, K.J.: Lecture notes on identification.

Hints

a) Note that the main part of the calculation is performed only once, in the SQR command.

b) Note that the order in which parameters are discarded allows an easy way of determining the delay parameter k, also for systems with coloured noise, cf. the example below.

c) The intended mode of operation is to start with a full range of parameter values also allowing a range in the delay k. Then STRUC and LS are used alternatingly in order to discard parameters achieving a model with the lowest allowable number of parameters. The AIC helps in this operation.

d) It is possible, sacrifying some flexibility, to write a macro that allows STRUC, SQR, and LS to be used together in a fashion similar to the ML command:

```
MACRO LSID MODEL < DATA N
STRUC STRF
NA MAX N
NU MAX 1
NB MAX N
SQR RMAT < DATA STRF
LS MODEL < STRF
END
```

Example

In this example, we use roughly the same data WRK as in the ML example, only that there is a time delay in the system and that the noise level is lower. Hence we know or suspect that the noise is correlated, thus violating a basic assumption of the LS-theory. All the same, a model that fits the data can be produced by LS provided the order is high enough.

The following command sequence together with the generated computer output shows the main ideas. Note the use of a macro to somewhat reduce the typing effort. The finally achieved model gives residuals whose standard deviation is less than 1 % in excess of the lambda value given by a second order ML model.

```
>STRUC STRF
     >NA MAX 8
     >NU MAX 1
     >NB MAX 8
     >KB MIN 1
     >KB MAX 4
     >KB ACT 4
     >X
>SQR R<WRK STRF
>LS LSMOD<STRF
```

```
LS LSMOD<STRF
79.09.14 - 16:45:39

RESULT OF PARAMETER REDUCTION(S):

DISCARDED PAR.        VLOSS          AIC

NONE                 85.2465       611.651
B 1( 1)              86.1615       609.786
B 1( 2)              86.4055       606.352
B 1( 3)              110.897       652.261
```

```
>STRUC <STRF
     >KB ACT 3
     >NB ACT 1
     >X
>MACRO ITER
     >DELET (T) LSMOD
     >LS LSMOD<STRF
     >END
>ITER
```

```
Identification
LS


LS LSMOD<STRF
79.09.14 - 16:47:00

RESULT OF PARAMETER REDUCTION(S):

DISCARDED PAR.        VLOSS          AIC

NONE                 85.2465       611.651
B 1( 1)              86.1615       609.786
B 1( 2)              86.4055       606.352
B 1(11)              86.4087       602.359
B 1(10)              86.4100       598.362
B 1( 9)              86.5139       594.602
B 1( 8)              86.5213       590.620
B 1( 7)              87.5702       589.030
B 1( 6)              88.2926       586.673
B 1( 5)              97.3882       602.282
B 1( 4)              122.814       644.676

>STRUC <STRF
      >NB ACT 3
      >NA ACT 1
      >X
>ITER


LS LSMOD<STRF
79.09.14 - 16:47:54

RESULT OF PARAMETER REDUCTION(S):

DISCARDED PAR.        VLOSS          AIC

NONE                 85.2465       611.651
B 1( 1)              86.1615       609.786
B 1( 2)              86.4055       606.352
B 1(11)              86.4087       602.359
B 1(10)              86.4100       598.362
B 1( 9)              86.5139       594.602
A  ( 8)              86.5934       590.786
B 1( 8)              86.5945       586.789
A  ( 7)              87.6784       585.277
B 1( 7)              88.2696       582.620
A  ( 6)              88.6399       579.458
B 1( 6)              89.6397       577.701
A  ( 5)              89.7082       573.854
A  ( 4)              98.7530       589.066
A  ( 3)              136.603       649.957
A  ( 2)              290.541       796.891
```

```
Identification
LS


>STRUC <STRF
      >NA ACT 4
      >X
>ITER


LS LSMOD<STRF
79.09.14 - 16:50:25

RESULT OF PARAMETER REDUCTION(S):

DISCARDED PAR.        VLOSS          AIC

NONE                 85.2465       611.651
B 1( 1)              86.1615       609.786
B 1( 2)              86.4055       606.352
B 1(11)              86.4087       602.359
B 1(10)              86.4100       598.362
B 1( 9)              86.5139       594.602
A  ( 8)              86.5934       590.786
B 1( 8)              86.5945       586.789
A  ( 7)              87.6784       585.277
B 1( 7)              88.2696       582.620
A  ( 6)              88.6399       579.458
B 1( 6)              89.6397       577.701
A  ( 5)              89.7082       573.854


>LIST (T) LSMOD


LIST (T) LSMOD
79.09.14 - 16:51:27

BEGIN

"LS LSMOD<STRF
"79.09.14 - 16:50:25
"
DISCRETE MISO TRANSFER FUNCTION

SAMPLE INTERVAL 1. S

LAMBDA 0.947144 +- 4.73572E-2

LOSS FUNCTION 89.7082

AIC 573.854

APOLYNOMIAL
1. Q^-0 - 0.727179 Q^-1 - 0.153094 Q^-2 + 6.68778E-2 Q^-3
 + 0.221625 Q^-4
```

```
Identification
LS


BPOLYNOMIAL 1
 Q^-3 * (1.0665 Q^-0 + 1.13069 Q^-1 + 0.877197 Q^-2)

CPOLYNOMIAL
1. Q^-0

END
```

ML

Purpose

To perform maximum likelihood identification of multiple input - single output systems on the form

$$A(q^{-1})y(t) = B_1(q^{-1})u_1(t) + \ldots + B_m(q^{-1})u_m(t) +$$

$$+ \lambda C(q^{-1})e(t).$$

Command

```
ML [(SW)] SYST[(NAME)] < DATA[(C1 .. )] NO [EXT]
```

SW	- switch telling whether subcommands are wanted or not SW = 'SC'/'VOID' (by default SW = 'VOID') SC - subcommands wanted VOID - subcommands not wanted
SYST	- system file name
NAME	- name of section within SYST
DATA	- name of input data file
C1 ..	- column number(s) (by default C1.. = 1, 2, ...)
NO	- model order
EXT	- those of the global variables V.EXT and AIC.EXT that exist as reals will receive the value of the loss function and Akaike's test quantity, respectively

Subcommands

```
INVAL 'ABC'/'C' SYST[(NAME)]
```
to fetch starting values for A-, B-, and C-parameters and initial values or just for C-parameters from the system file SYST, section NAME.
In the latter case the 1st iteration will be a least squares estimation of the A and B parameters if INVAL is not used, then all the parameters will have 0.0 for starting value.

```
FIX A (2) [VA2] (3) [VA3] B (21) [VB21] .....
```
to fix the parameters A2, A3 etc. to the values VA2, VA3 etc. If no values are given the parameters are fixed to zero or the values taken from a parameter file by the subcommand INVAL.

Identification
ML

SAVE [STDEV] [GRAD] [EVALS] [COMAT]
: to save standard deviation of the parameters, gradient of the parameter estimate, eigenvalues of the second derivative matrix and/or the covariance matrix

LOOK
: to display names of subcommands and reserved variables used by ML

KILL
: return immediately to the main program; no identification will be performed

X
: end the subcommand sequence and start the identification

Function

A maximum likelihood model of order NO is estimated from DATA(C1..) and written into the system file SYST. Note that the output of the system must be the last column specified or, if no column numbers are specified, the last column in DATA.

The convergence condition, the estimated parameters and usually their uncertainties and some other information is also output to the terminal. Cf. the example below.

Global variables used (default values underlined):

INIML. 1 initial values for the output will be estimated.
0 no estimation.
PRIML. 0 no printout.
1 loss function and lambda for starting values are printed as well as the final estimate with derivatives, second derivative matrix and inverse of the second derivative matrix. The final estimate is also displayed.
2 1 + the estimate printed for each iteration.
3 2 + derivatives, second derivative matrix and inverse of the second derivative matrix printed for each iteration. Each estimate is displayed.
LIML. 1 the residuals will be limited to $3 * \lambda$ in each iteration, i.e. if $\varepsilon(t) > 3\lambda$, then $\varepsilon(t) = 3\lambda$.
0 no limitation.
ITML. maximum number of iterations (default 20).

These variables may at any time be modified with a LET command.

Identification
ML

Method

A maximum likelihood estimate of the parameters in the model

$$A(q^{-1})y(t) = B_1(q^{-1})u_1(t) + \ldots + B_m(q^{-1})u_m(t) + \\ + \lambda C(q^{-1})e(t)$$

is obtained by minimizing the loss function

$$V(\theta) = \frac{1}{2}\Sigma\varepsilon^2(t)$$

where

$$C(q^{-1})\theta(t) = A(q^{-1})y(t) - B_1(q^{-1})u_1(t) - \ldots - \\ - B_m(q^{-1})u_m(t)$$

and

$$\theta = (a_1, \ldots, a_n, b_{11}, \ldots, b_{1n}, \ldots, b_{m1}, \ldots, b_{mn}, \\ c_1, \ldots, c_n).$$

The maximum likelihood estimate of λ will be

$$\lambda^2 = \frac{2}{N}V(\hat{\theta}),$$

where $\hat{\theta}$ is the minimum point of V. The minimization is performed iteratively by a combined Gauss-Newton and Newton-Raphson algorithm. The parameter accuracy is estimated by

$$\sigma^2_{\hat{\theta}_i} = \lambda^2[V''_{\theta\theta}(\hat{\theta})]^{-1}_{ii}.$$

Akaike's test quantity (n is the number of estimated parameters):

$$AIC = N * (\ln 2\pi * 2 \ln \lambda) + 2n$$

is expected to have a minimum when the number of parameters

is correct.

The convergence criteria are

$$\max_i \left| \frac{\Delta\theta_i}{\theta_i} \right| \leq 10^{-4} \quad \text{or} \quad |\Delta V| \leq 10^{-6}$$

The first step of the iteration is a least squares estimate of the parameters of $A(q^{-1})$ and $B_i(q^{-1})$ provided that the subcommand INVAL ABC is not used.

References

Astrom, K.J., T. Bohlin, and S. Wensmark: Automatic construction of linear stochastic dynamic models for stationary processes with random disturbances using operating records. Report TP 18.150 (1965), IBM Nordic Laboratory, Sweden.

Gustavsson, I.: Parametric identification of multiple input, single output linear dynamic systems. Report TFRT-3012 (1969), Dept of Automatic Control, Lund Institute of Technology, Lund, Sweden.

Cautions: Restrictions

Maximum order is 9, maximum number of input is 8 and maximum total number of estimated parameters is 25 including fixed parameters and initial values for the output. There is no maximum for the number of input data.

Hints

a) If just a least squares estimation is wanted, use no initial values for the A and B parameters and set PRIML. and ITML. to 1.

b) Time series analysis can be carried out if only one column is indicated since the number of inputs is always assumed to be one less than the number of columns specified and the output is always the last column specified.

c) A moving average model y = Ce can be estimated in the following way:
Fix all a_i, i=1,...,n, to zero.

Give any nonzero initial values to c_i, i=1,...,n.

d) Known relations between parameters can be introduced as indicated by the following examples:

Ex 1. Known relation: $b_1u_1(t-1) + 1.5\ b_1u_2(t-1)$.

Develop a signal: $u(t-1) = u_1(t-1) + 1.5\ u_2(t-1)$

and then estimate $b_1u(t-1)$.

Ex 2. Known relation: $a_1y(t-1) + 3a_1y(t-2)$.

Develop a signal: $y(t-1) = y(t-1) + 3y(t-2)$ and use this signal as a new input signal and estimate the coefficient a_1 as the corresponding b parameter.

e) Notice that the same technique as in d) can be used for e.g. the estimation of the coefficients a_{12} and a_{13} in the model

$$y(t) + a_1y(t-1) + a_{12}y(t-12) + a_{13}y(t-13) =$$
$$= b_1u(t-1) + e(t) + c_1e(t-1).$$

In this case develop a signal $u_2(t-1) = y(t-12)$ and use this signal as an additional input signal to estimate a_{12} and a_{13} as the parameters b_{21} and b_{22}.

f) If the experiment/simulation is not started in steady state with the levels subtracted from the data, it might be useful also to estimate initial output values of the difference equation

$$A(q^{-1})y(t) = B_1(q^{-1})u_1(t)+...+B_m(q^{-1})u_m(t) + \lambda C(q^{-1})e(t),$$

i.e. the values y(0), y(-1), ..., y(1-n), where n is the order of the system.

g) During the minimization, it can happen that the algorithm converges to a local minimum instead of a global one. To overcome this problem, start the algorithm with different initial values (subcommand INVAL). The system file used by INVAL may be created in one of two ways:
- Via the editor, maybe by altering some coefficients in

an existing system description.
- By fixing some coefficients to values different from the current (local) minimum, and then performing a new identification. The resultant system description is used as a new starting point, now with all parameters free.

h) There are several ways in which the results of an identification could or should be tested. Examples are found in the description of the commands RESID, DETER, and RANPA.

Example

Given is the recorded input (a PRBS signal) and output (Y) of a system as shown in Figure 1. In this case 200 samples were available although not all are shown in the figure.

To obtain a model through maximum-likelihood identification, the following series of commands were executed. (First the available data are moved into a single file WRK.)

```
>MOVE WRK < PRBS
>MOVE WRK(2) < Y(1)
>ML ML1 < WRK 1
>ML (SC) ML2 < WRK 2
    >SAVE COMAT
    >X
>LIST (T) ML2
```

The immediate result of the ML-command is output to the user terminal, shown in Figure 2 for the 2:nd order model, and into the output file. The system file ML2 is shown in Figure 3.

In this case the model ML2 was finally accepted (AIC increased to 838 for the 3:rd order model). The examples shown for the commands RESID, DETER, and RANPA illustrates some ways to test the models obtained in ML.

Identification
ML

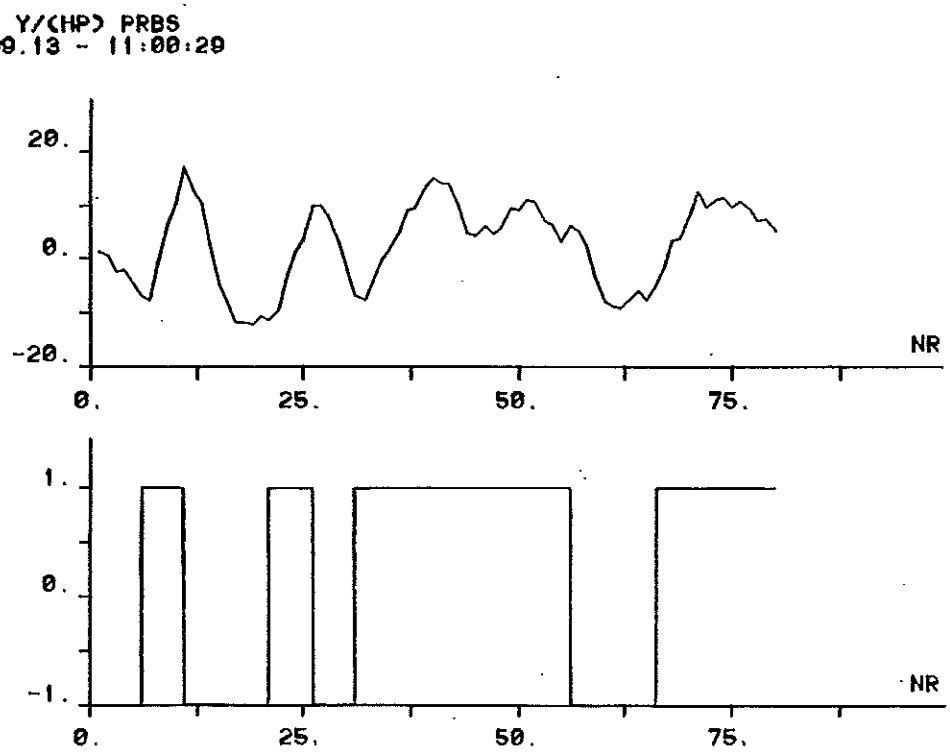


Figure 1. The first 80 values of the input signal (PRBS) and the output (Y) used in the example.

```
ML (SC) ML2<WRK 2
79.09.14 - 16:16:54

CONVERGENCE (DV/V< 1.0-006)
***************************

A1              -1.47060   +-   4.256836-002
A2               .677576   +-   3.278783-002
B1               .885117   +-    .194787
B2               .625777   +-    .272574
C1              -.789223   +-   8.064384-002
C2               .235413   +-   6.945371-002

LAMBDA           1.88372   +-   9.418594-002
LOSS FUNCTION    354.840
AIC              832.875
```

Figure 2. The output on the user's terminal when obtaining ML2.

```
Identification
ML



LIST (T) ML2
79.09.14 - 16:17:16

BEGIN

"ML (SC) ML2<WRK 2
"79.09.14 - 16:16:55
"
DISCRETE MISO TRANSFER FUNCTION

SAMPLE INTERVAL 1. S

LAMBDA 1.88372 +- 9.41859E-2

LOSS FUNCTION 354.84

AIC 832.875

APOLYNOMIAL
1. Q^-0 - 1.4706 Q^-1 + 0.677576 Q^-2

BPOLYNOMIAL
 Q^-1 * (0.885117 Q^-0 + 0.625777 Q^-1)

CPOLYNOMIAL
1. Q^-0 - 0.789223 Q^-1 + 0.235413 Q^-2

COVARIANCE MATRIX
 1.81207E-3 -1.32908E-3 -3.25037E-3  8.02317E-3  1.89845E-3
-1.32908E-3  1.07504E-3  1.62424E-3 -4.77565E-3 -1.41224E-3
-3.25037E-3  1.62424E-3   3.7942E-2 -4.70401E-2 -2.61057E-3
 8.02317E-3 -4.77565E-3 -4.70401E-2  7.42963E-2  7.70814E-3
 1.89845E-3 -1.41224E-3 -2.61057E-3  7.70814E-3  6.50343E-3
 1.61995E-4  6.57611E-5  -2.6613E-3  3.71847E-3 -2.36442E-3

 1.61995E-4
 6.57611E-5
 -2.6613E-3
 3.71847E-3
-2.36442E-3
 4.82382E-3
```

Figure 3. A listing of the file ML2.

Identification
SQR

SQR

Purpose

To compute the square-root matrix R of the least-squares identification algorithm. See the command LS.

Command

SQR RFIL < FNAME [(C1 C2 ..)] SFIL

RFIL - square root matrix file name
FNAME - input data file name
SFIL - structure file name

Function

The R-matrix is computed recursively from the data file. The output signal of the system is assumed to be the last column specified or, if no column numbers are given, the last column.

SQR will update the structure file SFIL with a notation of the name of the R-matrix and the number of data points used, as well as the name and column numbers of the data file. This serves two purposes:

a) This information is used by LS.

b) The existence of this information prevents STRUC from altering the maximum values, which are used by LS to properly interpret the R-matrix.

Hints

For a given set of maximum values, a single call to SQR is sufficient, still allowing repeated calls to STRUC, changing actual values, and to LS, computing parameter estimates.

Identification
STRUC

STRUC

Purpose

To give the structure of a model to be identified by the LS algorithm. Cf. the command LS.

Command

```
STRUC STRF
   or
STRUC [STRF1] < STRF2
```

```
STRF    - new structure file
SNAM1   - output structure file
SNAM2   - input structure file
```

Subcommands

In some commands below a switch exists, here designated sw. Its value is 'MAX'/'ACT', default value being 'ACT' (cf. subcommand KB). Its function is to indicate that a maximum or actual value is being specified for the variable in question.

REVERT
: This command will remove all information other than maximum and actual structure variables (cf. command SQR).

NA [sw] N
: NA, i.e. the degree of the A-polynomial is given the maximum/actual value N.

NU [sw] N
: NU, i.e. the number of inputs is given maximum/actual value N.

NB [sw] N1 N2 ... NNU
: NBi, i.e. the degrees of the B-polynomials are given maximum/actual values N1, N2, etc.

KB [sw] N1 N2 ... NNU
: KBi, i.e. the delay in each input are given maximum/actual values N1, N2 etc. Here a third alternative 'MIN' is allowed for sw. The default is 1, 0 is allowed, negative values are illegal. (In such a case, use command SLIDE.)

FIX A(N) [VN] ... B(M) [VM] ...
: The parameters a_n ... b_m etc. are given fixed values v_n, v_m etc. If no value is given, zero is assumed.

Identification
STRUC

UNFIX [A(N1 N2 ...)] [B(M1 M2 ...)]
The specified parameter(s) (default all) is unfixed.

KILL - resume main command mode, produce no results
X - output results, then resume main command mode

Function

In the first form of the command, a new structure file is created. In the second form the file STRF2 is updated giving a new file STRF1 or a new version of STRF2 if STRF1 is missing.

STRUC assumes a model on polynomial form

$$A(q^{-1})y(t) = \sum_{i=1}^{NU} q^{-k_i} B_i(q^{-1})u_i(t) + e(t).$$

The structural quantities NA, NU, NB(i), and K(i) can be given both maximum and actual values.

STRUC will receive information through subcommands and will perform various checks for consistency, eg.:

a) New maximum values are legal only for a new structure file or after the REVERT subcommand.

b) New actual values must be less or equal to maximum values.

c) The number of inputs (NU) must be specified prior to the commands NB or KB.

Miscellaneous
DELET

DELET

Purpose

Deletes files from the data base.

Command

```
DELET [(DMODE1)]FNAM1 [ [[(DMODE2)]FNAM2] ... ]

DMODE    - data mode indicator = 'D'/'T'/'A'
           D      - FNAM is assumed to contain binary data
           T      - FNAM is assumed to contain text
           A      - FNAM is assumed to be an aggregate file
(DMODE='D' by default)
FNAM     - file name
```

Function

If the specified file exists in the data base it is deleted, otherwise an error message is given.

Hints

The existence of the file may be tested by the command FTEST.

Example

Cf. the macro in the example in RANPA.

FHEAD

Purpose

To display file head parameters of data files and enable the user to change them.

Command

```
FHEAD [AGGREG:]FILE

AGGREG  - aggregate file name
FILE    - file name
```

Subcommands

```
INDEX VALUE
          set the INDEX:th parameter to VALUE
LOOK[K] - display the K:th (default all) file head
          parameter(s)
KILL    - exit from FHEAD without updating the file head
X       - exit from FHEAD and update the file head

Note    - an immediate, READ-ONLY form also exists:
          >FHEAD [AGGREG:]FILE 'LOOK' [K]  equivalent to:
          >FHEAD [AGGREG:]FILE
               >LOOK [K]
               >KILL
```

Function

The command has two different forms. One takes subcommands and also allows changes to be made. The other form is a read only form that does not enter subcommand mode. The output on the terminal is shown in the example below.

Cautions, Restrictions

It is not possible to change parameter 7. Attempts to alter parameters 1, 2, 3, and 10 will produce a warning message and should generally be avoided by users not very familiar with the internal orginazation of data files.

Hints

Reorganization of data files is often best done by special purpose commands like CUT, CONC, PICK, MOVE etc. In extreme cases the sequence FORMAT - (EDIT) - CONV could be used.

Miscellaneous
FHEAD

Examples

Two forms of the command are shown below with the accompanying output.

```
>FHEAD A
     >LOOK

FHEAD A

 1. NUMBER OF ROWS              2
 2. NUMBER OF COLUMNS           2
 3. THIRD DIMENSION             1
 4. SAMPLE INTERVAL             3 TICKS = 3.00000     S
 5. DATE RECORDER               0 (YY.MM.DD)
 6. TIME RECORDED               0 (HH:MM:SS)
 7. CONSTANT RECORD LENGTH 0    1 (1 MEANS YES)
 8. GENERATED BY COMMAND NR     2
 9. FILE TYPE                   3 = MATRIX
10. SKIP COUNT                  0

     >KILL

>FHEAD A LOOK 4

 4. SAMPLE INTERVAL             3 TICKS = 3.00000     S
```

FTEST

Purpose

To test the existence of files.

Command

```
FTEST [(DMODE)] FNAME

DMODE   - data mode indicator = 'A'/'D'/'T'
          A     - aggregate file
          D     - binary data file
          T     - text file
          (by default DMODE = 'D')
FNAME   - file name
FTEST.  - reserved variable returned = 0/1
          0     - the file does not exist
          1     - the file exists
```

Function

The command tries to access a file with the given name and type. If it was possible, the reserved global variable FTEST. is given the value 1, otherwise it is given the value 0.

TURN

Purpose

To alter the value of some program switches.

Command

TURN SWITCH STATE

SWITCH - switch name = 'TEXT'/'TIME'/'GRAPH'/'DK'
- TEXT - enables/disables all text output on the display (default: TEXT is enabled)
- TIME - if disabled, data will be plotted versus sample number, else versus time units (default: TIME is disabled)
- GRAPH - enables/disables all graphics output (default: GRAPH is enabled)
- DK - enables/disables command logging into text files (default: DK is enabled)

STATE - switch state = 'ON'/'OFF', if switch = 'TIME', 'H'/'M'/'S' is used instead of 'ON'
- ON - the switch is enabled
- H - plotting will be versus time in hours
- M - plotting will be versus time in minutes
- S - plotting will be versus time in seconds
- OFF - the switch is disabled

Function

The switches are set according to the value in the argument list.

Cautions, Restrictions

Note that the operation of the program also is controlled through the more flexible method of global variables, cf. the general guide.

Alphabetical Command List