Interactive management of time series

Bianchi, Carlo and Calzolari, Giorgio and Corsi, Paoli

Ibm Scientific Center, Pisa, Italy

November 1974

INTERACTIVE MANAGEMENT OF TIME SERIES

C. Bianchi, G. Calzolari and P. Corsi

A very important element in order to study the structure of a national economy, is the organization of a data base consisting of the observations, over time, on the variables which characterize the economy.

The main purpose of this data base is to offer the researcher a supply of information which can be analyzed, revised and updated in a repetitive way. This can be very well accomplished by an interactive approach which, by allowing a direct dialogue between the user and the computer, facilitates this iterative procedure.

A developed interactive user oriented language makes it possible for a large number of users to have simultaneous access to the same information, without interfering with one another and to use this information, in order to study any kind of characteristics of the economy under control.

The central file holds various series of data which are handled globally by one single program, called IMTS (Interactive Management of Time Series).

By utilizing this program, it is possible to use data both from the central file (official data) and from a private single user file (confidential data).

Summarizing, the research capabilities supplied by the IMTS package are analysis, transformation and regression of stored data. These features can be used independently or in any combination by the IMTS language, which provides a single command for each application: it is, therefore, possible to write regression equations as well as algebraic expressions defining transformations of the economic variables.

In the IMTS environment, not only the IMTS commands but also all the CMS commands are accepted and executed. This is performed by a suitable arrangement of the virtual main storage.

The storage may be divided into four zones:

1) Core of the CMS operating system, free area for system functions and the transient area (0-12000 hex.).

2) Low-address free area, designed to hold the work matrix, or to load the system's or user's module format programs. (EDIT, COMBINE, SORT, NEW, CREARCH, APLIUP, etc.) (12000-36000 hex.).

3) Protected area, containing the program routines permanently held in the storage (static routines), such as the coding and decoding routines, I/O routines, etc. (36000-50000 hex. approx.).

4) High-address free area, intended for the dynamic loading of plotter or regression routines (50000-end of virtual memory).

Before interpreting a command, the program operates on the NUCON table of the CMS, changing 5 addresses as follows

LOWEXT (lower end of the high-storage addresses occupied by the operating system via FREE and EXTEND routines): the value is removed and replaced by 34000 (hex.); whatever module may be executed (of the system or of a user), or TEXT module loaded, no storage area may be allocated (e.g.,: used) above the specified address. LOCCNT (location counter) and HIMAIN (upper end of the low-storage addresses allocated by the GETMAIN routine of the operating system): the content is removed and replaced by 12000 (hex.) in both cases. LSTADR (address of the last page of the virtual machine memory): and LDRTBL (last address in the loader tables, practically speaking the same as the last address in the virtual memory): the content is removed and replaced by 35000 and 36000 (hex), respectively.

Immediately afterwards, the program checks whether or not the command to be executed is a CMS command. If so, the CMS takes over and executes the command, and then returns to the 'calling' program, which resets the addresses changed in the NUCON table. If not, the addresses changed in the NUCON are immediately reset and control is passed over to the coding and decoding routines (permanently housed in the storage). These routines use the free area (12000-36000) organized as the work matrix. If the command given calls for the use of a regression or plotter routine not resident in the storage, the requested routine is dynamically loaded by the CMS loader into the high-address free area and, among other parameters, the address of the work matrix is passed to it.

Any CMS command may be executed. However, it is not possible to execute chained commands relating to a common storage area (for example, LOAD and START for a program, though LOAD (XEQ) is possible, and so is $).

In particular, the following CMS commands are frequently needed: EDIT - ALTER - COMBINE - LISTF - STAT - PRINT and OFFLINE PRINT - SORT - APLIUP to which are also added: NEW - CREARCH which are the names of utility programs (contained in module format in the virtual machine which contains the time-series file) loaded and executed by the CMS, in the same way as the modules which may be handled by the system.

From a practical point of view, as concerns the IMTS commands, they are issued in the sphere of the IMTS language.

Issuing of commands and printing of results are carried out by the user through the terminal. The user must issue at the terminal the command to be executed, eventually followed by the names of the time series (from the central file or private series; in the event of the same names, the private series is the first to be searched for), numerical constants, operator codes and function names.

Generally speaking, a command must be contained in one line, nevertheless the continuation on one or more subsequent lines is allowed, by using the special continuation sign (=) at the end of each line, except the last. The program decodes each line as soon as it is written, but it does not begin execution of the command until it comes to a line not ending with the continuation sign.

The following operations may be carried out on time series and on any constants:

1) Operations and functions executed by routines held in the storage (static routines):

Logical operators:
.AND., .OR., .NOT., .LT., .LE., .EQ., .NE., .GE., .GT.

Arithmetic operators:
+(Addition), -(Subtraction), x(Multiplication), /(Division), **(Raise to power), LOG (Natural logarithm), LOG10(Decimal logarithm), EXP (Exponential (base e)), SIN (Sine (Radians)), COS (Cosine (Radians)), ATAN (Arc tangent (Radians)), SINH (Hyperbolic sine), COSH (Hyperbolic cosine), ABS (Absolute value), MAX (Maximum value), MIN (Minimum value), MEAN (Mean value), VAR (Variance), INT (Integer part), RV (Rate of variation), LAG(nn) (Data lagging), ONE (nn/mm) (Selection of one single item of data from a series), SEL (nn/mm, ll/kk) (Selection of part of a series, CPR(nn) (Compression of a series), IF (For discontinuities and decision rules).

2) Operations and functions carried out by dynamic loading routines:

PLOT (inline plotter), OLS,OLSR (ordinary least squares estimate), TSLS (two-stage least squares estimate), LISE (limited information single equation estimate), DMS (connection to DMS/2).

During the operations for the coding and decoding of commands, some types of formal errors are likely to occur in the program as, for example, number of open parenthesis different from number of closed parenthesis, quotes openend, but not closed, to contain names of series, and so on.

After indicating on the terminal the type of error and the name of the routine containing it (indication not normally essential for the user, but very useful for the system engineer in setting up new functions), the program automatically makes them applicable to the EDIT requests of

CMS to the file in which the wrong command is held. Issuing the usual EDIT requests, the user can correct the wrong command and, by returning the control to the main program, can execute the correct command.

The file, the data processing program and the utility programs are contained in one single CMS minidisk. Only one virtual machine has read/write access to this disk. This means that the file may be updated only by this same virtual machine.

All the other authorized virtual machines have READ-ONLY access to this disk by the following CMS command: LOGIN 193 A, P; which can be given automatically by an EXEC procedure.

The work files for each virtual machine are created in the work disk of the respective machine, and may therefore be manipulated by the single user.

The data processing program is in MODULE format; it may therefore be called merely by writing at the terminal the following command: IMTS. As soon as the program takes over control an arrow is printed at the terminal and the keyboard is released, pending indication of the operations to be carried out. On completion of processing, the program restarts from the beginning, repeating the arrow. This kind of answer allows the user to realize the environment (IMTS) in which he is operating at the moment.

The IMTS command must be issued in the CMS environment; to close it and reenter the CMS, it is sufficient to write FINE in columns 1-4.

An example of the automatic error recovery procedure, followed by the standard output for the OLS estimation is hereunder presented.

```
--->
ols(log'mm',log'dl',log'k'
(D2)ERROR : PARENTHESIS OR THEN-ELSE DO NOT BALANCE
(AMDM1) ERROR RECOVERY PROCEDURE
USE EDIT COMMANDS.
DEFAULT TABS SET.
EDIT:
p
OLS(LOG'MM',LOG'DI',LOG'K'
c /'    /')/
OLS(LOG'MM',LOG'DI',LOG'K')
file

--->
go

OLS(LOG'MM',LOG'DI',LOG'K')

THERE ARE  19 DATA FOR EACH SERIES, FROM 1952/ 1 TO 1970/ 1 :  1 EACH YEAR.
```

```
EXECUTION BEGINS...
1 SIMBOLIC UNIT (6/8)
2-9 NAME
10 RESIDUALS (0/1)
-........-
6import  0
MULTIPLE REGRESSION.....IMPORT
```

VARIABLE NO.	CORRELATION X VS Y	REGRESSION COEFFICIENT	STD. ERROR OF REG.COEF.	COMPUTED T VALUE
2	0.99567	2.12933	0.04094	52.01243
3	0.18790	1.19420	0.41229	2.89648

DEPENDENT
1

INTERCEPT	-19.07669
MULTIPLE CORRELATION	0.99716
R**2	0.99433
STD. ERROR OF ESTIMATE	0.05089
STD. DETERMINANT	0.98193

A N O V A

SOURCE	D.F.	SUM OF SQUARES	MEAN SQUARES	F VALUE
REGRESSION	2	7.2646	3.6323	1402.44488
RESIDUAL	16	0.0414	0.0026	
TOTAL	18	7.3061		

DURBIN WATSON STATISTIC = 1.3079