

**V55 - A PROGRAM FOR INTERACTIVE VISUALIZATION
OF SPECTROGRAM SCAN RECORDS**

Bellas Y.

**Department of Astronomy and
National Astronomical Observatory,
Bulgarian Academy of Sciences,
72 Lenin blvd., 1784 Sofia, Bulgaria**

Abstract: A program "V55", part of the general S-software for digital processing of stellar coude-spectrograms in the Rozhen National Astronomical Observatory (NAO) is presented. The V55 provides an interactive visual-graphic control of spectrogram scan records and their processing in the NAO. The program assures tools to set-up the graphic environment (display, graph-images, cursors), to start/end particular image's activity, to move the currently active element, to process particular positions (save, visualize, delete), and to perform miscellaneous help functions. Each function is activated by a particular command the user inserts in an interactive dialogue, or the V55 gets from user-written command-files in another, indirect-command input mode. The program is created on the DEC PDP 11/34 minicomputer of the NAO, under the RSX-11/M system's FORTRAN IV PLUS language, and controls a DEC's VT-55 graphic-terminal.

Keywords: software, graphic-visualization, spectrogram

1. INTRODUCTION

A specialized software, referred to as "S", is elaborated in the Rozhen National Astronomical Observatory (NAO) for digital processing of stellar coude-spectrograms, starting with spectrograms scanning/recording, and ending with their digital restoring to quantities of astrophysical interest. In order to visualize the recorded scans, to control some of the stages of the processing (filtering, continuum fitting), and to help a further processing of the restored spectrum, a specialized program should be created. Such a program should provide tools to display a graphic-image of data parsed by a file recorded in a particular format, to scale the image as well as to "move" it. A flexible cursor should be available to fix desired points on the image (to save their positions and to transfer them from/to the program (say, by a special file). It would be valuable to have the possibility to type text-strings, as well.

For this purpose, a program "V55" is created, as part of the S-software, on the PDP 11/34 minicomputer and its VT-55 graphic terminal. This paper presents the V55 program, as follow: the V55's tools (frames, graph-elements, files) are described in the second paragraph; the functions of the V55 are given in the third; its realization is presented in the fourth paragraph; the fifth gives a general information on its use.

The V55 is fully described in a documentation including an user's manual "V55-MANUAL" and a listing of the program's modules "V55-CODE". Both parts are available as text-files on the PDP as well as on an IBM-PC or compatible computer.

2. V55'S TOOLS

There are introduced few basic terms for the "tools" by which the V55 program acts, namely: the screen-frames, the graphic-elements, and the working files.

2.1 Screen-frames

The entire VT-55 display's screen (24 rows of 80 columns) is "arranged" by the program in three, partially overlapped parts, referred to as fixed-, text- and graphic-frame. (fig. 1)

The fixed-frame is the outermost part of the screen. It includes the three top rows, the leftmost column, and fixed part of the bottom row. This is the V55 reserved place (not accessible by the user, excluding a command field) where it outputs particular information. The top-rows are arranged in four vertical parts, from left to right: the dialogue-part, the V55-title, the parts for the "passive" and the "active" graph-element. There are three fields in the dialogue-part: the command input field in the top where the user types the desired command-string; the one to verify the last command execution, in the middle; a message-field in the bottom. The graph-element parts (passive/active) includes fields for the element's identifier and for its current position. There are the y-origin, y-end, x-origin, and x-end fields in the left column and the bottom row of the fixed-frame, for displaying coordinates of the currently selected image's graphic-frame.

The text-frame is the internal part of the screen, not occupied by the fixed, where the user can place (accessible by row- and column-number) any textual information say, the

graph's identifier, the axes labels, as well as to clear it, without affecting the graph-elements. The text-frame is also the place where the V55 displays the desired runtime help information.

The graphic-frame is the one where the graph-elements are really displayed. Physically, it coincides with the textual one excluding the bottom row and the two rightmost columns, but it is accessible, rather, in terms of "graphic-points" (integers from 0 to 500 for the x-axis and from 0 to 200 for the y; the XOYO is the bottom, leftmost graphic-point). The frame is marked by margins - vertical and horizontal lines; no point is accessible out of these margins.

2.2 Graphic-elements

Three graphic-elements are usable working under the V55 program, namely: graph-image, cross-cursor, and mark-cursor. The image visualizes, by a sequence of graphic-points, the whole or a part of an input data-sequence parsed to the V55 by a working-file.

The graph can be "enabled" to be imaged as "normal" (by graph-points), it can be "shaded" - filled up between the bottom margin and the points, or it can be with "reversed" abscissa (x-values decrease when graph-points increase; it is always "shaded"). On the other hand, the graphic-points can "image" exactly the corresponding record-data (normal regime), but also, if the input record is imaged reduced, there can be presented the average-values (mean regime) or the min/max range (extremum regime) of the corresponding records.

The cross-cursor is an optional graphic-element (formed by the intersection of a vertical and a horizontal line); it can be put in any internal point of the graphic-frame. The mark-cursor is another optional element imaged like a short vertical bar. The mark can be used like the cross, but it is "connected" vertically to the corresponding graph, always fixing a graph-point on the very image.

Up to two graph-images can be displayed simultaneously, but only one can be active. The active graph's enable-mode, data-regime, input-data file name, and scale (magnification) parameters can be updated; the graph-image can be displayed, shifted, erased (cleared from the screen).

Each displayed graph can use its own cursor, so there can be imaged up to two crosses and two marks, simultaneously. The cursor (either cross or mark) of the selected graph can be activated. When it is active, the cursor can be shifted, its step can be updated, the position-data processing can be carried out; the cursor can be erased (if not active) and/or it can be deactivated (the graph-image is reactivated).

The V55 saves continuously the position of each displayed element (coordinates of the images origin points XOYO, the center coordinates for the cross-cursors, and the abscissa and the corresponding image's ordinate for the mark-cursor); these for the currently selected graph are displayed in the particular fields of the fixed-frame. The position on the graphic-frame can be described in any of the three terms: graphic-point - the pixel's X- and Y-position on the frame; record-number - the corresponding record's number and value; object-values - the real value that the record represents.

The V55 also saves and correspondingly updates particular

parameters necessary to transform between these terms. Only the real object-values are displayed on the fixed-frame.

2.3 Working-files

The user's data for a graph-image are parsed to the V55 by, so-called, image-data files. Three formats are allowed for such one file, namely: "DD - SPC", "PLOT", and "DATA". Nevertheless the format, the file must be a direct-type one (accessible by the record number) with unformatted (binary) integer records of one-word length (two bytes). As many as 65535 data are allowed to be recorded in the file. The only difference between the formats consists in the type of an initial "header" in the file.

The "DD -SPC" is a release of the FITS format created in the NAO (Bellas, 1988) for handling coude-spectrogram scan records on disk-files under the RSX operating system of the DEC PDP 11/34 minicomputer. Such file contains a header and the records of the particular spectrogram-band scanned. The header (1440 records) is considered as ASCII-text containing particular information about the spectrogram-band, the very scanning, and the processing. There is a format-code in the header that the V55 checks, as well as particular records contain parameters about the records y-scale, the x-step and origin that the V55 gets and uses to "transform" the records to real object-values and vice-versa. Such files are created by the S-software's FRM reformatting program (Bellas, 1988) from tape-files originally recorded by the software's SCN package for spectrograms scanning (Bellas, Borisova, 1988), or are written by the filtering and restoring programs.

The "PLOT" format's header (40 two-bytes records) can be considered as eight rows of 10 characters ASCII-text. The first row contains the format's code, and the next two - the object's identifier. Follows four rows with the origin and step for the x-axis, and the y-origin and scale. The very last is the number of the recorded data. Such format files are created by the filtering and continuum-fitting programs for visual control (to be published).

A "DATA" format file contains no header, only records, so there are accepted, by default, 0.0 values for the origin of the axes and 1.0 for the step and scale. When an image-data file is "incorporated" to the V55, the latter checks if it is a "DD - SPC" format file; if this fails, the check up is made for the "PLOT" format and if this is not the case, the file is considered as "DATA" one.

On the other hand, there can be parsed "position-data" to and from the V55 fixing a sequence of cursor's positions. An ASCII-formatted file can be used containing records of the x and y coordinates (object-values) for each position. As many as 250 positions can be recorded in the file. By such files, created by the corresponding programs, the user can update visually the digital-filter or the continuum-fitting.

3. V55 FUNCTIONS

The V55 supplies up to sixteen general functions divided in four types, namely: the general-purpose functions, the shifting ones, these for position-data optional processing, and the miscellaneous functions. A particular, upper-case character or a special symbol (command's code; see further)

is used to identify the particular function.

3.1 General-purpose functions

These are intended to fix setups for the graphic-elements and to initialize the desired element. The graph activating and its mode/regime setting are carried out by the so-called G-functions. Such one function activates the desired cursor for the graph currently selected. This function displays the cursor, unlikely the graph-image activating (any shifting function must be used if not yet displayed on the screen). By particular S-function the scale for the graph imaging or the step for the cursors shifting can be updated (depending on the active element).

The working-file name for the currently selected graph should be set before its imaging; the named input file is incorporated, a check up of its usability (format) is done, particular parameters (identifier, y-scale, x-step, records number and min/max) are determined. The file can be changed whenever the graph is active. Similarly, the name for the position-data file can be set (a cursor should be active) in order to use the positions optional handling functions. These settings are carried out by F-functions.

On the other hand, the user can erase the cursor from the screen ending its activity ("end") as well as the cursor can be leaved, but passive ("escape"); in any case, the selected graph-image becomes active. Now, the very graph-image can be deactivated and erased. Finally, the user can complete and stop the V55 program's run ("exit"). All these are usable by particular E-functions.

3.2 Shifting (move and display)

The activated graph-image would be, initially, displayed only with such one function; the original records will be scaled as fixed by the user; the graphic-frame's origin will correspond to set by the function's x/y-origin value. Then, the shifting can be used to "move" the image updating its frame's origin. But, when a cursor is active such a function simply shifts the latter to the new graph-point closest to the value fixed by the function; the cross is "centered" in this point, whereas the marker is only "x-shifted" by the function, following by ordinate the graph-image.

There are X- and Y-functions to shift in an absolute mode fixing the new position by its coordinates, and functions to shift along the fixed direction (D-function - shifting down, L - left, R - right, U - up), by a fixed amount, relatively the current position or by a fixed step.

Any two functions can be combined, if not opposite in the direction. Additionally, there is available a M-function to repeat the last shifting. If the latter was "relative", the direction can be optionally changed to opposite.

3.3 Positions optional processing

These, so called P-functions allow to accept (and save) active cursor's positions, to change or delete the already saved, to rearrange the sequence of the saved positions by the x-value, to verify them, as well as to "undo" the last function (accept, change, delete, sort). All these functions are performed on the basis of a working-file incorporated in

advance for position-data processing (see 2.3).

3.4 Miscellaneous functions

Of these, the T-functions handles the text-frame (2.1) processing that includes typing a text-string on a fixed place (by its row and column numbers), either vertically or horizontally, and similarly erasing some part or the whole text-frame. The object's identifier of the currently active graph can be displayed by such a function, as well.

A run-time help is provided by the H-functions. The whole help-information or its particular parts can be displayed interactively on the screen , waiting the user's action if more than one "screen" is to be displayed (the text-frame's original contents is restored after the help completes). By a H-function the full help information can be written in a special text-file as well, that can be printed out. These functions use the special file the V55 creates whenever is run and the file doesn't exist.

Additionally, a runtime-status information for the V55 is provided by so-called ?-functions. It is temporarily shown in the verify- and message-field of the fixed-frame (2.1). The functions allow to update the displaying duration, and to choose the desired information to be shown (the status of the graph-elements, the incorporated working-files, the last shift-command used, the image-scales, the cursors steps).

Finally, the @-functions are helpful to work with the V55 in the so-called indirect-commands-input regime (see 5.3). These are intended to switch-on the regime, optionally to fix the special file from where the indirect-commands will

be taken and/or to point the command to be started with. The status of the regime (file's name and pointer) can be shown, but also the currently pointed command or the whole file can be listed by particular @-function.

4. V55 PROGRAM REALIZATION

The V55 program is created on the PDP 11/34 minicomputer under the FORTRAN-IV-PLUS language ("PDP-11 FORTRAN", 1979). It consists of a program-dispatcher and a set of subroutines it calls to execute the particular function, as fixed by an internal flag. The elementary graphic functions are executed by calling the system's module PLOT55 ("VT55", 1977) for the VT55-display control primitives that is included, too. The general scheme of the particular modules interconnection is illustrated on the Fig.2. All the source-code modules are included in a general file V55.FTN (the "V55-CODE" document contains its full printing); the latter is translated by the F4P compiler ("FORTRAN IV PLUS", 1975). Then, the system's LBR utility ("RSX-11M", Utilities, 1979) is used to create, from the obtained object-modules file V55.OBJ, a library of the modules, the V55.OLB file, necessary to build the task.

The executable module V55.TSK is then created by the TKB ("RSX-11M/M-PLUS", Task Builder, 1979), using an overlaying as fixed in a special file V55.ODL; the resulting task-image file is about 100 Kw, but the memory-resident task occupies, really, no more than 14 Kw at a time. A special command-file V55.CMD is written that assures the V55 creating (merging the modules to create the source-code file, compiling it, creating the library, and building the task).

5. V55 PROGRAM USING

The particular function is activated by a fixed command the user inserts during an interactive dialogue or the V55 reads in a special, user-prepared (in advance) command-file in an indirect-input mode. Regardless the input-mode, the program prompts it is ready for a command, accepts it, and verifies it when successfully completed or outputs a fixed error-message if something fails.

5.1 Command-string

The command-string consists of three parts: code, switch, and parameters, included in this sequence and not separated by any character. The command-code is an upper-case letter, the "@" or "?" symbol. It is obligatorily the first element in the command. The code fixes its general function (see in the table below).

The command-switch is an additional element, obligatory for some of the codes. It consists of one or two upper-case letters. The command-switch fixes the particular function; the usable ones depend on the particular code.

The command-parameter is an optional element, that some commands use. It can be either a number-parameter, a textual one (enclosed by opening/ending apostrophes), a symbol, or their combination. The allowed command-parameters depend on the fixed command in each particular case. These are used to "parse" the desired, user-fixed parameters to the function, or to switch on its optional execution mode. If an allowed is not included, a defaulted parameter is set by the V55.

Follow a table of available commands (code and optional switches) giving the type (General, Miscellaneous, Optional, Shifting) and the general function for each allowed code.

CODE	COMMANDS	TYPE	GENERAL FUNCTION
@	@CO, @IN, @LI	-M-	indirect-commands processing
?	?D, ?F, ?G, ?M, ?S	-M-	actual-status information
D	D	-S-	down (relative) moving
E	END, ESC, EXI	-G-	ending functions
F	FIL, FPD	-G-	file-data incorporating
G	GA, GE, GR	-G-	graphic elements set-ups
H	HEL	-M-	help-information outputting
L	L	-S-	left (relative) moving
M	M	-S-	moving-command repeat
P	PA, PD, PR, PU, PV	-O-	position data-file processing
R	R	-S-	right (relative) moving
S	SC, ST	-G-	scale/step setting
T	TER, TYP	-M-	text-string processing
U	U	-S-	up (relative) moving
X	X	-S-	x-axis (absolute) moving
Y	Y	-S-	y-axis (absolute) moving

5.2 Interactive-dialogue regime

The "interactive-dialogue" is the general regime, that is automatically switched on whenever the V55 task is started. Initially, the V55-frames (see 2.1) are displayed on the VT-55 terminal's screen, the user's directory is looked for the test graph-images and the runtime-help files that are created, if do not exist (warning messages are displayed).

Then, it messages it is ready for input, the user inputs a particular command that is accepted and executed; the ready prompt is displayed again, and so on, until the V55 exiting is desired.

More precisely, an analysis of the command's code and of the string's general correctness (internal spaces, unpaired apostrophes) is carried out; if this fails, a particular error-message is displayed followed by the ready prompt. If the check up succeeded, the particular module is activated (a flag is set for this, depending on the command's code).

There is performed another checking in the called module for the optional switch and parameters and for the program's status (an internal status-word is being continuously saved fixing the graph-elements and files status). If this fails, the control is returned for a fixed error-message and for another command (like above); else the fixed function is activated. When the execution is successfully completed, a verifying message (the executed command-string) and the ready-prompt are displayed. But if some fault occurred, the corresponding error-message, like above, is output.

There is only one way to end this interactive-dialogue cycle and to stop the program - by the "EXI" command or by pressing the "CTRL" and "Z" keys simultaneously. The latter clears and releases immediately the VT-55, no matter the V55 program-status.

5.3 Indirect-commands-input regime

This special regime permits an automatic execution of a fixed sequence of commands. The V55 reads the commands in an

user-created (in advance) text-file. The latter must contain "command-rows" (a command per row), sequentially ordered as to be executed. They would be arranged in "command-blocks", delimited by special break-rows (the "@" symbol in the first column).

The indirect regime is activated by the "@CO" command, by the parameters of which, the user can fix the desired file's name and/or to point the command-row from where to start the run. Now, the program runs as in the dialogue regime, but after the ready prompt, it automatically reads the pointed command-row and increases the pointer, executes the command, reads the next, and so on. This cycle can be interrupted in one of the three cases: if an error occurs (command-string, status, execution); if a breaking-row is encountered (it is simply displayed); if the command-file's end is reached. In any case, the interactive regime is automatically reset and the ready prompt is displayed.

Two additional @-commands are usable (see 5.1) to obtain an information about the current command-file and pointer as well as to list the current command-row or block. Of course, the @-commands are available only in the dialogue regime.

5.4 Program's output information

Some functions output particular inform-messages (see in the 3.4) that are temporarily displayed in the verify- and in the message-field of the fixed-frame. On the other hand, the activating/ending (3.1) and the shifting (3.2) functions updates the corresponding graph-elements (passive, active, selected graph-image) coordinate-fields in the frame.

In case of an error (either in the command's format, in the status, during the execution) a fixed error-message is displayed, instead of the command-verifying one. The error is fixed by a negative value in an execution-flag, that the program correspondingly sets. The general reason for the error is fixed in the message; more detailed explanation can be obtained using the help-functions (see 3.4).

Using the VT-55's hard-copy unit, the user can get a copy of the whole screen, whenever this is desired (the Fig.1 is a real example taken by this way).

6. CONCLUSIONS

The V55 program assures tools to visualize the recorded in the NAO spectrograms, to control their digital processing by the S-software's programs, as well as to save archives of the scanned/processed spectra.

On the other hand, the indirect regime allows to perform a standard processing with minimal user-control. The runtime help provides enough information to exploit the program by an experienced user, whereas the detailed documentation is intended for the beginners. The latter contains also proper information, necessary to study and update the V55.

The applied modular structure of the program allows to update the commands, to include new functions, or to create similar routines for interactive, visual control.

Acknowledgments: The author would like to thank Y. Borisova for her remarks on the program and for her kindly assistance during the tests.

REFERENCES

Bellas Y., Astrophysical Investigations, 1988 (in press)

Bellas Y., Borisova Y., Astrophysical Investigations, 1988
(in press)

"FORTRAN IV PLUS", User's Guide,
DIGITAL EQUIPMENT CORPORATION, Maynard, Massachusetts,
(Dec. 1975)

"PDP-11 FORTRAN", Language Reference Manual,
DIGITAL EQUIPMENT CORPORATION, Maynard, Massachusetts,
(Dec. 1979)

"VT55", Programming Manual,
DIGITAL EQUIPMENT CORPORATION, Maynard, Massachusetts,
(Feb. 1977)

"RSX-11M", Utilities, chapter 14,
DIGITAL EQUIPMENT CORPORATION, Maynard, Massachusetts,
(Jun. 1979)

"RSX-11M/M-PLUS", Task Builder,
DIGITAL EQUIPMENT CORPORATION, Maynard, Massachusetts,
(Jun. 1979)

