"OBSERVATION OBSCURER" - TIME SERIES VIEWER, EDITOR AND PROCESSOR

I.L. Andronov

Department of Astronomy, Odessa National University,
T.G.Shevchenko Park, Odessa 65014 Ukraine, *il-a@mail.od.ua*

ABSTRACT. The program is described, which contains a set of subroutines suitable for fast viewing and interactive filtering and processing of regularly and irregularly spaced time series. Being a 32-bit DOS application, it may be used as a default fast viewer/editor of time series in any computer shell ("commander") or in Windows. It allows to view the data in the "time" or "phase" mode, to remove ("obscure") or filter outstanding bad points; to make scale transformations and smoothing using few methods (e.g. mean with phase binning, determination of the statistically optimal number of phase bins; "running parabola" (Andronov, 1997, As.Ap. Suppl. 125, 207) fit and to make time series analysis using some methods, e.g. correlation, autocorrelation and histogram analysis; determination of extrema etc. Some features have been developed specially for variable star observers, e.g. the barycentric correction, the creation and fast analysis of "O-C" diagrams etc. The manual for "hot keys" is presented. The computer code was compiled with a 32-bit Free Pascal (www.freepascal.org).

Key words: Data reduction

1. Introduction

Despite hundreds of programs exist for various operating systems, which allow to visualize data and create graphs of different complexity, it takes time to run the program, load the data file. Many programs may draw graphs with automatically determined limits with further possibility of formatting the graph.

One of the aims of this program is fast visualizing the data, which allows the batch mode. Besides, it has additional possibilities oriented to data checking and preliminary data reduction. A part of these features is oriented to specific astronomical needs, partially connected to the variable star research. After running, the data are visualized with determining the limits automatically. The cursor marks the point, the characteristics of which are shown. This possibility allows to check the data, especially outstanding points and, if decided that this point is wrong, to delete (obscure) it. If the data are the compilation, there may be groups of points of joint origin (same filter, telescope, observer), which should be deleted or analyzed. There is a possibility of a convertation of times to phases and vice versa, to scale and transform coordinates from linear to logarithmic and back, inverting etc.

The arrays reserved are the following: t_i - argument; x_i - signal; s_i - error estimate (sometimes may be used as a working array); e_i - working array. Contrary to electronic tables, the number of arrays is fixed. The graph always show a dependence of x_i vs t_i. Hereafter the value i will indicate a number of the current point, i_1 and i_2 are the first and last numbers of the point seen at the current screen.

The program is issued "as is", without warranty of any kind. If using, please refer to this paper.

The program is rapidly operated by "hot keys", contrary to the menu-oriented mouse control, which is usually applied in professional packages. Such an approach allows to speed up the work after learning the keys for most frequently used operations. The program can be classified as a *viewer* or simplified *editor,* complementary to that used in the shells like Norton Commander, Volkov Commander or FAR.

Running the program

The simplest way to run is just to launch "*o.exe*". Then the promt for the file name occurs. The input file should contain columns of numbers separated by spaces. The commas and the tabulation symbol (#9) are not acceptable. It is possible to add one to four parameters in the fixed sequence:

[path]o.exe [[FileIn] [[Column] [[FileOut] [Skip]]]],

where *[FileIn]* is the input file. Parameter *[Column]* is the column number - default is 0 for the "time" (t) axis and 1 for the "signal" (x) axis. If the column is set to zero, $t_j = j$, $j := 1..N$ where j is the trial number of observations, and N is their total number read by the program. If changing the column from within the program ("**C**" button, one may use negative values to use their absolute values without changing t from current values to column zero. The parameter *[FileOut]* is the name of the output file, to which one may write

results of data transformation/reduction. There is a special value "-", which will force typing to the screen the input data. This option is needed, if there are errors in the input data ("illegal real", often appears, if there are additional symbols or absence of the space separating the columns). The last parameter allows to skip [*skip*] lines at the beginning of the file. So large files may be viewed or processed in parts.

To exit the program, press **F10**, **"Escape"** or **"Enter"**; other "hot keys" are not duplicated;

The 16-bit "lite" version *O.EXE* is compiled with Turbo Pascal 6.0 (www.borland.com) and works with the number of data $N \leq 10\,000$, whereas the 32-bit version *OO.EXE* compiled with the Free Pascal (www.freepascal.org) has a limit $N \leq 300\,000$ and a lot of additional features.

If the number of data exceeds the limit, only a limiting value is read. To read other parts of the file, one may skip N_B lines. If the data are not sorted according to the argument, a note appears with a listing of inverse $t_{i-1} > t_i$. In the 32-bit version, it is possible to sort these data in ascending order of t_i. In the 16-bit version, the program determines the graphic resolution and automatically uses monochrome Hercules or 16-color EGA mode. The appropriate driver *herc.bgi* or *egavga.bgi* should be copied in a fixed directory *c:\Andronov*. In the 32-bit version, only EGA mode is possible.

For further designations, we use symbols $^{@}$("Alt"), $^{\Uparrow}$("Shift") and $^{\wedge}$("Ctrl"). If few digits are listed after #, e.g. #123, this corresponds to a special symbol with this ASCII code. These digits should be pressed only at the numerical keyboard, after *Alt* is pressed, and releasing *Alt* after the code is entered completely. Alternatively, $^{@}$6 means "Alt 6" (6 at the main, not numerical, keypad). The letters may be inserted in any case. Some "hot keys" are binary, e.g. *Q R* means that the keys *Q* and *R* are pressed consequently. The number of the current point is designated as i, whereas usage of the index j means that all data points are transferred: $j := 1..N$. If the index $j-1$ is used, then the cycle is $j := 2..N$.

The output files with fixed names corresponding to different functions, have the "*.dat*" extension, if they may be used as columns of data for further viewing/plotting, and have no extension, if they contain values of various nature. All these files have an underbrace "_" symbol as the first mone in the file name.

The "hot keys" used in both versions, they are shown in the **bold** font, whereas the advanced commands, which are used in the 32-bit version only, are shown in the *slanted* font.

The first used line of the file is a control one. The program uses it for automatic definition of the format, which is used for the point and axis labels and the output. An entry likewise "-.1" is interpreted as -0.1 with adding a missing position. The data should be

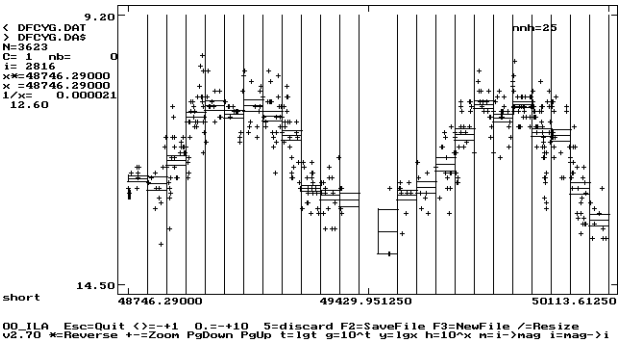

Figure 1: Interface of the program. Part of the light curve of pulsating variable star DF Cyg from the AFOEV database. The parameters listed at the left part of the figure (up to bottom):
input file (*dfcyg.dat*), output file (*dfcyg$.dat*), number of data $N = 3623$, number of column $m = 1$, number of lines skipped $nb = 0$, number of current point marked with a vertical line $i = 3623$, current argument t_i (x*), current argument t_i (x, not affected), inverse argument $1/t_i$, value of the signal x_i. The numbers at the borders show the limits of the argument and signal, respectively. Two bottom lines contain a short help. After pressing **F4**, the interval is splitted into $m(= 20)$ subintervals separated by vertical lines. The horizonatal lines show mean values and $\pm 1\sigma$ values.

in fixed (not exponential) format in this line, whereas the rest may be in both formats. The format may be changed by pressing the "**J**" button - "$j1 : j2$" (j1 digits, including j2 digits after decimal point) for t and $j3 : j4$ for x.

After reading the columns for t and x, the data are plotted with automatic determination of the limits and corresponding scaling.

Note, there is no multiple "undo", thus, using "undo" with switches, one should use them in the reverse order (likewise hierearchical loops).

To make the work with the program faster, the operation is based on the "hot keys" instead of menu. Here we present their list with a short description.

Screen changes

+ - zoom (twice pixels per argument interval);
− - unzoom (twice argument interval per pixel);
L - input axis limits;
R - restore automatic limits;
E - center current point at the screen;
/ - resize current window according to the signal limits;
***** - change direction of the signal axis;
| - initial resize (set signal borders to limits for all data);
F6 , $^{\wedge}$**left** - shift window to the left by a half-width;
F7 - make current point the left limit of the interval;

F8 - make current point the right limit of the interval;
F9 , ^**right** - shift window to the left by a half-width;
J - change output format of the argument ($j1 : j2$) and signal ($j3 : j4$);

Tab - swap $t_j \Longleftrightarrow x_j$;
^**Tab** - swap $x_i \Longleftrightarrow s_i$.

Change the current point

Generally, if changing the point number i, it will be checked for (and, if needed, changed to) the range $[1, n]$, numbering neglecting skipped lines.
Home - first point of the screen;
End - last point of the screen;
^**Home** - first point of data;
^**End** - last point of data;
Down - lowest point of the screen;
Up - upper point of the screen;
Del, **5** - delete point above cursor;

$\leftarrow$, "**3**" - i:=i+1;
$\rightarrow$, "**1**" - i:=i-1;
0 - i:=i-10;
. - i:=i+10;
4 - i:=i-100;
5 - i:=i+100;
Z - i:=i-1000;
X - i:=i+1000;
7 - previous local extremum;
9 - next local extremum;
V - correct current point assuming regular argument spacing and cubic interpolation using 4 symmetric points, i.e. $x_i := (4 * (x_{i-1} + x_{i+1}) - x_{i-2} - x_{i+2})/6$ for $2 < i < n - 1$; this option may be used, if the point is erreneous and should be deleted, but from some reason the gap in the data is less wanted (e.g. if computing an autocorrelation function, fast Fourier transform etc.);
")" - delete all points with current signal value (remove horizontal line). This option may be used, if such a value indicates wrong value (out of detection);

Data transformation switches

First click makes changes, second one on the same button - makes an inverse transformation. Please note: the inverse transformation will be made for *current* vector, so, if it was changed between pressing the buttons, the restored vector will *not coincide* with the initial one. If restoring, the rule is simple: the inverse transforms should be made in an inverse order.

M - the data x are transferred from intensities to magnitudes in the sense $x_j := -2.5 \lg(x_j/x_{max})$;
I - the data x are transferred from magnitudes to intensities in the sense $x_j := 10^{-0.4(x_j - x_{min})}$;
T - linear/log time scale: $t_j := \lg(t_j)$;
Y - linear/log signal scale: $x_j := \lg(x_j)$;
H - log/linear signal scale: $x_j := 10^{x_j}$;
G - exponential/linear scale: $t_j := 10^{t_j}$;
\ - inverse argument $t_j := 1/t_j$ (e.g. time-frequency switch);
F - time - phase (the initial epoch T_0, period P and quadratic term Q may be separately read into the program after pressing the buttons **O**, **P** and **Q**, respectively. By default, they are read from the file *c:\Andronov\c.z* .
% - add unity to signal from current point to the end (used for a diagram phase vs time);
^ (shift 6) - subtract unity from signal from current to the end;
#2 ..*#5* - divide current signal value x_i by 2..5 (this option is used e.g. when plotting time intervals between characteristic points, and few moments between them may be lost;)
^**Z** - decrease shift u by unity and plot a map $(x_k, x_{k+u}$, $k = 1, n - u)$; minimal value of $u = 1$;
^**X** - increase shift u by unity and plot a map $(x_k, x_{k+u}$, $k = 1, n - u)$; maximal value of $u = n - 2$;
^**R** restore the arguments from a working array and show a full interval of data;
⇑**F9** - divide: $x_i := x_i/t_i$, $i := 1, n$; if $t_i = 0$ then $x_i := 10^{32}$;

Time-phase transformations are made according to the linear expression $t = T_0 + P_0 \cdot E + Q \cdot E^2$, the phase is computed as $\phi = E - E_j$, where $E_j = \mathrm{INT}\ (E)$ is the integer cycle number;

O - input initial epoch T_0 and write to file;
P - input period P and write to file;
[- input quadratic term and write to file;
F - toggle t_i time-phase (computed for current T_0, P_0, Q);
^**S** - show phase curve for the next frequency listed in the file (cyclic rule: $i : ((i - 1) \bmod N) + 1$);
^**A** - show phase curve for the previous frequency listed in the file;

(- compute deviations from the best linear fit and write to the file *_o-c_lin.dat*;
^**P** - set period value to the value last determined with a 3-point parabolic fit (with **"Ins"** button) or current argument t_i. Initial epoch T_0 and the quadratic term Q are set to zero;
^**O** - set initial epoch T_0 to the argument of current point t_i and save this value;
C - change column; if $C > 0$, the zero column is set to the argument t_i, and the C^{th} column as the signal x_i; if $C = 0$, the zero column is set to the signal x_i and argument $t_i = i$; if $C < 0$, the argument t_i is not changed and the $(-C)^{th}$ column as the signal x_i; the last possibility is used e.g. to plot nonzero column as the argument: press **C 2 Enter Tab C -3 Enter**. As a result, t_i is the second column, and x_i is the third column.
^**N** - the argument $t_j := j$ and the signal x_j is not changed;

B - skip lines till the current point inclusive;

^**U** - evaluation of the smoothed value using the linear fit;

A - compute autocorrelation function for all N data: $\mathrm{ACF}_{u+1} := R_u/R_0$, $R_u := \sum_{i=1}^{N-u} x_i x_{i+u}$, the argument is set to $t_i := i - 1$.

@*6* - remove all data outside the window and replace the data with the autocorrelation function.

F2 - save all data

@**F2** - save data from the current window only;

F3 - read new file;

F4 - split current window into subintervals $l := 1..m$, compute mean $\bar{x}_l$, unbiased root mean square deviation of the data from the mean σ_l, error estimate of the mean σ_{xl}. Results $(l, \bar{x}_l), \sigma_l, \sigma_{xl})$ are written (appended) to the file _f4.dat.

^**F4** - compute mean values from m subsequent points and append to the same file;

@**F4** - histogram of the data from current window (splitted into m subintervals).

F5 - switch between "short" and "long" types of marking the intervals. In the "short" mode, the output file name is transferred from the input name by changing the last symbol to "!" ("#" in the "long" mode). The interval limits marked with **F7** and **F8** are written to the file: n_1, t_{n1}, n_2, t_{n2}. In the "long" mode, the output format is: number of points; next lines: t_i, x_i, $i := n_1..n_2$.

Ins - Determination of the extremum nearest to the current point by using 3 points $(i-1, i, i+1)$. In the left window, the information appears on the argument and signal value in this approximation. If the argument axis is transformed, the initial value is also shown. Eg., if the argument was transferred logarithmically, then one may see the value in the linear scale.

⇑**2** - switch between presenting data as small/large circles; in the 32-bit version, large symbols mode also corresponds to a local cubic interpolation line

Work with phases

‘ $x_j :=$frac$(x_j)-$ leave only fractional part of the signal;

Q F - $x_j := v - \mathrm{int}(v + 0.5)$, $v := (t_i - T_0)/P_0$, $j = 1..N$; i.e. x_j is the phase corresponding to the current values of T_0 and P_0.

& $x_j := t_{j+1} - t_j$, $j = 1..N-1$; $x_i = x_i + j_i$, where an integer j_i is determined so that $j_1 := 0$ and $|x_{i+1} - x_i| \le 0.5$. This option is used when the signal is the phase. If the phase variations are smooth, this option is usually sufficient to restore proper cycle numbering; otherwise one may use manual cycle renumbering using buttons @*5* and @*6*.

@*5* - $x_i := x_i + 1$, $i := i + 1$;

@*6* - $x_i := x_i - 1$, $i := i + 1$;

U U - Determination of the corrected values of the period P and initial epoch T_0 assuming x_i are the phases, t_i are times, and (possibly) s_i are error estimates. For such redetermination, the current values of T_0 and P_0 are used (either read from the "*c.z*" file or after pressing **O** or **P**.

Error column

K insert column number for the array s_j (e.g. with error estimates);

^**W** effective error corresponding to the mean weight: $\sigma_h = \left(\frac{1}{n}\left(\sum_{i=1}^{N} \sigma_i^{-2}\right)\right)^{-1/2}$, $\sigma_a = \frac{1}{n}\sum_{i=1}^{N} \sigma_i$, $N_{eff} = \left(\sum_{i=1}^{N} \sigma_i^{-4}\right)^2 / \sum_{i=1}^{N} \sigma_i^{-2}$.

Work with data labels

While working with data obtained from different sources, sometimes it is needed to select a part of them according to the source, e.g. to the observer. For time series analysis of the AFOEV (http://cdsarc.u-strasbg.fr/afoev) and VSOLJ (http://www.kusastro.kyoto-u.ac.jp) data, we use an intermediate format (time, magnitude, 3-symbol abbreviation of the observer). These 3 last symbols are read (in the 32-bit version) as an array and are shown at the screen with the information of the current point. Few operations are available with the subset of data having the same mark:

^- - put mark (observer's abbreviation) of the current point to be a selected one. All points with the same label will be drawn in another (???) color.

^*Del* - delete all observations with the same mark;

^*Ins* - leave only observations with the same mark, delete others;

"Non-undo" data transformation.

^*G* - generator of the signals, including periodic (sinusoidal and asymmetric), uncorrelated random (Gauss, χ^2, Cauchy et al. distributions) autoregressive models of the first and user-defined order and chaotic processes;

Q 0 - $x_j := a * x_j + b * s_j$, $j = 1..N$; the coefficients a and b are requested; this option is used e.g. if computing 1σ deviations from the signal or to compute deviations $O - C$ $(x_j - s_j)$ of the signal from the fit;

Q S - $x_j := \sqrt{x_j^2 - a^2}$, $j = 1..N$; the coefficient a is requested;

Q D - approximation of the derivative dx/dt by linear expression: $x_{j-1} := (x_j - x_{j-1})/(t_j - t_{j-1})$; $t_{j-1} := (t_j + t_{j-1})/2$.

Q Y - $d(x^2)/dt$. This option is used e.g. in the scalegram analysis. Computing the scalegram the test function $\sigma_{O-C}(\Delta t)$ (unbiased r.m.s. deviation of the data from the fit, Andronov, 1997), one may determine the derivative of the energy on $\lg \Delta t$ to obtain the scalegram.

Q 1 - t_j to the integer cycle number (e.g. primary minimum in eclipsing variables);

Q 2 - $t_j - P_0/2$ to the integer cycle number (e.g. secondary minimum);

Q P - $x_j := x_j \cdot P_0$, $j = 1..N$. This option is used to transfer phases to "O-C".

Q I - Table of interpolated data using local cubic fit using 4 nearby data points. This is a spline of power 3 and defect 2, i.e. despite the interpolation is correct for the polynomial of power up to 3, the first derivative may be discontinuous at the arguments of points. The same interpolation is used for drawing an interpolation curve (button).

Q 6 - square: $x_j := x_j^2$, $j := 1..N$.

Q S , $^{@}S$ - sort data (t_i, x_i, s_i) in ascending order of t_i.

*Q * - switch "show s_j" (used e.g. to show both data and a fit);

Q / - table of values smoothed using running parabolae

*Q ** - double the data using local cubic polynomial assuming regular spacing of arguments. This option is used to make graph more smooth.

Q T - time correction: input two times, where corrections are made. This option is used e.g., if the file contains computer times, which should be corrected using linear approximation.

Q Q - linear transformation: $x_j := a + b \cdot x_j + c \cdot t_j$, the coefficients a, b, c should be requested.

Q R - inverse computation $x_j := (x_j - a - c \cdot t_j)/b$, is made.

Q ? - computes the scalegram assuming a histogram-like fit (spline of power 0 and defect 1) of the data present at the current window for numbers of subintervals $m = 1..50$.

The weights of all the data are assumed to be the same. The following test functions are written to the file with fixed name *_scale.dat*: the trial number of subintervals m, number of subintervals with at least one point m_1; r.m.s. deviation of the smoothing value from the mean σ_C; unbiased value of the r.m.s. deviation of the data from corresponding smoothing values σ_0; the "S/N" ("signal/noise") ratio σ_C/σ_{xc}; the r.m.s. error estimate of the smoothing value σ_{xc}; the number of points in the current window n_v. The file "_scale.dat" may be studied separately, but, after running the present command, the program determines m_{max} corresponding to the maximal value of "S/N" and uses it for showing the "histogram-like". This feature is an extension of that available by pressing *F4*, because allows to use the optimal value of m instead of the user-defined one.

Launching external programs

$^{@}R$ - runs external program from the list in the file "c:\Andronov\0.mnu"; the format of the file is "Short description of the program","full name of the file to be run"; the symbol ";" in the first line means that the line is ignored as a comment; the symbol "-" in the first line means that the line contains "minus" symbols and is used for separation of different groups of programs; likewise menu in Volkov Commander, it is possible to make submenus by making a link to a file with ".mnu" extension. The only restriction for one submenu is 20 lines, just to make them not exceeding the standard PC text screen. At the end of the command line, the file name is added, as a parameter. This option is suitable, if using an external program to proceed (edit file, make computation or preparation of the print-quality figure etc). E.g., if the file contains the line *Edit,c:\vc\e.exe*, the command executed will be: *c:\command.com /c c:\vc\e.exe filename* , where *filename* is the name of the current file being processed by the program *OO.exe*. Of course, this option may be used as a launcher of *any* DOS/Windows programs, neglecting the *filename* parameter. As usual, the filenames containing spaces, should be written likewise "c:\Program Files\FAR\far.exe".

$^{@}T$ - the same as above, but the data visible at the current screen are written to the file *tmp.dat*, and this fixed file name is used as a parameter instead of the name of the current file containing all the data. This option may be suitable if processing subintervals of the data using an external program, e.g. for determination of the extrema or other local characteristics of variability. So the commands $^{@}T$ and $^{@}R$ may be used to make possibility of using external programs instead of plugins in the Windows programs.

Q B - runs "c:\Andronov\Bary.exe", the external program, which computes barycentric correction used in the variable star research and then replaces arguments t_i by barycentric values.

Running parabola fits

For smoothing the data, we have used the method of running parabolae with a filter half-width Δt (Andronov, 1990). This free parameter Δt may be determined from the scalegram analysis (Andronov, 1997). However, the scalegrams are computed using a separate program, thus Δt should be input manually after pressing $^{\Uparrow}F6$. The commands are:

$^{\Uparrow}F6$ - input Δt and compute the smoothing values at the arguments t_i and store them as s_i instead of possible error estimates.

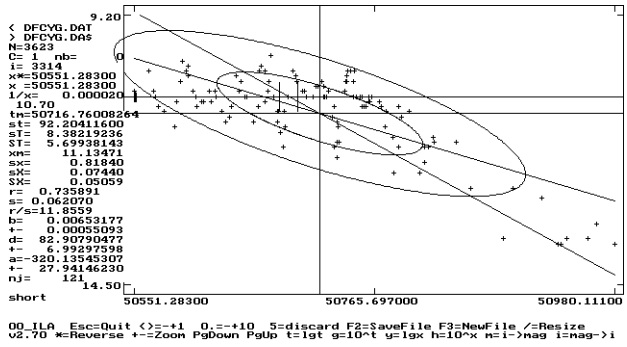

Figure 2: Results of the correlation analysis. The regression lines are $x = x_m + b(t - t_m) = a + bt$, $t = t_m + d(x - x_m) = c + dt$, the correlation coefficient r, its error estimate s, number of points nj, mean argument tm, its r.m.s. deviation from the mean, standard error of the mean for constant sT and linear ST fits.

Evaluation of the trigonometric polynomial

This option for the data visible at the screen is available by pressing two buttons *Q 8*. The fixed value of the period is used, being equal to that used for the phase computations, by default, being read from the parameter file *c.z* or changed after pressing the *P* button. In the phase curve mode (switched by the *F* button), the period is set to unity. The test functions computed for different values of the degree of the trigonometric polynomial $s = 0..s_m$ are written both to the screen and the file *_four* and are subject to make decision on the degree of the polynomial. The following columns are shown: $s-$ the degree of the polynomial (starting from 0, i.e. constant); $sig\_0 = \sigma_0-$ the unit weight error, $sxc = \sigma[x_C]-$ mean error estimate of the values of the smoothing function computed at the arguments of real data; S/N_a- "amplitude signal-to-noise ratio", which is equal to the r.m.s. deviation of the fit from the mean to the mean error estimate of the fit; *"Degeneracy"* of the matrix of normal equations being equal to 1 for diagonal matrices and 0 for degenerate ones; $Lp-$ is the "false alarm probability" to get a signal of such an amplitude for a pure white noise of the amplitude σ_0; S/N_F- is the Fischer's ratio of variances with 2 and $n_j - 2s - 1$ degrees of freedom; $Ampl-$ is the amplitude of the last harmonic wave used (with a period of P/s) and $Ampl/sig$ is the ratio of this amplitude to its error estimate. To choose the optimal value of s, one may use several test functions. A classic way is to use the Lp, which shows the Fischer's probability of 10^{-Lp}, so one may use the minimum limiting value of Lp_m, e.g. 2.5 corresponding to the "3σ" criterion and to determine a maximum value of s, for which $Lp \geq Lp_m$. We use another criterion similar to that in the method of "running parabola" scalegrams - the maximum of S/N_a - which usually corresponds to much smaller values of s, often even $s = 1$ (single harmonic fit). In our program, the optimal value of s, corresponding to this criterion, is determined and proposed as the default one in the forthcoming dialog. However, one may use other value, e.g. corresponding to the minimum of the test function $\sigma[x_C]$. However, for noisy signals, the minimum corresponds to $s = 0$, i.e. to the mean value, neglecting variations. The formulae and the description of the more advanced specialized program FDCN is described by Andronov (1994).

Further information written to the file *_four_.dat* is: the unit weight error σ_0, then the coefficients of the fit

$$x_c(t) = c_1 + \sum_{j=1}^{s}(C_{2j}\cos(j\omega t) + C_{2j+1}\sin(j\omega t) \quad (1)$$

and their error estimates. The matrix A_{ij}^{-1}, which is needed for computation of errors of the smoothing function and its characteristics (see Andronov 1994 for more details), is printed at the end of the file, and may be used by external programs. In this version, the weights are assumed to be equal for all data points.

The output file *_four_c.dat* may be created on request to be used for creating figures. It contains argument or phase, the smoothed value x_C, its error estimate σ and then two columns $x_C - \sigma$ and $x_C + \sigma$.

The file *_four.dat* contains 3 columns $(t_i, x_i, x_C(t_i))$ and may be used for further preparation of printer-quality graph. It should be noted, that, despite this file contains n points, the fit itself is computed for the points at the screen only. So, for "global" fit, the screen should contain all the data before processing.

Further development

In the current version, the main purpose is to include features, which need visualisation and interactive processing. The set of well-working specialised programs for the time series analysis may be run either separately or from within the program *OO.exe*. From this point of view, there is no necessity to translate these programs from other languages to FreePascal. However, some new features may be included in further releases of this program. The bugs may be reported to the author at *il-a@mail.od.ua*

Acknowledgements. The author is thankful to the developers of the Turbo Pascal 6.0 (www.borland.com) and Free Pascal (www.freepascal.org) for their compilers allowing to prepare these programs. For illustration, we have used the data from the AFOEV international database (cdsarc.u-strasbg.fr/afoev). Many features have been added on requests from my colleagues.

References

Andronov I.L.: 1994, *Odessa Astron. Publ.*, **7**, 49.
Andronov I.L.: 1997, *As.Ap. Suppl.*, **125**, 207.