# GIANO cookbook for data reduction

**prepared by N. Sanna, L. Origlia F. Massi, E. Oliva**

**February 2015**

## About this document

This document is intended as a first, reference guide to obtain 1-D calibrated spectra, by using the pipeline provided by the GIANO team.
It contains a brief introduction on how retrieve files from the TNG Archive (Section **1.0**) and the reference calibration data used for the spectrum extraction (Section **1.1**), some indications to take into account before running the pipeline (Sections **2**), the description of the pipeline step by step (Section **3; 3.1; 3.2**). A summary of the main actions and results is reported in Section **4**.

## 1.0 Retrieving data from the archive

All the calibration and science spectra collected during an observing night are saved in the archive. Three types of spectra are archived: all the multiple non-destructive read-outs (codified as **GIANORAW.yyy-mm-ddThh-mm-ss.000.fts)**, the quick look frames (codified as **GIANOQL.yyyy-mm-ddThh-mm-ss.000.fts**), and the final ramp-reduced spectra (codified as **GIANO.yyyy-mm-ddThh-mm-ss.000.fts)**
Going to the archive webpage (http://ia2.oats.inaf.it/archives/tng) and using the Program and Password that the user has received, it is possible retrieve the GIANO 2-D raw spectra.
The pipeline needs the ramp-reduced spectra.
In the archive searching page it is possible to select the data using the box **Obs. Type** (CALIBRATION for the wavelength calibration files, DARK for the dark frames, FLAT for the flat-field frames, SCIENCE for the objects and the sky, if requested). Once the frames are selected, click on the **Get Files** button and then on the tar file to retrieve the data.

## 2.0 Warnings before starting to run the pipeline

A comprehensive and detailed description of the pipeline, the IRAF tasks and scripts can be found in the GIANO data reduction manual. Here we summarize the main features.
The pipeline is based on a series of IRAF tasks, some of them written appositely for GIANO and contained in the ***giano_tools*** package that must be installed in a given directory. In order to do that, edit the *giano_tools.cl* file and set the full path of the directory containing the ***giano_tools*** package in the line *set gianot = "path/"*. Then edit the *login.cl* file in your IRAF home directory and add the line *task $giano tools = path/giano tools.cl.*
It is recommended to create a directory for each night of observation and reduce the full dataset of that night.

## 2.1 The data required for the spectrum extraction

Before run the pipeline, the user must be sure to have all the required calibration and science frames. In particular the necessary calibration data are: ***i***) a series of dark frames with the same exposure time of the flat-fields (typically 10 frames of 60 sec); ***ii***) a series of dark frames with the same exposure time of the U-Ne lamp files (typically 10 frames of 300 sec); ***iii***) a series of flat-field frames (typically 10 frames of 60 sec); ***iv***) one or more U-Ne lamp files (typically of 300 sec).

In case of compact sources observed by nodding on fiber, the science data are a series of target spectra acquired both with fiber A and fiber B.
In case of extended sources observed in stare mode, the science data are a series of target and sky frames.

## 3.0 GIANO spectra extraction and wavelength calibration

The pipeline for spectrum extraction requires a number of steps for file preparation and then a number of steps to identify the orders on the echellogram and performing the 1-D wavelength calibrated extraction.

## 3.1 Preparation steps

The first four steps are the same for compact objects and extended ones.

*1) Rename the *fts files in *fits files*

IRAF requires the extension *.fits* for the frames. Hence the user must rename the 2-D GIANO spectra. It can be with the command ***mv*** outside IRAF or with the command ***rename*** in IRAF.

For all the other steps the user must use IRAF and load the ***giano_tools*** package. In order to edit the parameter file of all the IRAF tasks just type ***epar*** "name of the task" (i. e. *epar giano_header*).

*2) Clean the flat-field frames and U-Ne lamps*

Both the flat-fields and the calibration lamps need the subtraction of the dark frame. This means that an averaged dark frame of 60 sec (for flat-fields) and one of 300 sec (for U-Ne lamps) must be created using the task ***imcombine.***

Then the averaged dark frame of 60 sec must be subtracted from all the flat-fields and the averaged dark frame of 300 sec from all the calibration lamps. This can be done using ***imarith.***

*3) Correct frames for bad pixels*

All the calibration and science frames must be correct for bad pixels. This can be done using the task ***clean_up*** that uses the bad pixels mask in the ***giano_tools*** package. It requires an input list of the frames, a suffix to add at the frame name (usually *cl_*) and parameters set as follows.

```
PACKAGE = giano_tools
   TASK = clean_up

imlist  =                         File with list of images:
outstr  =                    cl_  String to be added to input image name:
badima  = gianot$badpix_mask.fits  Bad pixel image:
xbox    =                     40  Size of median-filter box along X:
ybox    =                      1  Size of median-filter box along Y:
fix     =                     no  Use fixpix?:
fixfile =                         List of bad pixel masks for fixpix:
(mode   =                      q)
```

*4) Create the averaged flat-field frame and the averaged U-Ne lamp frame (if necessary)*

An averaged flat-field frame and an averaged calibration lamp frame (if more than one lamp were acquired) are required. This can be easily done using ***imcombine*** and the files corrected for the bad pixels.

*5) Subtract the background from the science frames*

Due to the different acquisition mode used for compact/point sources (nodding-on-fiber) and extended objects (stare mode), this step requires different actions for the two cases. In any case the bad-pixel corrected frames are required.

In the **nodding on fiber acquisition mode,** a series of frames with the target alternatively on fiber A and fiber B are obtained. The user needs to compute with the IRAF task ***imarith*** the (A-B) difference for each nodding cycle and possibly combine (e.g. sum) all the (A-B) frames. The resulting 2D frame will have two positive (fiber A) and two negative (fiber B) traces for each order.
In the **stare acquisition mode** different frames for the target and the sky are acquired. Both target and sky frames need to be combined and then the (target-sky) difference need to be computed. The resulting 2D frame will have four positive traces for each order (both fiber A and B are illuminated by the target light since extended).
In any case, it is better to limit the number of the science frames at ~25, when using list of files.

This is valid for all the extraction steps.

*6) Update the data headers*

It is necessary to add some fields to the headers of the calibration lamp and science spectra. This is possible using the script ***giano_header*** included in the ***giano_tools*** package that requires two different file lists, one with the U-Ne lamp(s) and one with all the science frames.

## 3.2 Order identification in the GIANO echellogram

The identification and the geometry of the 49 orders in the GIANO echellogram is done on the averaged flat-field frame. Four traces per order (196 in total), corresponding to the two sliced fibers, need to be identified. This is done by using the IRAF task ***apflatten***, which uses the package ***echelle*** and calls other tasks, namely ***apdefault***, ***apresize***, ***aptrace***, and ***apfind***. To be sure that all the orders have been identified, first of all the user needs to run the task ***giano_find_trace*** and use the output file as reference image in ***apflatten***.

In all these tasks the user needs to set parameters as follows.

```
                                  I R A F
                    Image Reduction and Analysis Facility
PACKAGE = giano_tools
   TASK = giano_find_trace

input_fr= ▉                        Frame to use as a reference to extract traces:
out_fram=                          Output frame after scattered light removal:
save_sc =                    no    Save scattered light map?
interac_=                   yes    Fit traces interactively?
displine=                  1000    Dispersion line:
gnsum   =                    10    Number of dispersion lines to sum or median:
gminsep =                    3.    Minimum separation between spectra:
gmaxsep =                   52.    Maximum separation between spectra:
(mode   =                     q)
```

```
                              IRAF
                    Image Reduction and Analysis Facility
PACKAGE = echelle
   TASK = apflatten

input   = ▉                     List of images to flatten
output  =                       List of output flatten images
(apertur=                 1-196) Apertures
(referen=                      ) List of reference images

(interac=                   yes) Run task interactively?
(find   =                    no) Find apertures?
(recente=                    no) Recenter apertures?
(resize =                    no) Resize apertures?
(edit   =                   yes) Edit apertures?
(trace  =                   yes) Trace apertures?
(fittrac=                   yes) Fit traced points interactively?
(flatten=                   yes) Flatten spectra?
(fitspec=                   yes) Fit normalization spectra interactively?

(line   =                  1000) Dispersion line
(nsum   =                    10) Number of dispersion lines to sum or median
(thresho=                  10.) Threshold for flattening spectra

(pfit   =                 fit1d) Profile fitting type (fit1d|fit2d)
(clean  =                    no) Detect and replace bad pixels?
(saturat=                 INDEF) Saturation level
(readnoi=                    0.) Read out noise sigma (photons)
(gain   =                    1.) Photon gain (photons/data number)
(lsigma =                    4.) Lower rejection threshold
(usigma =                    4.) Upper rejection threshold

(functio=              legendre) Fitting function for normalization spectra
(order  =                     1) Fitting function order
(sample =                     *) Sample regions
(naverag=                     1) Average or median
(niterat=                     3) Number of rejection iterations
(low_rej=                    3.) Lower rejection sigma
(high_re=                    3.) High upper rejection sigma
(grow   =                    0.) Rejection growing radius
(mode   =                     q)

                              IRAF
                    Image Reduction and Analysis Facility
PACKAGE = imred
   TASK = echelle

(extinct= onedstds$kpnoextinct.dat) Extinction file
(caldir =                       ) Standard star calibration directory
(observa=           observatory) Observatory of data
(interp =                 poly5) Interpolation type
(dispaxi=                     1) Image axis for 2D/3D images
(nsum   =                     1) Number of lines/columns/bands to sum for 2D/3D images

(databas=               database) Database
(verbose=                    no) Verbose output?
(logfile=                logfile) Text log file
(plotfil=                      ) Plot file

(records=                      ) Record number extensions
(version= ECHELLE V3: July 1991)
(mode   =                     ql)
($nargs =                      0)
```

```
                          IRAF
              Image Reduction and Analysis Facility
PACKAGE = echelle
   TASK = apdefault

(lower  = ▉                -2.) Lower aperture limit relative to center
(upper  =                   2.) Upper aperture limit relative to center
(apidtab=                     ) Aperture ID table

(b_funct=             chebyshev) Background function
(b_order=                     2) Background function order
(b_sampl=          -10:-6,6:10) Background sample regions
(b_naver=                  -100) Background average or median
(b_niter=                     3) Background rejection iterations
(b_low_r=                    1.) Background lower rejection sigma
(b_high_=                    1.) Background upper rejection sigma
(b_grow =                    0.) Background rejection growing radius
(mode   =                     q)
```

```
                          IRAF
              Image Reduction and Analysis Facility
PACKAGE = echelle
   TASK = apfind

input   = ▉                    List of input images
(apertur=                  196) Apertures
(referen=                     ) Reference images

(interac=                  yes) Run task interactively?
(find   =                  yes) Find apertures?
(recente=                  yes) Recenter apertures?
(resize =                   no) Resize apertures?
(edit   =                  yes) Edit apertures?

(line   =                 1000) Dispersion line
(nsum   =                   10) Number of dispersion lines to sum or median
nfind   =                  196  Number of apertures to be found automatically
(minsep =                   3.) Minimum separation between spectra
(maxsep =                  52.) Maximum separation between spectra
(order  =           increasing) Order of apertures
(mode   =                    h)
```

```
                          IRAF
              Image Reduction and Analysis Facility
PACKAGE = echelle
   TASK = apresize

input   = ▉                      List of input images
(apertur=                      ) Apertures
(referen=                      ) Reference images

(interac=                   no) Run task interactively?
(find   =                  yes) Find apertures?
(recente=                   no) Recenter apertures?
(resize =                   no) Resize apertures?
(edit   =                  yes) Edit apertures?

(line   =                INDEF) Dispersion line
(nsum   =                    1) Number of dispersion lines to sum or median
(llimit =                  -2.) Lower aperture limit relative to center
(ulimit =                   2.) Upper aperture limit relative to center
(ylevel =                INDEF) Fraction of peak or intensity for automatic width
(peak   =                  yes) Is ylevel a fraction of the peak?
(bkg    =                  yes) Subtract background in automatic width?
(r_grow =                   0.) Grow limits by this factor
(avglimi=                   no) Average limits over all apertures?
(mode   =                    q)
```

```
                                  I R A F
                Image Reduction and Analysis Facility
PACKAGE = echelle
   TASK = aptrace

input   = ▉                    List of input images to trace
(apertur=                     ) Apertures
(referen=                     ) List of reference images
(interac=                   yes) Run task interactively?
(find   =                   yes) Find apertures?
(recente=                    no) Recenter apertures?
(resize =                    no) Resize apertures?
(edit   =                    no) Edit apertures?
(trace  =                   yes) Trace apertures?
(fittrac=                   yes) Fit the traced points interactively?

(line   =                  1000) Starting dispersion line
(nsum   =                    10) Number of dispersion lines to sum
(step   =                    10) Tracing step
(nlost  =                     3) Number of consecutive times profile is lost before quitting

(functio=               spline3) Trace fitting function
(order  =                     3) Trace fitting function order
(sample =                     *) Trace sample regions
(naverag=                     1) Trace average or median
(niterat=                     3) Trace rejection iterations
(low_rej=                    3.) Trace lower rejection sigma
(high_re=                    3.) Trace upper rejection sigma
(grow   =                    0.) Trace rejection growing radius
(mode   =                     q)
```

The outputs of this task are two: a fits file corresponding to the flatten image, that the pipeline does not use, and a subdirectory named *database* where the text file with the position of the orders is saved.

In order to use the reference positions of the 196 traces identified by ***apflatten***, and properly extract the orders in the science frames, the user needs to run the task ***split_file***. This task still works on the averaged-flat-field frame and consists of two actions: on one side it copies four time the 2D frame with suffix *_lowest, _middown, _midup, _topmost* (see Figure 1) in the main directory, on the other side in the subdirectory *database* it generates four files, with suffix *_lowest, _middown, _midup, _topmost*, each one containing the corresponding trace position of each order.

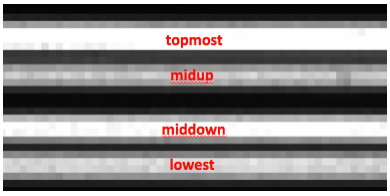


**Figure 1**. Portion of the echellogram showing the four traces of each order, corresponding to the two sliced fibers.

The parameters in the ***split_file*** task must be set as follows.

```
PACKAGE = giano_tools
   TASK = split_file

imtrace =                      Reference image with traces (without extention):
numf    =                   4  :Number of traces per order (2,4)
ext     =                fits  Image file extension (fits,imh,...):
dir     =            database  Directory holding the aperture files (default database)
(mode   =                   q)
```

Also the 2D calibration lamp frame and the science (A-B) or (target-sky) frames are split in four files, one per each trace, by using the task ***copy_file***. The outputs are the frames with suffix *_lowest, _middown, _midup, _topmost*.

In case of (A-B) frames the two negative traces are automatically transformed in positive ones.

```
PACKAGE = giano_tools
   TASK = new_copy_file

anysub  =                 yes  Are there any subtracted (A-B) frames?
inlist  =                      File listing subtracted (A-B) frames:
inmark  =                   1  Positive peaks are at the lower (1) or upper (2) two tracks?:
nonodd  =                      File listing frames with positive peaks at all tracks (calibration lamps, etc.):
(mode   =                   q)
```

## 3.2 Spectrum extraction

At this point all the preliminary steps are done and everything is ready for the proper 1-D spectrum extraction and wavelength calibration.
The user needs to prepare four lists of science frames, one per trace *(_lowest, _middown, _midup, _topmost*) and run the task ***apedit***, setting the parameters as follows.
In case of a high number of science frames (typically > 25) the user needs to prepare more lists to avoid problems with some IRAF tasks (e.g ***apedit***, ***doecslit***, ***sarith***).

```
                                  I R A F
                       Image Reduction and Analysis Facility
PACKAGE = echelle
   TASK = apedit

input   = ▉           @low_new  List of input images to edit
(apertur=                 49) Apertures
(referen=     flatmedio_lowest) Reference images

(interac=                yes) Run task interactively?
(find   =                 no) Find apertures?
(recente=                 no) Recenter apertures?
(resize =                 no) Resize apertures?
(edit   =                yes) Edit apertures?

(line   =               INDEF) Dispersion line
(nsum   =                  10) Number of dispersion lines to sum or median
(width  =                  4.) Profile centering width
(radius =                  4.) Profile centering radius
(thresho=                  0.) Detection threshold for profile centering
(mode   =                   q)
```

Once all the orders have been identified, the user can extract and calibrate in wavelength each trace of each order, using the IRAF task ***doecslit***, which also calls the task ***sparams.***
The user needs to set parameters of the ***doecslit*** and ***sparams*** tasks as follows.

```
                                    I R A F
                         Image Reduction and Analysis Facility
PACKAGE = echelle
   TASK = doecslit

objects = ▉                    List of object spectra
(apref  =                     ) Aperture reference spectrum
(arcs   =                     ) List of arc spectra
(arctabl=                     ) Arc assignment table (optional)
(standar=                     ) List of standard star spectra

(readnoi=                   8.5) Read out noise sigma (photons)
(gain   =                   2.2) Photon gain (photons/data number)
(datamax=                INDEF) Max data value / cosmic ray threshold
(norders=                    49) Number of orders
(width  =                    5.) Width of profiles (pixels)

(dispcor=                   yes) Dispersion correct spectra?
(extcor =                    no) Extinction correct spectra?
(fluxcal=                    no) Flux calibrate spectra?
(resize =                    no) Resize object apertures?
(clean  =                    no) Detect and replace bad pixels?
(trace  =                    no) Trace object spectra?
(backgro=                  none) Background to subtract
(splot  =                    no) Plot the final spectra?
(redo   =                    no) Redo operations if previously done?
(update =                    no) Update spectra if cal data changes?
(quicklo=                    no) Approximate quicklook reductions?
(batch  =                    no) Extract objects in batch?
(listonl=                    no) List steps but don't process?

(sparams=                     ) Algorithm parameters
(mode   =                    ql)
```

IRAF
Image Reduction and Analysis Facility

```
PACKAGE = echelle
   TASK = sparams

(line   =                  INDEF) Default dispersion line
(nsum   =                     10) Number of dispersion lines to sum or median
(extras =                     no) Extract sky, sigma, etc.?

                                  -- AUTOMATIC APERTURE RESIZING PARAMETERS --
(ylevel =                   0.05) Fraction of peak or intensity for resizing

                                  -- TRACE PARAMETERS --
(t_step =                     10) Tracing step
(t_funct=                spline3) Trace fitting function
(t_order=                      2) Trace fitting function order
(t_niter=                      1) Trace rejection iterations
(t_low  =                     3.) Trace lower rejection sigma
(t_high =                     3.) Trace upper rejection sigma

                                  -- BACKGROUND AND SCATTERED LIGHT PARAMETERS --
(b_funct=               legendre) Background function
(b_order=                      2) Background function order
(b_naver=                   -100) Background average or median
(b_niter=                      3) Background rejection iterations
(b_low  =                     1.) Background lower rejection sigma
(b_high =                     1.) Background upper rejection sigma
(buffer =                     2.) Buffer distance from apertures
(apscat1=                       ) Fitting parameters across the dispersion
(apscat2=                       ) Fitting parameters along the dispersion

                                  -- APERTURE EXTRACTION PARAMETERS --
(weights=                   none) Extraction weights (none|variance)
(pfit   =                  fit1d) Profile fitting algorithm (fit1d|fit2d)
(lsigma =                     3.) Lower rejection threshold
(usigma =                     3.) Upper rejection threshold

                                  -- ARC DISPERSION FUNCTION PARAMETERS --
(thresho=                    10.) Minimum line contrast threshold
(coordli= gianot$U_Ne_Ar_lines_list_cl.dat) Line list
(match  =                   0.1) Line list matching limit in Angstroms
(fwidth =                    6.) Arc line widths in pixels
(cradius=                   10.) Centering radius in pixels
(i_funct=               legendre) Echelle coordinate function
(i_xorde=                      4) Order of coordinate function along dispersion
(i_yorde=                      4) Order of coordinate function across dispersion
(i_niter=                      3) Rejection iterations
(i_low  =                     3.) Lower rejection sigma
(i_high =                     3.) Upper rejection sigma
(refit  =                    yes) Refit coordinate function when reidentifying

                                  -- AUTOMATIC ARC ASSIGNMENT PARAMETERS --
(select =                average) Selection method for reference spectra
(sort   =                     jd) Sort key
(group  =                    ljd) Group key
(time   =                     no) Is sort key a time?
(timewra=                   17.) Time wrap point for time sorting

                                  -- DISPERSION CORRECTION PARAMETERS --
(lineari=                    yes) Linearize (interpolate) spectra?
(log    =                     no) Logarithmic wavelength scale?
(flux   =                    yes) Conserve flux?

                                  -- SENSITIVITY CALIBRATION PARAMETERS --
(bandwid=                   10.) Bandpass widths
(bandsep=                   10.) Bandpass separation
(s_inter=                    yes) Graphic interaction to examine/define bandpasses
(s_funct=                spline3) Fitting function
(s_order=                      1) Order of sensitivity function
(fnu    =                     no) Create spectra having units of FNU?
(mode   =                      q)
```

The first time that the user run the task ***doecslit*** on a trace (e.g. *_lowest* files), a series of master U-Ne lines must be identified (see Figure 2.). This is done interactively.

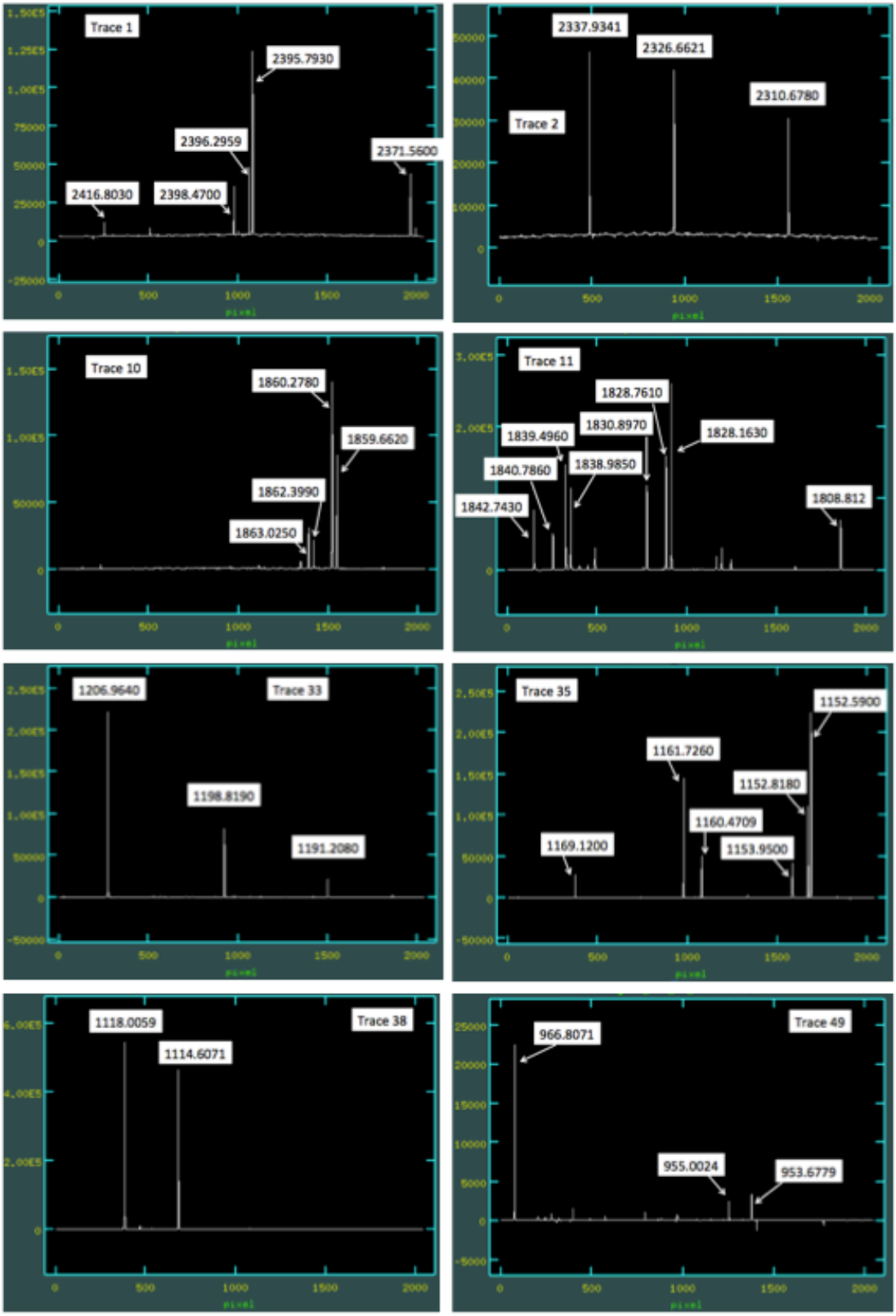


**Figure 2**. Master list of U-Ne lines that needs to be identified interactively.

To add the wavelength the user selects a line and type ***m,*** inserts the value and then ***return***. In this way the lines are registered. To change order type ***o*** and the order number. When all the lines in the selected eight orders have been registered, they must be fit in order to identify several lines in all the 49 orders. To do that, the user should press ***f*** to fit and then ***q*** and type ***:maxfeatures 500***, ***return, l*** and ***f*** and then ***q***. When this identification is completed, press ***q*** and all the spectra listed as input are automatically extracted.

In order to apply the wavelength calibration solution found for a given trace to the other three traces, the user can run the task ***giano_reidentify.*** It requires the already extracted lamp (suffix *.ec*) and the lamp to be calibrated. The user must run this task three times, one per trace.

At this point all the traces of the U-Ne lamps have been calibrated. To extract the 1-D spectra just run ***doecslit*** three times, using as input the list of the target traces not yet extracted.
The first time that the user run the task on a new trace, the wavelength calibration solution will be shown. Press ***q*** to confirm and automatically all the spectra will be extracted.

To obtain the final 1-D wavelength calibrated spectra other two actions are required: the correction for the flat-field and the combination of the four traces.
The first action can be easily done using the task ***flat_1D*** that extract the 1-D spectra of the four traces of the averaged flat-field and divide each 1-D target spectrum for the corresponding flat spectrum. Then for each target the four 1-D extracted and wavelength calibrated spectra can be combined together using the task ***sarith***.

The final results are the 1-D extracted and wavelength calibrated spectra for each target as a fits table (*name_fl.ec.fits*).
For each target also the 1-D extracted and wavelength calibrated spectrum not corrected for flat-field is also saved as a fits table (*name.ec.fits*).
If the user prefers the ASCII format, it is possible to convert the fits table in 49 .txt files (*name_fl.ec_#order.txt*), one for each order. This can be done using the task ***fits2text***.