Guide to the NESSIE Reduction Task DONESSIE

Francisco Valdes

IRAF Group - Central Computer Services
National Optical Astronomy Observatories††
September 1990

ABSTRACT

The IRAF **nessie** Reduction Package is specialized for the extraction, flat fielding, fiber throughput correction, wavelength calibration, and sky subtraction of *Nessie* fiber spectra. It contains the general tools needed for each reduction step. However, the main emphasis of this package is to streamline the complete reduction of *Nessie* data by combining all the steps into a single task called **donessie**. This guide describes what this task does, it's usage, and parameters.

1. Introduction

The IRAF **nessie** Reduction Package is specialized for the extraction, flat fielding, fiber throughput correction, wavelength calibration, and sky subtraction of *Nessie* fiber spectra. It contains the general tools needed for each reduction step. However, the main emphasis of this package is to streamline the complete reduction of *Nessie* data by combining all the steps into a single task called **donessie**. The goals of this task are:

[1] Combine all the reduction steps into a single task.

[2] Do the interactive calibration setup first, and only as needed, and allow the repetitive extraction, flat fielding, wavelength calibration, and sky subtraction to be done noninteractively (and possibly as a background or batch job).

[3] Perform much of the record keeping and processing checks so that only unprocessed spectra are processed.

[4] Implement special operations and algorithms for this type of data which require many steps with the general tools.

This guide describes only the special processing task **donessie**. The other reduction and analysis tasks available in the **nessie** package are described in their help pages. It is assumed that the reader is familiar with the IRAF environment and command language. For new IRAF users see one of the introductory guides given in the references and the user support staff.

The guide begins with a usage outline. If you have used this package before and simply need a refresher or checklist then this section is for you. A new user can read this to see how the task will be used or skip to later sections and return to it when ready to reduce some data. The following section outlines the basic reduction steps implemented in the task. This section is kept relatively brief but does fully describe what the task does. The remaining sections describe the parameters, data files, and algorithms in varying detail. There are many options and features available; too many to be covered even superficially. This guide concentrates on the basic features and the options recommended by the author and staff. Since the reduction task is based on more general tasks, identified in the discussion, all the options and features are described more fully in the help pages for these tasks.

2. Usage Outline

This section outlines the basic usage. Later sections go into greater depth on the parameters and what the task does. The outline may be used as a refresher or checklist for those familiar with this task.

---

††Operated by the Association of Universities for Research in Astronomy, Inc. under cooperative agreement with the National Science Foundation.

[1] Prepare aperture identification tables for each plugboard to be processed. Beam numbers should be assigned as follows: 0 for sky, 1 for object, 2 for arc.

[2] Prepare an arc aperture replacement file if needed.

[3] Prepare an arc assignment table if desired.

[4] If there are many spectra per setup prepare "@ files" for the objects, the standard comparison lamps, and any auxiliary arc spectra.

[5] Set the basic **nessie** package parameters.

[6] Modify any desired algorithm parameters in the parameter set **params**.

[7] Set the parameters for **donessie** and run the task. This is repeated for each distinct data set. The flat fields and arcs may or may not need to be redefined for new plugboards.

[8] The spectra are found and default fixed sized apertures are assigned. You are asked if you want to resize the apertures. The default resizing is based on finding the level which is 5% of the peak. You then have the option of entering the aperture editing loop to check the aperture positions, numbers, identifications, and sizes. Finally the spectra are traced. You are asked if you want to review and modify the traces and functions fit. This can be done selectively and turned off at any point with "NO".

[9] If the data have been flat fielded separately specify a null name, "" with no blanks, for the flat field spectrum. However, normally the flat field corrections are done after extraction by **donessie**. The flat field and (optional) sky flat spectra are used to generate a normalized response correction including fiber throughput. If the `fitflat` option is used then the average extracted flat field spectrum from all the fibers is plotted and you interactively fit a smooth function using the **icfit** commands. The fitted function is then divided into each fiber flat field. If a sky flat spectrum is specified then the total sky counts through each fiber are used to correct the total flat field counts. Note that it is legal to only specify a sky or dome flat image to compute just a throughput correction. The final response spectra are normalized to a unit overall mean.

[10] If the dispersion correction option is not selected the task goes to step [13]. The first arc in the standard comparison arc list, called the reference arc, is extracted and flat fielded automatically and the task **identify** is executed on the middle fiber. Identify a few arc lines with 'm', use the 'l' line list identification command to automatically add additional lines and fit the dispersion function. Check the quality of the dispersion function fit with 'f'. When satisfied exit with 'q'.

[11] The remaining fibers are reidentified. You have the option to review the line identifications and dispersion function for each fiber and interactively add or delete arc lines and change fitting parameters. This can be done selectively and turned off at any point with "NO".

[12] If auxiliary arcs (dome lights) are defined, the first arc in the auxiliary arc list is extracted and flat fielded automatically, the dispersion function from the reference arc is applied, and the task **identify** is executed on the middle fiber. Mark and identify one or more lines for tracking the wavelength zero point shift. When satisfied exit with 'q'. The same lines in the remaining fibers are automatically reidentified. Note that these shift arcs have precedence over any internal arc fibers.

[13] The object spectra are now automatically processed. If the `batch` flag is set and the `skyedit` and `splot` flags are not set then the object spectra are processed by submitting a batch job and control returns to the user. Note that the `skyedit` and `splot` flags may be turned off with "NO" during the course of processing the objects, after which any remaining spectra will be processed as a batch job.

[14] When sky subtracting, the individual sky spectra may be reviewed and some spectra eliminated if the `skyedit` option is selected, by pointing the cursor and typing 'd'. The last deleted feature may be recovered with the 'e' key. After exiting the review you are asked for the combining option. The type of combining may be dictated by the number of sky fibers to be used.

[15] Additional objects can be processed or the dispersion and sky subtraction options can be turned on if not done previously. The task will not reprocess spectra unless forced by the `update` or `redo` flags.

3. Processing Outline

This section outlines the operations performed by the processing task **donessie**. One goal of this task is to minimize the number of interactive steps. The (possibly) interactive steps are marked with an asterisk below. A second goal is to skip steps and spectra done previously.

[0] The input consists of object images, a spectrum position reference image, a comparison lamp image, and optional lamp flat field image, sky flat field image, additional comparison lamp images, mercury or sky line images, a table of aperture identifications, a table of arc replacement apertures, and an arc assignment table.

The output consists of the extracted and calibrated spectra in two or three dimensional *multispec* format. Optional output are a verbose log file, a plot file, and the sky spectrum subtracted from the object fibers. In addition there will be various database files created during processing and containing information which may be of interest.

[1]* If there are no aperture definitions in the database for the specified aperture reference spectrum then the apertures are defined and traced. The apertures are first found automatically across the center of the image and the user is given the option of automatically resizing the apertures. The user is then given the option of reviewing and possibly modifying the apertures. The user is also given the option of interactively reviewing and adjusting the spectrum traces and fitting function.

[2] The flat field response spectrum, which includes a fiber throughput correction, is computed, if not done previously, when a flat field spectrum and/or a sky flat spectrum is specified. The response spectrum name is formed from the concatenation of the flat field name, the sky flat field name, and the extension "norm.ms". Thus, changing either the flat field or sky flat field will initiate a new response function.

[2a]* If the `fitflat` option is selected and a flat field is specified the average (over all fibers) of the extracted flat field spectra is plotted and the user fits a smooth function using the **icfit** commands. The fitted function is divided into the individual flat field spectra to remove the basic shape of the flat field source spectrum. This has the advantage of largely preserving the instrumental counts in the extracted objects. Note that the interactive part of the function fitting may be turned off and the default fitting parameters set using parameters in the **params** parameter set.

[2b] If both a lamp and sky flat field are used, the sky spectra are extracted and divided by the flat field from [2a]. The mean counts in each fiber are found and a correction is applied to the flat field to bring all the sky flat field counts to the same value. This corrects for possible differences in illumination of the fibers between the flat field source and sky sources.

[2c] If only a sky flat field (which might be a lamp flat field) is specified then the sky spectra are extracted and the total counts through each fiber are computed for the throughput correction.

[2d] The final response spectra are normalized by the mean counts over all fibers.

[3] If dispersion correction is not selected then steps [3] and [4] are skipped. Otherwise the first arc in the standard comparison arc list is selected as the *dispersion reference arc*. All other arcs will be reidentified against this arc. Also the final linearized wavelength coordinates are based on this reference arc. If the reference arc has been processed this step is skipped.

[3a] The reference arc is extracted and flat fielded.

[3b]* The task **identify** is executed on the middle fiber. The user defines the dispersion function interactively by identifying arc lines and fitting a function. Some knowledge of this task is required.

[3c]* After exiting with a dispersion function the remaining fibers are reidentified. A statistics line is printed for each reidentified fiber giving the RMS of the fitted line wavelengths relative to the true wavelengths. The user may examine and modify each set of line reidentifications and dispersion fit if desired. This may be done selectively and turned off at any point.

[3d]* The user is queried for the desired final linear dispersion parameters. All spectra will ultimately be put on the same wavelength scale. The default dispersion parameters are based on the range and number of pixels in the reference arc. However, the user may select any dispersion parameters desired.

[3e] The extracted reference arc is linearized.

[4] Auxiliary arcs may be used to track the shift of the wavelength zero points. Note this takes precedence over simultaneous arc fibers which provide the same type of correction. The first arc in the list is selected as the reference for this type of arc. The auxiliary arc lines used may be mercury lines from the dome lights or possibly strong night sky lines. If the auxiliary reference arc has been processed this step is skipped.

[4a] The reference auxiliary arc is extracted and flat fielded.

[4b]* The reference dispersion function is transferred to the central fiber. The task **identify** is executed on the central fiber. The user marks and identifies one or more lines to be used in tracking the zero point shifts.

[4c] The remaining fibers are reidentified automatically.

[4d] The extracted auxiliary reference arc is linearized.

[5] Object spectra which have not been processed are processed noninteractively unless the `skyedit` option is selected. Note that extraction, dispersion correction, and sky subtraction may be done at separate times or simultaneously depending on the processing options selected. Also the `redo` and `update` flags may cause previously processed spectra to be reprocessed. If the `batch flag` is set and the `skyedit` and `splot` flags are not set then the spectra will be processed by submitting a batch job. Otherwise the task will continue in the foreground.

[5a] The object spectra are extracted and flat fielded.

[5b]* If dispersion correcting, the fundamental calibration arcs are determined either from the arc lists or an arc table. The default is to interpolate from the arc lists based on the UT parameter. *If an auxiliary arc list is specified then calibration arcs will be determined from this list, otherwise they will be determined from the primary arc list.* The selected arcs are extracted and reidentified against the reference arcs as needed.

If the objects are not extracted as a batch job, a statistics line for each fiber is printed and the user may decide whether to examine the dispersion fit interactively and possibly delete lines or change fitting parameters.

If the simultaneous arc fibers are used *and the auxiliary arc spectra are not used* then these fibers, identified by the parameters `arcaps` and `arcbeams`, are extracted and wavelength shifts from the dispersion function of the same fibers in the calibration arcs are determined. The shifts are interpolated to the extracted positions of the object fibers and applied to the dispersion functions from the calibration arcs. Once the final dispersion function has been determined for the fibers the spectra are linearized to the same wavelength parameters as the reference arc.

[5c]* If sky subtracting, the sky fibers are selected and combined. The `skyedit` option allows plotting all the individual sky spectra and deleting contaminated spectra. This option may be applied selectively or turned off at any point. This option must be turned off in order for batch processing to occur.

The default combining option is the "avsigclip" algorithm of **imcombine**. If the `skyedit` option is used you will be queried for the combining option. This allows resetting the algorithm depending on the number of sky fibers used. The combined sky is then subtracted from the specified object fibers. By default the sky fibers are also included as object fibers as a check on the quality of the sky subtraction.

[6]* If the `splot` flag is set you are queried whether to examine the spectra with the task **splot**. You may answer NO to permanently turn off this option or simply yes or no to examine the extracted spectra. Within `splot` you may select apertures to be examined with the '#' key. You may measure line positions, equivalent widths, edit out poor sky line subtractions, etc.

4. Spectra and Data Files

The basic input consists of *Nessie* object and calibration spectra stored as IRAF images. The raw CCD images must be processed to remove overscan, bias, and dark count effects. This is generally done using the **ccdred** package (see the references for information about this package). The recommended procedure is to perform the flat fielding as part of the extraction. If flat fielding is not done then one should use a sky or dome flat for fiber throughput corrections.

The task **donessie** uses several types of calibration spectra. These are dome quartz lamp flat fields (the internal quartz lamp is not recommended due to illumination problems), blank sky flat fields, comparison lamp spectra, auxiliary mercury line (from the dome lights) or sky line spectra, and simultaneous arc spectra taken during the object observation. The dome flat field and sky flat field, auxiliary emission line spectra, and simultaneous comparison fibers are optional. The sky or dome flat fields provide fiber throughput corrections. If a lamp flat field is used then the sky flat is optional assuming the lamp flat field has the same fiber illumination. If flat fielding is not done then a sky or dome flat should be used for the throughput corrections. Because only the total counts through each fiber are used sky flat exposures need not be of high quality.

The comparison lamp exposures suffer from vignetting resulting in some fibers being poorly illuminated. By rearranging the fibers in the calibration plugboard and taking additional exposures one can obtain good arc spectra through all fibers. In the processing one exposure is specified as the primary arc spectrum. If that is used for all fibers then nothing further need be done in the way of setup. However, if multiple arc exposures are made then a setup file, called an *arc replacement file*, is required to identify which fibers from which spectra are to be used in place of the fibers from the primary arc spectrum. Note that the first exposure of a multiple exposure is the only arc specified and the extracted arcs will ultimately contain data from the other exposures as specified in the replacement file. All the exposures are considered one logical observation referenced by the primary name.

The format of the arc replacement file is lines containing the primary arc image name, a secondary arc image name, and the apertures/fibers from the secondary arc to be used. There can be more than one secondary exposure though it is unlikely. Figure 1 gives an example of this setup file.

Figure 1: Example Arc Aperture Replacement File

```
cl> type arcreplace
nesjun042c nesjun049c 1,7,9,13,17,19,28,34
```

The principle arc exposure is "nesjun042c" (the only one specified in *arcs1*), the secondary arc is "nesjun049c", and the secondary apertures are 1, 7, etc. The syntax for the list of apertures also includes hyphen delimited ranges such as "8-10".

The assignment of arc calibration exposures to object exposures is generally done by selecting the nearest in time and interpolating. There are other options possible which are described under the task **refspectra**. The most general option is to define a table giving the object image name and the one or two arc spectra to be assigned to that object. That file is called an *arc assignment table* and it is one of the optional setup files which can used with **donessie**.

Because of the difficulties in taking comparison lamp exposures and the good stability of the instrument, there are two mutually exclusive methods for monitoring shifts in the dispersion zero point from the basic arc lamp spectra. One is to use some fibers to take a simultaneous arc spectrum while observing the program objects. The fibers are identified by aperture or beam numbers. The second method is to use *auxiliary line spectra*, such as mercury lines from the dome lights. These spectra are specified with an auxiliary shift arc list.

The first step in the processing is identifying the spectra in the images and defining apertures for extraction. An optional, though highly recommended, setup file is used. The file, called an *aperture identification file*, contains lines consisting of an aperture number, a beam number, and an object identification. These must be in the same order as the fibers in the image. The aperture number may be any unique number but it is recommended that the fiber number be used. The beam number is used to flag object, sky, arc, or other types of spectra. The default beam numbers used by the task are 0 for sky, 1 for object, and 2 for arc. The object identifications are optional but it is good practice to include them so that the data will contain the object information independent of other records. Figure 2 shows an example for the *blue* fibers from a board called M33Sch2.

Figure 2: Example Aperture Identification File

```
cl> type m33sch2
1 1 143
2 1 254
3 0 sky
4 1 121
5 2 arc
   .
   .
   .
44 1 s92
49 -1 Broken
45 1 156
46 2 arc
47 0 sky
48 1 phil2
```

Note the identification of the sky fibers with beam number 0, the object fibers with 1, and the arc fibers with 2. Also note that broken fiber 49 is actually between fibers 44 and 45. The broken fiber entries, given beam number -1, are optional but recommended to give the automatic spectrum finding operation the best chance to make the correct identifications. The identification file will vary for each plugboard setup. Additional information about the aperture identification file may be found in the description of the task **apfind**.

An alternative to using an aperture identification file is to give no file, the "" empty string, and to explicitly give a range of aperture numbers for the skys and possibly for the sky subtraction object list.

The final reduced spectra are recorded in a *multispec* format image. This is a special two or three dimensional image in which each line in a plane is a one dimensional spectrum with aperture, title, and dispersion information

stored in special image header keywords. The third dimension is created when using cleaning or variance weighting and contains the raw (uncleaned/unweighted) spectrum and variance spectrum (see **apextract** or **apsum** for further details). This type of spectral format is accepted by most of the one dimensional spectroscopy tools such as the plotting tasks **splot** and **specplot**. The special task **scopy** may be used to extract specific apertures or to change format to individual one dimensional images. The reduced multispec image has the same name as the original two dimensional image with an added `.ms` extension.

5. Package Parameters

The **nessie** package parameters set the standard I/O functions. Figure 3 shows the default package parameters.

Figure 3: Example Package Parameter Set for NESSIE

```
cl> epar nessie
                          I R A F
           Image Reduction and Analysis Facility
PACKAGE = imred
   TASK = nessie

(latitud=      31.958419722451) Observatory latitude (degrees)

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

(version= V3: July 1990)
```

The verbose parameter selects whether to print everything which goes into the log file on the terminal. It is useful for monitoring everything that the task does. The log and plot files are useful for keeping a record of the processing. A log file is highly recommended. The interactive graphics device is set by the environment variable STDGRAPH and the cursor input uses the CL parameter `gcur`.

6. Processing Parameters

Figure 4 shows the parameters and a typical setup for the processing task **donessie**. Appendix A reproduces the parameter descriptions from the task help. The list of objects and arcs can be @ files if desired. The aperture reference spectrum is usually the same as the flat field spectrum though it could be any exposure with enough signal to accurately define the positions and trace the spectra. The first list of arcs are the standard Th-Ar or HeNeAr comparison arc spectra (they must all be of the same type). The second list of arcs are the auxilary emission line exposures described previously, though they could also be any type of spectrum with at least one strong line.

Figure 4: Example Parameter Set for DONESSIE

```
cl> epar donessie

                          I R A F
           Image Reduction and Analysis Facility
PACKAGE = nessie
   TASK = donessie

objects =   nesjun106c  List of object spectra
(apref  =   nesjun110c) Aperture reference spectrum
(flat   =   nesjun110c) Flat field spectrum
(skyflat=             ) Sky flat spectrum (optional)
(arcs1  =   nesjun042c) List of arc spectra
(arcs2  =    nesjun111) List of shift arc spectra
(arcrepl=   arcreplace) Special aperture replacements
(arctabl=             ) Arc assignment table (optional)
```

```
(readnoi=           5.) Read out noise sigma (photons)
(gain   =           5.) Photon gain (photons/data number)
(dispaxi=            2) Dispersion axis (1=along lines, 2=along columns)
(fibers =           44) Number of fibers
(width  =           4.) Width of profiles (pixels)
(minsep =           5.) Minimum separation between fibers (pixels)
(maxsep =           7.) Maximum separation between fibers (pixels)
(apidtab=      m33sch2) Aperture identifications
(objaps =             ) Object apertures
(skyaps =             ) Sky apertures
(arcaps =             ) Arc apertures
(objbeam=          0,1) Object beam numbers
(skybeam=            0) Sky beam numbers
(arcbeam=            2) Arc beam numbers

(fitflat=          yes) Fit and ratio flat field spectrum?
(clean  =           no) Detect and replace bad pixels?
(dispcor=          yes) Dispersion correct spectra?
(savearc=          yes) Save simultaneous arc apertures?
(skysubt=          yes) Subtract sky?
(skyedit=          yes) Edit the sky spectra?
(savesky=          yes) Save sky spectra?
(splot  =           no) Plot the final spectrum?
(redo   =           no) Redo operations if previously done?
(update =          yes) Update spectra if cal data changes?
(batch  =           no) Extract objects in batch?
(listonl=           no) List steps but don't process?

(params =             ) Algorithm parameters
```

The arc replacement file and arc assignment table were described previously. Note that even if an arc assignment table is specified, *all arcs to be used must also appear in the arc lists* in order for the task to know the type of arc spectrum.

The detector read out noise and gain are used for cleaning and variance (optimal) extraction. The dispersion axis defines the wavelength direction of spectra in the image if not defined in the image header by the keyword DISPAXIS. The width and separation parameters define the dimensions (in pixels) of the spectra (fiber profile) across the dispersion. The width parameter primarily affects the centering. The maximum separation parameter is important if missing spectra from the aperture identification file are to be correctly skipped. The number of fibers can be left at the default and the task will try to account for missing fibers. However, this may lead to occasional incorrect identifications so it is recommended that only the true number of fibers be specified. The aperture identification file was described earlier.

The task needs to know which fibers are object, sky if sky subtraction is to be done, and simultaneous arcs if used. One could explicitly give the aperture numbers but the recommended way, provided an aperture identification file is prepared, is to select the apertures based on the beam numbers. If no list of aperture or beam numbers is given then all apertures or beam numbers are selected. Sky subtracted sky spectra are useful for evaluating the sky subtraction. Since only the spectra identified as objects are sky subtracted one can exclude fibers from the sky subtraction. For example, if the `objbeams` parameter is set to 1 then only those fibers with a beam of 1 will be sky subtracted. All other fibers will remain in the extracted spectra but will not be sky subtracted.

The next set of parameters select the processing steps and options. The flat fitting option allows fitting and removing the overall shape of the flat field spectra while preserving the pixel-to-pixel response corrections. This is useful for maintaining the approximate object count levels and not introducing the reciprocal of the flat field spectrum into the object spectra. The clean option invokes a profile fitting and deviant point rejection algorithm as well as a variance weighting of points in the aperture. These options require knowing the effective (i.e. accounting for any image combining) read out noise and gain. For a discussion of cleaning and variance weighted extraction see **apvariance** and **approfiles**.

The dispersion correction option selects whether to extract arc spectra, determine a dispersion function, assign them to the object spectra, and rebin the spectra to a linear (or log-linear) wavelength scale. If simultaneous arc fibers are defined there is an option to delete them from the final spectra when they are no longer needed.

The sky subtraction option selects whether to combine the sky fiber spectra and subtract this sky from the object fiber spectra. *Dispersion correction and sky subtraction are independent operations.* This means that if dispersion correction is not done then the sky subtraction will be done with respect to pixel coordinates. This might be desirable in some quick look cases though it is incorrect for final reductions.

The sky subtraction option has two additional options. The individual sky spectra may be examined and contaminated spectra deleted interactively before combining. This can be a useful feature in crowded regions. The final combined sky spectrum may be saved for later inspection in an image with the spectrum name prefixed by `sky`.

After a spectrum has been processed it is possible to examine the results interactively using the **splot** tasks. This option has a query which may be turned off if there are multiple spectra to be processed.

Generally once a spectrum has been processed it will not be reprocessed if specified as an input spectrum. However, changes to the underlying calibration data can cause such spectra to be reprocessed if the `update` flag is set. The changes which will cause an update are new reference apertures, new flat fields, and a new arc reference. If all input spectra are to be processed regardless of previous processing the `redo` flag may be used. Note that reprocessing clobbers the previously processed output spectra.

The `batch` processing option allows object spectra to be processed as a background or batch job. This will only occur if sky spectra editing and **splot** review (interactive operations) are turned off, either when the task is run or by responding with "NO" to the queries during processing.

The `listonly` option prints a summary of the processing steps which will be performed on the input spectra without actually doing anything. This is useful for verifying which spectra will be affected if the input list contains previously processed spectra. The listing does not include any arc spectra which may be extracted to dispersion calibrate an object spectrum.

The last parameter (excluding the task mode parameter) points to another parameter set for the algorithm parameters. The default parameter set is called `params`. The algorithm parameters are discussed further in the next section.

7. Algorithms and Algorithm Parameters

This section summarizes the various algorithms used by the **donessie** task and the parameters which control and modify the algorithms. The algorithm parameters available to the user are collected in the parameter set **params**. These parameters are taken from the various general purpose tasks used by the **donessie** processing task. Additional information about these parameters and algorithms may be found in Appendix A, the help for **donessie**, and the actual task executed. These tasks are identified below. The aim of this parameter set organization is to collect all the algorithm parameters in one place separate from the processing parameters and include only those which are relevant for *Nessie* data. The parameter values can be changed from the defaults by using the parameter editor,

```
cl> epar params
```

or simple typing `params`. The parameter editor can also be entered when editing the **donessie** parameters by typing `:e params` or simply `:e` if positioned at the `params` parameter. Figure 5 shows the default parameter values.

Figure 5: Default Parameter Set for PARAMS

```
cl> params
                                I R A F
                  Image Reduction and Analysis Facility
PACKAGE = nessie
   TASK = params

(line   =        INDEF) Default dispersion line
(nsum   =           10) Number of dispersion lines to sum
(order  =   decreasing) Order of apertures

                        -- DEFAULT APERTURE LIMITS --
(lower  =          -3.) Lower aperture limit relative to center
(upper  =           3.) Upper aperture limit relative to center

                        -- AUTOMATIC APERTURE RESIZING PARAMETERS --
```

```
(ylevel =         0.05) Fraction of peak or intensity for resizing
(peak   =          yes) Is ylevel a fraction of the peak?
(bkg    =          yes) Subtract background for resizing?
(avglimi=           no) Average limits over all apertures?

                         -- 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

                         -- APERTURE EXTRACTION PARAMETERS --
(weights=         none) Extraction weights (none|variance)
(lsigma =           3.) Lower rejection threshold
(usigma =           3.) Upper rejection threshold
(nsubaps=            1) Number of subapertures

                         -- FLAT FIELD FUNCTION FITTING PARAMETERS --
(f_inter=          yes) Fit flat field interactively?
(f_funct=      spline3) Fitting function
(f_order=           20) Fitting function order

                         -- ARC DISPERSION FUNCTION PARAMETERS --
(coordli= onedstds$idhenear.dat) Line list
(match  =          10.) Line list matching limit in Angstroms
(fwidth =           4.) Arc line widths in pixels
(i_funct=    chebyshev) Coordinate function
(i_order=            3) Order of dispersion function
(i_niter=            2) Rejection iterations
(i_low  =           3.) Lower rejection sigma
(i_high =           3.) Upper rejection sigma
(refit  =          yes) Refit coordinate function when reidentifying?
(addfeat=           no) Add features when reidentifying?

                         -- AUTOMATIC ARC ASSIGNMENT PARAMETERS --
(select =       interp) Selection method for reference spectra
(sort   =     utmiddle) Sort key
(time   =          yes) Is sort key a time?
(timewra=          17.) Time wrap point for time sorting

                         -- DISPERSION CORRECTION PARAMETERS --
(interpo=        poly5) Interpolation type
(log    =           no) Logarithmic wavelength scale?
(flux   =          yes) Conserve flux?

                         -- SKY SUBTRACTION PARAMETERS --
(combine=    avsigclip) Sky combine option:
                            none, average, median, minreject,
                            maxreject, minmaxrej, sigclip,
                            avsigclip
(scale  =           no) Scale skys by the mode?
```

7.1. Extraction

The identification of the spectra in the two dimensional images and their extraction to one dimensional spectra in multispec format is accomplished using the **apextract** tasks. The first parameters through `nsubaps` control the extractions.

The dispersion line is that used for finding the spectra, for plotting in the aperture editor, and as the starting point for tracing. The default value of `INDEF` selects the middle of the image. The aperture finding, adjusting, editing, and tracing operations also allow summing a number of dispersion lines to improve the signal. The number of lines is set by the `nsum` parameter.

The order parameter defines whether the order of the aperture identifications in the aperture identification file (or the default sequential numbers if no file is used) is in the same sense as the image coordinates (increasing) or the opposite sense (decreasing). If the aperture identifications turn out to be opposite to what is desired when viewed in the aperture editing graph then simply change this parameter.

When the apertures for the reference spectrum are first identified and defined, the edges of the apertures relative to the centers of the spectra (measured in pixels) are set by the parameters `lower` and `upper` (see **apfind**). The initial apertures are the same for all spectra but they can be automatically resized. The automatic resizing can be set at a fraction of the peak pixel or as an absolute data level with or without a background subtracted (see **apresize**). The default is to resize the apertures at 5% of the peak pixel.

The next set of parameters control the tracing and function fitting of the aperture reference positions along the dispersion direction. The position of a spectrum across the dispersion is determined by the centering algorithm (see **center1d**) at a series of evenly spaced steps, given by the parameter `t_step`, along the dispersion. The step size should be fine enough to follow position changes but it is not necessary to measure every point. The fitted points may jump around a little bit due to noise and cosmic rays even when summing a number of lines. Thus, a smooth function is fit. The function type, order, and iterative rejection of deviant points is controlled by the other trace parameters. For more discussion consult the help pages for **aptrace** and **icfit**. The default is to fit a cubic spline of two pieces with a single iteration of 3 sigma rejection.

The actual extraction of the spectra by summing across the aperture at each point along the dispersion is controlled by the next set of parameters. The default extraction simply sums the pixels using partial pixels at the ends. The options allow selection of a weighted sum based on a Poisson variance model using the `readnoise` and `gain` detector parameters. Note that if the `clean` option is selected the variance weighted extraction is used regardless of the `weights` parameter. The sigma threshold for cleaning are also set in the **params** parameters. For more on the variance weighted extraction and cleaning see **apvariance** and **approfiles** as well as **apsum**.

The last parameter, `nsubaps`, is used only in special cases when it is desired to subdivide the fiber profiles into subapertures prior to dispersion correction. After dispersion correction the subapertures are then added together. The purpose of this is to correct for wavelength shifts across a fiber.

7.2. Flat Field and Fiber Throughput Corrections

Flat field corrections may be made during the basic CCD processing; i.e. direct division by the two dimensional flat field observation. In that case do not specify a flat field spectrum; use the null string "". The **donessie** task provides an alternative flat field response correction based on division of the extracted spectra by the extracted flat field spectra. A discussion of the theory and merits of flat fielding directly verses using the extracted spectra will not be made here. The **donessie** flat fielding algorithm is the *recommended* method for flat fielding since it works well and is not subject to the many problems involved in two dimensional flat fielding.

In addition to correcting for pixel-to-pixel response the flat field step also corrects for differences in the fiber throughput. Thus, even if the pixel-to-pixel flat field corrections have been made in some other way it is desirable to use a sky or dome flat observation for determining a fiber throughput correction.

The first step is extraction of the flat field spectrum, if specified, using the reference apertures. Only one flat field is allowed so if multiple flat fields are required the data must be reduced in groups. After extraction one or more corrections are applied. If the `fitflat` option is selected the extracted flat field spectra are averaged and a smooth function is fit. The default fitting function and order are given by the parameters `f_function` and `f_order`. If the parameter `f_interactive` is `yes` then the fitting is done interactively using the **icfit** commands. The task used for the fitting is **fit1d**.

The fitted function is divided into the individual flat field spectra to remove the basic shape of the spectrum while maintaining the relative individual pixel responses and any fiber to fiber differences. This step avoids introducing the flat field spectrum shape into the object spectra and closely preserves the object counts.

If a sky flat field spectrum is available (an observation of blank sky usually at twilight) it is extracted. If no

flat field is used the average through each fiber is computed and this becomes the response normalization function. Note that a dome flat may be used in place of a sky in the sky flat field parameter for producing throughput only corrections. If a flat field is specified then each sky spectrum is divided by the appropriate flat field spectrum. The total counts through each fiber are multiplied into the flat field spectrum thus making the sky throughput of each fiber the same. This correction is important if the illumination of the fibers differs between the flat field source and the sky. Since only the total counts are required the sky or dome flat field spectra need not be particularly strong though care must be taken to avoid objects.

The final step is to normalize the flat field spectra by the mean counts of all the fibers. This normalization step is simply to preserve the average counts of the extracted object and arc spectra after division by the response spectra.

These flat field response steps and algorithm are available as a separate task called **msresponse**.

7.3. Dispersion Correction

Dispersion corrections are applied to the extracted spectra if the `dispcor` processing parameter is set. This is a complicated process which the **donessie** task tries to simplify for you. There are three basic steps involved; determining the dispersion functions relating pixel position to wavelength, assigning the appropriate dispersion function to a particular observation, and resampling the spectra to evenly spaced pixels in wavelength.

The comparison arc spectra are used to define dispersion functions for the fibers using the tasks **identify** and **reidentify**. The interactive **identify** task is only used on the central fiber of the first comparison lamp spectrum (and possibly the central fiber of the first auxiliary line spectrum) to define the basic reference dispersion solution from which all other fibers and comparison lamp spectra are automatically derived.

The set of arc dispersion function parameters are from **identify** and **reidentify**. The parameters define a line list for use in automatically assigning wavelengths to arc lines, a parameter controlling the width of the centering window (which should match the line widths, twice the full width at half maximum), the dispersion function type and order, parameters to exclude bad lines from function fits, and parameters defining whether to refit the dispersion function, as opposed to simply determining a zero point shift, and the addition of new lines from the line list when reidentifying additional arc spectra. The defaults should generally be adequate and the dispersion function fitting parameters may be altered interactively. One should consult the help for the two tasks for additional details of these parameters and the operation of **identify**.

When using auxiliary line spectra for monitoring zero point shifts one of the fiber spectra is plotted interactively by **identify** with the reference dispersion function for the appropriate fiber. The user marks one or more lines which will be used to compute zero point wavelength shifts in the dispersion functions automatically.

The assignment of arc spectra for a particular object spectrum can be a confusing operation. It greatly depends on the number and timing of the arcs. Because the *Nessie* arcs require manually moving fibers and since the instrument is quite stable, the fundamental arcs should be taken infrequently, usually only at the beginning and end of the night. There are two methods, which are mutually exclusive, for monitoring and correcting the basic dispersion functions for zero point wavelength shifts. One is to devote some of the fibers to recording a simultaneous comparison lamp spectrum. The zero point shifts in those fibers are then interpolated to the other fibers. The second method is with easily taken auxiliary arcs (or night sky line) spectra. These provide independent shifts for each fiber and may be based on a different set of lines than the comparison lamp lines.

If there are no simultaneous or auxiliary arcs then the basic comparison lamp dispersion functions are assigned fiber by fiber to a particular object exposure. The assignment of arcs is done either explicitly with an arc assignment table (parameter `arctable`) or based on a header parameter. The task used is **refspectra** and the user should consult this task if the default behavior is not what is desired. The default is to interpolate linearly between the nearest arcs based on the universal time (corrected to the middle of the exposure).

If an auxiliary arc list is specified (parameter `arcs2`) then the same assignment operation is used based on the set of auxiliary arcs. Note that the basic comparison lamp arcs are not used in this case except for defining the reference dispersion functions.

If simultaneous arcs are used they are reidentified for a shift relative to the comparison lamp spectra selected by **refspectra**. A linear function of aperture position is determined for the shifts and applied to the dispersion functions from the assigned calibration arcs for the non-arc spectra. Note that if there are two comparison lamp spectra (before and after the object exposure) then there will be two shifts applied to two dispersion functions which are then combined using the weights based on the header parameters (usually the observation time).

The last step of dispersion correction (interpolating the spectrum to evenly spaced pixels in wavelength) is relatively straightforward. The algorithm parameters allow selecting the interpolation function type, whether to conserve flux per pixel by integrating across the extent of the final pixel, and whether to linearize to equal linear or

logarithmic intervals. The latter may be appropriate for radial velocity studies. The default is to use a fifth order polynomial for interpolation, to conserve flux, and to not use logarithmic wavelength bins. These parameters are described fully in the help for the task **msdispcor** which performs the correction.

7.4. Sky Subtraction

Sky subtraction is selected with the `skysubtract` processing option. There are three steps involved, selecting the sky spectra, combining them into a master sky spectrum, and subtracting it from the specified object fibers. Selection of the sky spectra begins with finding those spectra with the specified aperture and beam numbers. If the combining algorithm is "`none`" then it is assumed that there are exactly the same number of sky fibers as object fibers and in the same order. The subset of sky apertures is subtracted from the subset of object fibers matched in order. This may be useful occasionally though there is a big penalty to be paid in noise.

More commonly the sky fibers are combined into a single master sky spectrum which is then subtracted from each object spectrum. If the `skyedit` option is selected the sky spectra are plotted using the task **specplot**. By default they are superposed to allow identifying spectra with unusually high signal due to object contamination. To eliminate a sky spectrum from consideration point at it with the cursor and type 'd'. There is a new key stroke in **specplot**, 'e', which undeletes the last deleted spectrum. This allows recovery of incorrect or accidental deletions.

The sky combining algorithm parameters define how the individual sky fiber spectra, after interactive editing, are combined before subtraction from the object fibers. The goals of combining are to reduce noise, eliminate cosmic-rays, and eliminate fibers with inadvertent objects. The common methods for doing this are a median and a special sigma clipping algorithm. The maximum rejection option is also reasonable. The second parameter determines whether the individual skys are first scaled to a common mode. The scaling should be used if the throughput is uncertain, but in that case you probably did the wrong thing in the throughput correction. If the sky subtraction is done interactively, i.e. with the `skyedit` options selected, then after selecting the spectra to be combined a query is made for the combining algorithm. This allows modifying the default algorithm based on the number of sky spectra selected.

The combined sky spectrum is subtracted from only those spectra specified by the object aperture and beam numbers. Other spectra, such as comparison arc spectra, are retained unchanged. One may include the sky spectra as object spectra to produce residual sky spectra for analysis. The sky spectra when not combining or the combined master sky spectra may be saved if the `saveskys` parameter is set. The saved sky is given the name of the object spectrum with the prefix "sky". The ".ms" extension is eliminated if a combined spectrum is used since it is a simple one dimensional spectrum.

8. References

8.1. IRAF Introductory References

Work is underway on a new introductory guide to IRAF. Currently, the work below is the primary introduction.

P. Shames and D. Tody, *A User's Introduction to the IRAF Command Language*, Central Computer Services, NOAO, 1986.

8.2. CCD Reductions

F. Valdes, *The IRAF CCD Reduction Package -- CCDRED*, Central Computer Services, NOAO, 1987.

F. Valdes, *User's Guide to the CCDRED Package*, Central Computer Services, NOAO, 1988. Also on-line as `help ccdred.guide`.

P. Massey, *A User's Guide to CCD Reductions with IRAF*, Central Computer Services, NOAO, 1989.

8.3. Aperture Extraction Package

F. Valdes, *The IRAF APEXTRACT Package*, Central Computer Services, NOAO, 1987 (out-of-date).

8.4. Task Help References

Each task in the **nessie** package and tasks used by **donessie** have help pages describing the parameters and task in some detail. To get on-line help type

`cl> help` *taskname*

The output of this command can be piped to **lprint** to make a printed copy.

```
     apall - Extract 1D spectra (all parameters in one task)
  apdefault - Set the default aperture parameters
    apedit - Edit apertures interactively
    apfind - Automatically find spectra and define apertures
 aprecenter - Recenter apertures
  apresize - Resize apertures
    apsum - Extract 1D spectra
   aptrace - Trace positions of spectra

     bplot - Batch plot of spectra with SPLOT
  continuum - Fit and normalize the continuum of multispec spectra
   identify - Identify arc lines and determine a dispersion function
  msdispcor - Dispersion correct multispec format spectra
 msresponse - Create fiber response spectra from flat field and sky spectra
 refspectra - Assign reference spectra to observations
 reidentify - Reidentify arc lines and determine new dispersion functions
 sapertures - Set aperture information in spectra
     scopy - Copy spectra including aperture selection and format changes
    skysub - Sky subtract extracted fiber spectra
     slist - List spectrum headers
  specplot - Stack and plot multiple spectra
    splot - Plot and analyze spectra

  donessie - Process NESSIE spectra
    demos - Demonstrations and tests

       Additional help topics

  apextract - Package parameters and general description of package
 approfiles - Profile determination algorithms
 apvariance - Extractions, variance weighting, cleaning, and noise model
```

9. Appendix A: DONESSIE Parameters

objects
: List of object spectra to be processed. Previously processed spectra are ignored unless the *redo* flag is set or the *update* flag is set and dependent calibration data has changed. Extracted spectra are ignored.

apref = ""
: Aperture reference spectrum. This spectrum is used to define the basic extraction apertures and is typically a flat field spectrum.

flat = "" (optional)
: Flat field spectrum. If specified the one dimensional flat field spectra are extracted and used to make flat field and throughput corrections to all extracted spectra. If not specified it is assumed that the images have been flat fielded separately.

skyflat = "" (optional)
: Sky flat field spectrum. If specified the total flux through each fiber is used to correct for fiber throughput. Note a dome flat spectrum may specified for computing a throughput correction only.

arcs1 = "" (at least one if dispersion correcting)
: List of primary arc spectra. These spectra are used to define the dispersion functions for each fiber apart from a possible zero point correction made with secondary shift spectra or arc calibration fibers in the object spectra. One fiber from the first spectrum is used to mark lines and set the dispersion function interactively and dispersion functions for all other fibers and arc spectra are derived from it.

arcs2 = "" (optional)
: List of optional shift arc spectra. Features in these secondary observations are used to supply a wavelength zero point shift through the observing sequence. One type of observation is dome lamps containing characteristic emission lines.

arcreplace = "" (optional)
: Special aperture replacement file. A characteristic of Nessie spectra is that it requires two exposures to illuminate all fibers with an arc calibration. The aperture replacement file assigns fibers from the second exposure to the first exposure. Only the first exposures are specified in the *arcs1* list. The file contains lines with the first exposure image name, the second exposure image name, and a list of apertures from the second exposure to be used instead of those in the first exposure.

arctable = "" (optional) (refspectra)
Table defining which arc spectra are to be assigned to which object spectra (see **refspectra**). If not specified an assignment based on a header parameter, *params.sort*, such as the observation time is made.

readnoise = 0. (apsum)
Read out noise in photons. This parameter defines the minimum noise sigma. It is defined in terms of photons (or electrons) and scales to the data values through the gain parameter. A image header keyword (case insensitive) may be specified to get the value from the image.

gain = 1. (apsum)
Detector gain or conversion factor between photons/electrons and data values. It is specified as the number of photons per data value. A image header keyword (case insensitive) may be specified to get the value from the image.

dispaxis = 2 (apextract)
Default dispersion axis. The dispersion axis is 1 for dispersion running along image lines and 2 for dispersion running along image columns. If the image header parameter DISPAXIS is defined it has precedence over this parameter.

fibers = 49 (apfind)
Number of fibers. This number is used during the automatic definition of the apertures from the aperture reference spectrum. It is best if this reflects the actual number of fibers excluding broken fibers to avoid possible misidentifications. However, the automatic method may eliminate broken fibers and a later interactive review may also be used to adjust the identification.

width = 5. (apedit)
Approximate full width of fiber profiles. This parameter is used for the profile centering algorithm.

minsep = 5. (apfind)
Minimum separation between fibers. Weaker spectra or noise within this distance of a stronger spectrum are rejected.

maxsep = 7. (apfind)
Maximum separation between adjacent fibers. This parameter is used to identify missing fibers. If two adjacent spectra exceed this separation then it is assumed that a fiber is missing and the aperture identification assignments will be adjusted accordingly.

apidtable = "" (apfind)
Aperture identification table containing the fiber number, beam number defining object, sky, and arc fibers, and a spectrum title. Typically a user prepares this for each plugboard. Broken fibers should be included in this list since they will automatically be excluded.

objaps = "" , skyaps = "" , arcaps
List of object, sky, and arc aperture numbers. These are used to identify arc apertures for wavelength calibration and object and sky apertures for sky subtraction. Note sky apertures may be identified as both object and sky if one wants to subtract the mean sky from the individual sky spectra. Typically the different spectrum types are identified by their beam numbers and the default values are left unchanged.

objbeams = "0,1" , skybeams = "0" , arcbeams
List of object, sky, and arc beam numbers. The convention is that sky fibers are given a beam number of 0, object fibers a beam number of 1, and arc fibers a beam number of 2. The beam numbers are typically set in the *apidtable*. Note it is valid to identify sky fibers as both object and sky.

fitflat = yes
Fit the composite flat field spectrum by a smooth function and divide each flat field spectrum by this function? This operation removes the spectral signature of the flat field lamp from the sensitivity correction and avoids modifying the object fluxes.

clean = yes (apsum)
Detect and correct for bad pixels during extraction? This is the same as the clean option in the **apextract** package. If yes this also implies variance weighted extraction.

dispcor = yes
Dispersion correct spectra (rebin to uniform wavelength coordinates)?

savearcs = yes
Save any simultaneous arc apertures? If no then the arc apertures will be deleted after use.

skysubtract = yes
Subtract sky from the object spectra? If yes the sky spectra are combined and subtracted from the object spectra as defined by the object and sky identification parameters.

skyedit = yes
Overplot all the sky spectra and allow contaminated sky apertures to be deleted?

saveskys = yes
Save the sky spectra? If no then the sky spectra will be deleted after sky subtraction is completed.

splot = no
Plot the final spectra with the task **splot**?

redo = no
Redo operations previously done? If no then previously processed spectra in the objects list will not be processed (unless they need to be updated).

update = yes
Update processing of previously processed spectra if aperture, flat field, or dispersion reference definitions are changed?

batch = yes
Process spectra as a background or batch job provided there are no interactive options (*skyedit* and *splot*) selected.

listonly = no
List spectra to be processed but don't process?

params = "" (pset)
Name of parameter set containing additional processing parameters. The default is parameter set **params**. The parameter set may be examined and modified in the usual ways (typically with **eparam** or ":e params" from the parameter editor). The parameters are described below.

-- GENERAL PARAMETERS --

line = INDEF, nsum = 1
The dispersion line (line or column perpendicular to the dispersion axis) and number of adjacent lines (half before and half after unless at the end of the image) used in finding, recentering, resizing, editing, and tracing operations. A line of INDEF selects the middle of the image along the dispersion axis.

order = "decreasing" (apfind)
When assigning aperture identifications order the spectra "increasing" or "decreasing" with increasing pixel position (left-to-right or right-to-left in a cross-section plot of the image).

-- DEFAULT APERTURE LIMITS --

lower = -3., upper = 3. (apdefault)
Default lower and upper aperture limits relative to the aperture center. These limits are used when the apertures are first found.

-- AUTOMATIC APERTURE RESIZING PARAMETERS --

ylevel = 0.05 (apresize)
Data level at which to set aperture limits during automatic resizing. It is a fraction of the peak or an actual data level depending on the parameter *peak.* It may be relative to a local background or to zero depending on the parameter *bkg.*

peak = yes (apresize)
Is the data level specified by *ylevel* a fraction of the peak?

bkg = yes (apresize)
Subtract a simple background when interpreting the **ylevel** parameter. The background is a slope connecting the first minima away from the aperture center.

avglimits = no (apresize)
Apply the average lower and upper aperture limits to all apertures.

-- TRACE PARAMETERS --

t_step = 10 (aptrace)
Step along the dispersion axis between determination of the spectrum positions. Note the *nsum* parameter is also used to enhance the signal-to-noise at each step.

t_function = "spline3" , t_order = 2 (aptrace)
Default trace fitting function and order. The fitting function types are "chebyshev" polynomial, "legendre" polynomial, "spline1" linear spline, and "spline3" cubic spline. The order refers to the number of terms in the polynomial functions or the number of spline pieces in the spline functions.

t_niterate = 1, t_low = 3., t_high = 3.
Default number of rejection iterations and rejection sigma thresholds.

-- APERTURE EXTRACTION PARAMETERS --

weights = "none" (apsum)

Type of extraction weighting. Note that if the *clean* parameter is set then the weights used are "variance" regardless of the weights specified by this parameter. The choices are:

"none"

The pixels are summed without weights except for partial pixels at the ends.

"variance"

The extraction is weighted by the variance based on the data values and a poisson/ccd model using the *gain* and *readnoise* parameters.

lsigma = 3., usigma = 3. (apsum)

Lower and upper rejection thresholds, given as a number of times the estimated sigma of a pixel, for cleaning.

nsubaps = 1 (apsum)

During extraction it is possible to equally divide the apertures into this number of subapertures.

-- FLAT FIELD FUNCTION FITTING PARAMETERS --

f_interactive = yes (fit1d)

Fit the composite one dimensional flat field spectrum interactively? This is used if *fitflat* is set and a two dimensional flat field spectrum is specified.

f_function = "spline3" , f_order = 20 (fit1d)

Function and order used to fit the composite one dimensional flat field spectrum. The functions are "legendre", "chebyshev", "spline1", and "spline3". The spline functions are linear and cubic splines with the order specifying the number of pieces.

-- ARC DISPERSION FUNCTION PARAMETERS --

coordlist = "onedstds$idhenear.dat" (identify)

Arc line list consisting of an ordered list of wavelengths. Some standard line lists are available in the directory "onedstds$".

match = 10. (identify)

The maximum difference for a match between the dispersion function prediction value and a wavelength in the coordinate list.

fwidth = 4. (identify)

Approximate full base width (in pixels) of arc lines.

i_function = "spline3" , i_order = 1 (identify)

The default function and order to be fit to the arc wavelengths as a function of the pixel coordinate. The functions choices are "chebyshev", "legendre", "spline1", or "spline3".

i_niterate = 0, i_low = 3.0, i_high = 3.0

Number of rejection iterations and sigma thresholds for rejecting arc lines from the dispersion function fits.

refit = yes (reidentify)

Refit the dispersion function? If yes and there is more than 1 line and a dispersion function was defined in the arc reference then a new dispersion function of the same type as in the reference image is fit using the new pixel positions. Otherwise only a zero point shift is determined for the revised fitted coordinates without changing the form of the dispersion function.

addfeatures = no (reidentify)

Add new features from a line list during each reidentification? This option can be used to compensate for lost features from the reference solution. Care should be exercised that misidentified features are not introduced.

-- AUTOMATIC ARC ASSIGNMENT PARAMETERS --

select = "interp" (refspectra)

Selection method for assigning wavelength calibration spectra. Note that an arc assignment table may be used to override the selection method and explicitly assign arc spectra to object spectra. The automatic selection methods are:

average

Average two reference spectra without regard to any sort parameter. If only one reference spectrum is specified then it is assigned with a warning. If more than two reference spectra are specified then only the first two are used and a warning is given. This option is used to assign two reference spectra, with equal weights, independent of any sorting parameter.

following

Select the nearest following spectrum in the reference list based on the sorting parameter. If there is no following spectrum use the nearest preceding spectrum.

interp

Interpolate between the preceding and following spectra in the reference list based on the sorting parameter. If there is no preceding and following spectrum use the nearest spectrum. The interpolation is weighted by the relative distances of the sorting parameter.

match
Match each input spectrum with the reference spectrum list in order. This overrides the reference aperture check.

nearest
Select the nearest spectrum in the reference list based on the sorting parameter.

preceding
Select the nearest preceding spectrum in the reference list based on the sorting parameter. If there is no preceding spectrum use the nearest following spectrum.

sort = "utmiddle" (refspectra)
Image header keyword to be used as the sorting parameter for selection based on order. The header parameter must be numeric but otherwise may be anything. Common sorting parameters are times or positions.

time = yes (refspectra)
Is the sorting parameter a time?

timewrap = 17. (refspectra)
The time origin for sorting time values. This time should precede the first observation and follow the last observation in a 24 hour cycle.

-- DISPERSION CORRECTION PARAMETERS --

interpolation = "poly5" (msdispcor)
Wavelength dispersion correction interpolation type. The interpolation types are "linear" for linear, "poly3" for cubic polynomial, "poly5", for quintic polynomial, and "spline3" for cubic spline.

log = no (msdispcor)
Use linear logarithmic wavelength coordinates? Linear logarithmic wavelength coordinates have wavelength intervals which are constant in the logarithm of the wavelength.

flux = yes (msdispcor)
Conserve the total flux during interpolation? If *no* the output spectrum is interpolated from the input spectrum at each output wavelength coordinate. If *yes* the input spectrum is integrated over the extent of each output pixel. This is slower than simple interpolation.

-- SKY SUBTRACTION PARAMETERS --

combine = "avsigclip" (imcombine)
Type of sky spectra combining algorithm. The choices are:

```
     none - Don't combine but pair the objects and sky for subtraction.

      sum - Sum the input images
  average - Average the input images
   median - Median the input images
minreject - Reject the minimum value at each pixel
maxreject - Reject the maximum value at each pixel
minmaxrej - Reject the min and max values at each pixel
threshold - Reject pixels above and below thresholds
  sigclip - Apply a sigma clipping algorithm to each pixel
avsigclip - Apply a sigma clipping algorithm to each pixel
```

scale = no (imcombine)
Scale the sky spectra by the mode? This should not be necessary if the flat field / sky flat / throughput corrections have been properly made.