Cooking with DAOSPEC

by E. Pancino and P. B. Stetson

Version 1.0
4 may 2004

Abstract

DAOSPEC is a Fortran program written by Peter B. Stetson at the Dominion Astrophysical Observatory (DAO) of the Herzberg Institute of Astrophysics, National Reasearch Council, in Victoria, Canada. DAOSPEC automatically finds absorption lines in stellar spectra, fits the continuum, measures equivalent widths and identifies lines with the help of a laboratory linelist, and also provides a radial velocity estimate. DAOSPEC is designed to work on extracted, one-dimensional spectra that have been binned linearly in wavelength, either in FITS or in IRAF (.imh, .pix) format. This cookbook describes how to obtain, install, configure and use DAOSPEC.

NOTE: Please be always certain to have the latest version of the original program, by contacting Peter B. Stetson at:
`<Peter.Stetson@nrc-cnr.gc.ca>`.
Additional information can be obtained at:
`http://cadcwww.hia.nrc.ca/stetson/daospec`
`http://stars.bo.astro.it/~GC/personal/epancino/projects/daospec.html`

Contents

Chapter 1

Getting Started

1.1 System and Software requirements

DAOSPEC has been succesfully installed and tested on: *(1)* various PCs with Linux RedHat 9.0 or Fedora 1.0 and their fortran g77 compiler and *(2)* Sun machines running different Solaris versions and their standard f77 fortran compilers.

In order for DAOSPEC to compile properly you need:

- To be able to display the pretty pictures on the monitor, you will need supermongo (`http://www.astro.princeton.edu/~rhl/sm/`) and, in particular, the following libraries: `libplotsub.a`, `libdevices.a` and `libutils.a`. Since supermongo is not a free program, you may ask us for a version of DAOSPEC that does not include the graphical interface and therefore does not need the supermongo libraries to compile.

- To work with FITS files, you will need Pence's `cfitsio` library (`libcfitsio.a`) that can be obtained at: `http://heasarc.gsfc.nasa.gov/docs/software/fitsio/fitsio.html`

- To work with IRAF[1], you will need the following `IMFORT` libraries: `libimfort.a`, `libsys.a`, `libvops.a` and `libos.a`; they can be obtained at: (`http://iraf.noao.edu`)

[1]IRAF is distributed by the National Optical Astronomy Observatories, which is operated by the association of Universities for Research in Astronomy, Inc., under contract with the National Science Foundation.

1.2 Obtaining DAOSPEC

You can find some information at the following web pages:
`http://cadcwww.hia.nrc.ca/stetson/daospec`
`http://stars.bo.astro.it/~GC/personal/epancino/projects/daospec.html`
The latest version of the code can be obtained by contacting Peter Stetson (`<Peter.Stetson@nrc-cnr.gc.ca>`). You will be asked to sign an agreement form and then receive a gzipped tarfile containing the following files:

- `daospec.f` — The source code of DAOSPEC
- `iosubs.f` — Routines for pretty input and output
- `mathsubs.f` — Mathematical routines
- `lnxsubs.f` — Routines specific to Linux machines
- `unxsubs.f` — Routines specific to Unix machines
- `bothsubs.f` — Image I/O routines for FITS and IRAF
- `irasubs.f` — Image I/O routines for IRAF but not FITS
- `fitsubs.f` — Image I/O routines for FITS but not IRAF
- `Makefile.unix` — The makefile to compile DAOSPEC on a Sun
- `Makefile.linux` — The makefile to compile DAOSPEC on Linux
- `daospec.ps` — This cookbook
- `test.fits` — A test spectrum[2] to check your installation
- `daospec.opt` — A test configuration file
- `laboratory.dat` — A test laboratory linelist
- `testC.fits` — A sample output continuum file
- `testR.fits` — A sample output residuals file
- `test.daospec` — A sample DAOSPEC results file

[2] This is a small portion of the Sun spectrum, taken with UVES@VLT with R~100,000. The whole spectrum can be downloaded at:
`http://www.eso.org/observing/dfo/quality/UVES/pipeline/solar_spectrum.html`

1.3 Installing DAOSPEC

The following makefiles are provided: `Makefile.unix` and `Makefile.linux`. You must first edit the appropriate Makefile, adding the appropriate library paths. If you have FITS images but don't have the IRAF libraries, replace bothsubs with fitsubs everywhere; if you have IRAF images and don't have the FITS libraries, replace bothsubs with irasubs. Then rename the makefile it and compile DAOSPEC as follows:

```
cp Makefile.<system> Makefile
make daospec
```

DAOSPEC is dimensioned for spectra of up to 1,200,000 pixels . This can cause a `segmentation fault` error message and crash DAOSPEC if your machine does not have enough resources. First try to `unlimit` your resources (`ulimit` on Linux) and if that does not work, you should change the `MXSPEC` parameter in `daospec.f` and make it smaller than 1,200,000. Then you need to recompile the code

```
make daospec
```

1.4 Test your installation

Once DAOSPEC has been successfully installed, you can test your installation by using the test image `test.fits` with its laboratory linelist `laboratory.dat` and its configuration file `daospec.opt` (see next chapter to see more details on running DAOSPEC). First, make copies of the three sample output files provided:

```
cp testC.fits safe.testC.fits
cp testR.fits safe.testC.fits
cp test.daospec safe.test.daospec
```

then, run daospec by simply typing[3]

```
daospec
```

[3] If your DAOSPEC `PATH` is not set, you may need to type it in entirely, i.e. `/usr/local/bin/daospec`, or you may want to copy the executable in your working directory and then type `./daospec`.

at the `OPT>` prompt type return, and then type return to all the following questions. You will see a display appear, showing the progress of continuum fitting and line subtraction procedures (see Figures 2.4, 2.5, 2.6 and 2.7). At the end of the process, three files will be created: `testC.fits`, `testR.fits` and `test.daospec`. If no error message appears and if the three produced files are identical to the ones that were provided in the DAOSPEC package, your installation has been successful, congratulations!

1.5 Problem Reports and Suggestions

For any problem or suggestion you may have, please feel free to contact Peter Stetson `<Peter.Stetson@nrc-cnr.gc.ca>` (for specific technical software questions) or Elena Pancino `<elena.pancino@bo.astro.it>` (for more general questions on the use and testing of DAOSPEC), but please always remember to provide the exact copy of the program that you are using, along with the fits or IRAF files that produced the problem, and all the related configuration files.

Your feedback will be much appreciated and will help us in producing better documentation, webpages and a better version of DAOSPEC. Thanks!

1.6 Release Policy and Acknowledgements

DAOSPEC has been written for a specific purpose, i.e., to measure equivalent width of absorption lines by gaussian fitting on line-crowded spectra. It has been designed to work optimally on high resolution, high signal-to-noise spectra. DAOspec is publicly available for free, in the hope that it can be useful. The author declines any responsibility. Please be sure to use always the latest version of the original code. You are free to modify the code to suit your personal needs, but do it with responsibility and please do not distribute the code (modified or otherwise) to third parties; distributing the code without permission will make the lawyers in Ottawa very angry. If you find any bug or have any suggestion, please contact us.

We are now working on a scientific paper that will describe DAOSPEC in more detail. If you use DAOspec to produce and publish any scientific results, please always refer to the paper. For the moment you should simply use the reference *"P. B. Stetson & E. Pancino 2004, in preparation"*. We would also appreciate if you could add a footnote to DAOSPEC in your paper, containing the following text: *"DAOSPEC has been written by P. B. Stetson for the Dominion Astrophysical Observatory of the Herzberg Institute* *of Astrophysics, National Research Council, Canada."*.

Acknowledgments. We would like to thank the early users and testers of DAOSPEC, along with people that gave us useful suggestions and advice: Luca Pasquini, Manuela Zoccali, Vanessa Hill, Patrick Francois, Francesca Primas, Eugenio Carretta and Livia Origlia.

Chapter 2

Using DAOSPEC

DAOSPEC finds and measures the continuum and absorption lines in stellar spectra by iterative gaussian fitting and subtraction procedures. For this reason, DAOSPEC is particularly useful when dealing with spectra that are heavily crowded with absorption lines. The course of the reduction is as follows:

- Perform a crude initial determination of the continuum level in each pixel.
- Identify lines by searching the valid flux values with a kernel corresponding to the user-specified FWHM.
- From a *preliminary* comparison between the detected lines and the laboratory wavelengths, *estimate* the radial velocity and compute the pixel values of the user-specified wavelength limits (SHORT, LONG, LEFT, and RIGHT).
- BEGIN LOOP...
  - Subtract lines from the original spectrum according to current estimates of centroid positions and line strengths;
  - Redetermine the continuum from the line-subtracted spectrum;
  - Divide the spectrum by the newly determined continuum;
  - Redetermine line centroids and strengths;

  PERFORM THIS LOOP FIVE TIMES, cranking up a secret robustness factor each time.
- END LOOP

- Cross-identify lines in the spectrum with lines in the laboratory list; compute mean and standard deviation of implied radial velocities; reject provisional cross-identifications with discrepant implied radial velocities.

- Produce output line list and (if desired) continuum spectrum and subracted (residuals) spectrum.

The following Sections describe the input files needed, the configuration parameters and the resulting output file(s).

2.1 The Spectra

DAOSPEC works on extracted one-dimensional spectra, that have been rebinned linear in wavelength. They should be FITS or IRAF images (.imh, .pix), containing the coordinate system FITS header keywords CRPIX1, CRVAL1 and CDELT1 (Greisen & Calabretta 2002, A&A, 395, 1061) in the spectrum header. These keywords specify the relationship between pixel number and wavelength (usually expressed in Å).

DAOSPEC is dimensioned to work on spectra with a maximum number of 1,200,000 pixels. This can cause a `segmentation fault` error message and crash DAOSPEC if your machine does not have enough resources. See Section 1.3 to repair this.

2.2 The Linelist

DAOSPEC needs a linelist with laboratory rest wavelengths to identify spectral lines and to estimate the radial velocity. The linelist must be an ascii file, called `laboratory.dat`, and should be in the working directory, together with the spectra.

The first eight characters on each line are the laboratory wavelength of a spectral line. After the first eight characters you can have any text that you want. This will be copied literally into the output file, so you should put here any information that you want fed to some routine that will digest the output from DAOSPEC: species identifications, oscillator strengths, whatever.

You can put any informational header/trailer/embedded lines you want into the file, as long as they do not have anything that looks like a wavelength in the first eight characters. It will be worth your while to make this line list as complete as possible, as it will help the program to find the right radial velocity. Here's a five-line example (with a three-line header):

```
Omega Centauri - Test Star
FORMAT(F7.2,1X,A2,I1,F6.2,I3,F7.2,2a1,3f1.0,F6.1)
lambda  el  kiex ref  log gf  L01
3643.72 FE1 2.61  2   -1.44  47.2 - blended?
3670.02 FE1 2.85  2   -1.06  56.2
3670.81 CA1 2.48  2   -1.95   0.0 - too weak
3676.31 CA1 2.56  2   -0.99   0.0 - ???
3686.00 TI1 2.94  2   -0.13 113.5
```

Note that although the wavelengths occupy only the first 7 characters of each line, DAOSPEC will nevertheless try to read the first 8 characters as a wavelength. Since in this case the eighth character is a blank, this is acceptable. Any other non-numerical character in column 8 would cause the program to barf. Wavelengths larger than 10,000 Å can have only two decimal places.

2.3 Configuration File and Parameters

DAOSPEC reads the user-specified options (or parameters) from an ascii file called `daospec.opt`, that must be in the working directory together with the `laboratory.dat` and the spectra. If no `daospec.opt` file is provided, all parameters are assigned default values (see below) by DAOSPEC. The same is true for any parameter missing from `daospec.opt`. Numerical values for the parameters are specified by putting:

- at least the first two characters of the parameter label (either upper or lower case);
- an equals sign;
- a numerical value.

Any additional labelling on the line should be separated from the numerical value by at least one blank character. Here is an example of valid `daospec.opt` file:

```
sh=3600
BAD DATA VALUE = 0.
Fwied wice is nice = 12.5
le = 4101.  Window from H-delta to
ri = 4340 H-gamma
```

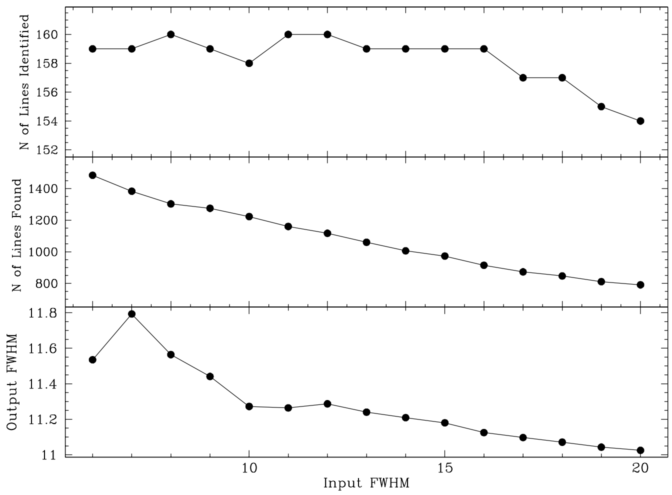

Figure 2.1: The three panels show (from top to bottom) how the number of lines identified in `laboratory.dat` varies with the input FWHM, how the total number of lines found varies with FWHM and how the final output FWHM varies with the input FWHM.

Additionally, the options can be modified from the command line of DAOSPEC (see Section 2.4), but these changes are not saved and are lost when exiting DAOSPEC. Here is an example of the parameters:

```
     FWHM (px, estimated) =    4.00     ORDER FOR CONTINUUM FIT =   10.00
   SHORT WAVELENGTH LIMIT = 3500.00      LONG WAVELENGTH LIMIT = 4518.00
      LEFT EDGE OF WINDOW = 3600.00       RIGHT EDGE OF WINDOW = 3700.00
           BAD DATA VALUE =    0.00      RESIDUAL CORE FLUX (%) =    1.00
  VELOCITY LIMIT (sigmas) =    2.00              WATCH PROGRESS =    0.00
  MINIMUM RADIAL VELOCITY =   -1.00     MAXIMUM RADIAL VELOCITY =    1.00
SMALLEST EQUIVALENT WIDTH =   15.00       CREATE OUTPUT SPECTRA =    1.00
   SCALE FWHM WITH LAMBDA =    0.00                    FIX FWHM =    1.00
```

The following sections describe each option in detail. Each section title shows in parenthesis the default value of the option parameter.

2.3.1 FWHM (fw=2.5)

FWHM is your initial estimate of the resolution of the spectrum *in units of pixels*. This is just to get the program started off on the right foot with identifying what is probably a line and what is probably noise. The precise numerical value of this parameter will be adjusted in the course of the re-

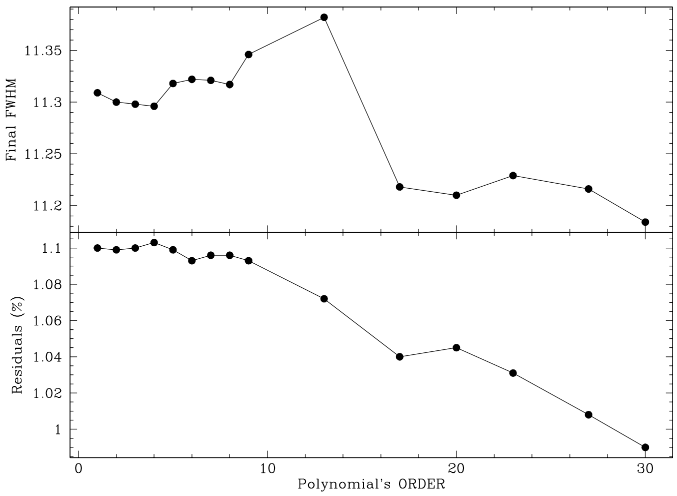

Figure 2.2: The two panels show (from top to bottom) how the final output FWHM varies with the ORDER parameter and how the residuals vary with the ORDER parameter.

ductions on the basis of the actual measured line widths, but the precise list of lines detected in the first place will vary in detail with the value of this parameter (see Fig. 2.1).

Spectral lines are modeled by *saturated gaussian* profiles: that is, they are proportional to

$$G(\lambda) = A\ e^{(-\delta\lambda^2/2\sigma^2)}$$

transformed through a function

$$F(G) = \frac{G(\lambda)}{1 + G(\lambda)}$$

such that F(G)→G as G→0 and F(G)→1 as G→ ∞. This is adequate for moderately saturated lines, but it cannot of course correctly reproduce seriously saturated lines (i.e., on the square-root/damping part of the curve of growth). The actual FWHM used for the measurements is also influenced by the SCALE and FIX parameters (see below).

2.3.2 Order for Continuum Fit (or=12)

ORDER is the order of the Legendre polynomial that will be fit to the continuum level in the spectrum. ORDER = 0 is a constant value, ORDER =

1 is a linear function of wavelength and so on. In general, the order of the polynomial sets the typical scale of variation, i.e. a 5^{th} order polynomial will vary on a scale that is $\sim \frac{1}{5} = 20\%$) of the total spectrum length. Anything up to 90^{th} order is permitted by the software.

The fitted continuum can be saved in a FITS file if CREATE (see below) is set to 1. The choice of the continuum order has a very small influence on the resulting FWHM (in our experience, about 0.1%) and the residuals are generally slightly improved when choosing higher orders (see Fig. 2.2).

2.3.3 Short/Long Wavelength Limit (sh/lo=1/999999)

SHORT and LONG wavelength limit are the extreme ends of the part of the spectrum that is valid; beyond these wavelengths the data are bad or you just are not interested. They should be specified in units of Å. SHORT and LONG also control the limits of the first display window, if WATCH PROGRESS = 1 (see below and Fig. 2.5).

2.3.4 Left/Right Edge of Window (le/ri=1/999999)

LEFT and RIGHT edge of window permit you to choose a shorter length of spectrum which will be plotted on your monitor (see below and Fig. 2.6), so that you can see in more detail what is happening as the reduction proceeds. They are only effective if WATCH PROGRESS = 1 and they should be specified in units of Å.

2.3.5 Bad Data Value (ba=0)

The software assumes that the flux values in the spectrum are positive numbers, with absorption lines having smaller numerical values than the continuum. You are informing the software that any pixel values less than the BAD DATA VALUE are defective and should be ignored. If you know that some stretches of spectrum are unwanted (between echelle orders, humongous broad Balmer lines, telluric features, ...), you can mask them off by using IRAF or other software to set them to numerical values less than BAD DATA VALUE.

2.3.6 Residual Core Flux (re=5)

In real stellar spectra, the center of a line is never really perfectly black because the effective surface of a stellar atmosphere is not at a temperature of absolute zero. By setting RESIDUAL CORE FLUX to some positive

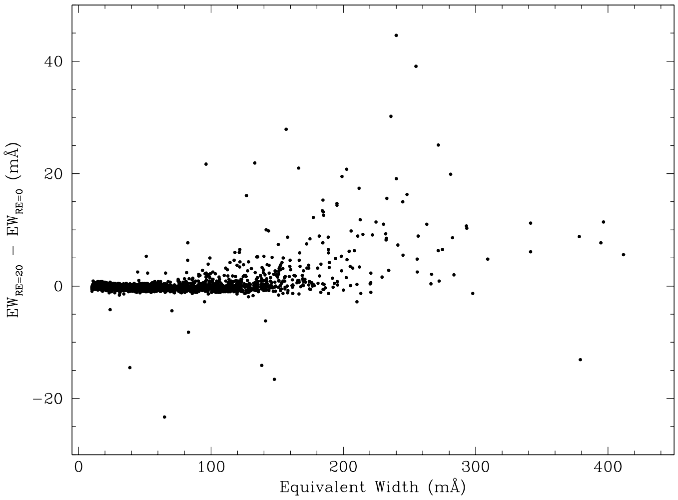

Figure 2.3: Difference between Equivalent Widths measured with RESIDUAL CORE FLUX = 20% (the appropriate value as measured on the H_α line at 6563Å) and RESIDUAL CORE FLUX = 0%, on `test.fits`.

value, you are telling the software that the lines should saturate to some particular positive flux value (expressed as a percent of the continuum level).

You can have an idea of the correct RESIDUAL value by looking at particularly strong and saturated lines in your spectrum, such as H_α at 6563Å. For example, `test.fits` has RE = 20.0 (i.e., 20% of the continuum value), while the telluric (i.e. the O_2 band around 6300Å) and interstellar absorption (i.e.the NaD lines at 5890Å) features will instead easily reach zero. For this reason, when DAOSPEC is optimally tweaked to measure weakish stellar features, it will not accurately measure telluric or interstellar features.

The RESIDUAL CORE FLUX parameter improves by a few mÅ (∼10% of the line strength) the measurements of lines of 100–200 mÅ, while it has a much larger impact on lines stronger than 200 mÅ (see Fig. 2.3). If you are interested in strong lines, you may choose to experiment with RESIDUAL CORE FLUX.

2.3.7 Velocity Limit (ve=3)

After the reductions are complete, the software will cross-identify lines detected in the spectrum with lines listed in the `laboratory.dat` file, to determine the final stellar radial-velocity estimate and to assign the measured equivalent widths to the matching line IDs. It will determine the radial velocity corresponding to each proposed line identification. Provisional identifications with implied radial velocities more than this number of standard deviations from the mean stellar radial velocity will be regarded as incorrect and the lines discarded (see Fig. 2.7).

2.3.8 Watch Progress (wa=1)

WATCH PROGRESS = 1 means draw pretty pictures on the monitor as the reductions proceed; 0 means do not (recommended for batch processing). We recommend launching DAOSPEC from an `xgterm`, expecially if working on Linux.

When you launch DAOSPEC with WA = 1, it will first display the DAOSPEC logo (Fig. 2.4), containing the maple leaf of the Canadian flag and the Southern Cross (the symbol of ESO, where part of this work has been carried out). Once you type in the spectrum name, reduction starts. For each of the five big iteration loops, DAOSPEC will first display the whole spectrum along with the residuals of the line subtraction procedure (Fig. 2.5), between the LONG and SHORT limits. Secondly, it will zoom in a region defined by the LEFT and RIGHT parameters, where more details can be appreciated (Fig. 2.6). After the five loops are finished and the two plots have been displayed five times, the radial velocity of each identified line is shown versus wavelength (Fig. 2.7). Two lines mark the σ rejection limits specified in VELOCITY LIMIT. At this point you can type in the name of the next spectrum to be reduced, or you can just hit the <ENTER> key to EXIT, at which point the DAOSPEC logo will appear again.

2.3.9 Min/Max Radial Velocity (mi/ma=-500/500)

The star is assumed to have an instantaneous geocentric radial velocity within these limits. Extreme permitted values are -1000 and +1000 km/sec (this maximum value can be modified by editing `daospec.f` and changing the value of the `VELLIM` parameter). Specifying a more restricted range of radial velocities through MIN and MAX will make things go a bit faster and will reduce the (already small) chance that the software will make an incorrect initial guess at the actual radial velocity.

2.3.10 Smallest Equivalent Width (sm=5)

The software will not print out results for any spectral line with an equivalent width smaller than this value (expressed in mÅ). It will carry along weaker lines internally, but will not report them in the output file.

2.3.11 Create Continuum File (cr=0)

CREATE OUTPUT SPECTRA = 1 means produce two output spectra (FITS or IRAF—the same as the input spectrum). The first contains the fitted continuum level as a function of wavelength, on the same wavelength and flux scale as the input spectrum, for overplotting if you want. Its default name is, e.g., `testC.fits` if the original spectrum was called `test.fits`. The second will be the *residual* spectrum, i.e. the original spectrum, normalized to the fitted continuum and with all the lines subtracted. Its default name is, e.g., `testR.fits`. CREATE OUTPUT SPECTRA = 0 means create neither the continuum file nor the residual file.

2.3.12 Scale FWHM with λ (sc=0)

When an echelle spectrum is wavelength calibrated and rebinned linear, it will have constant resolution:

$$R = \frac{\lambda}{\delta\lambda} = Const;$$

therefore the line width $\delta\lambda$ will increase with wavelength. If SCALE FWHM = 1, the software will take this fact into account when fitting lines and make FWHM proportional to wavelength. In this case, the whole reduction procedure will be slightly slower and the output FWHM estimate will refer to the center of the spectrum, as specified by SHORT and LONG. If SCALE FWHM = 0, the a single FWHM will be used to fit all lines, regardless of their wavelength.

2.3.13 Fix FWHM (fi=0)

When FIX = 1, DAOSPEC will not adjust the FWHM during the reduction procedure, but will instead apply the user-provided FWHM value. If SCALE = 1, this value will refer to the center of the spectrum, as specified by SHORT and LONG, and the actual FWHM will be proportional to wavelength. If FIX is left to 0, DAOSPEC will automatically find the best FWHM value, based on the residuals of the subtraction procedure.

2.4 Running DAOSPEC

Before running DAOSPEC, we recommend that you put closely related spectra into a common subdirectory, and dissimilar spectra into different directories. Rule of thumb: if two spectra can share a common daospec.opt and laboratory.dat file, put them into the same subdirectory. Otherwise, not.

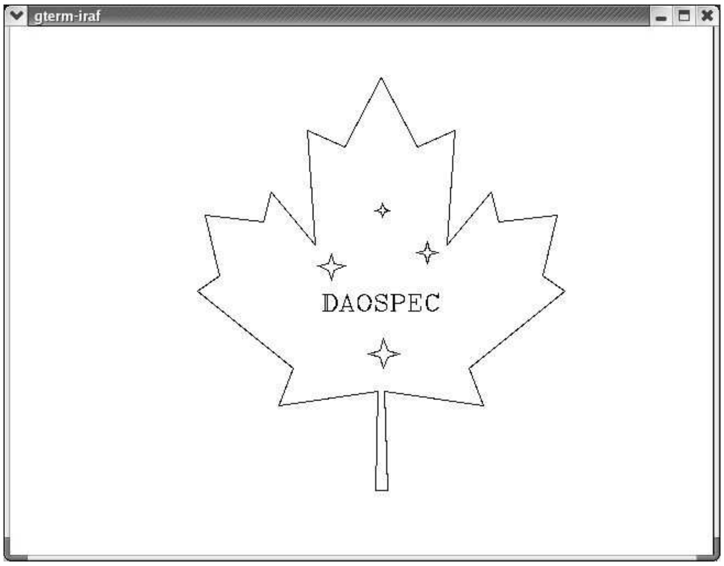

Figure 2.4: DAOSPEC initial window.

To run DAOSPEC (after you have properly installed it) simply type `daospec` on a terminal (in our experience, it is better to use an `xgterm`, expecially with Linux). If DAOSPEC's `PATH` is not properly set, you will have to type it in entirely, i.e. `/usr/local/bin/daospec`, or to copy the program in your working directory and then type `./daospec`.

When launched, DAOSPEC will automatically try to read the `laboratory.dat` and `daospec.opt` files. Then it will produce this output on screen:

```
     FWHM (px, estimated) =    4.00      ORDER FOR CONTINUUM FIT =   10.00
   SHORT WAVELENGTH LIMIT = 3600.00        LONG WAVELENGTH LIMIT = 4518.00
      LEFT EDGE OF WINDOW = 3500.00         RIGHT EDGE OF WINDOW = 3700.00
           BAD DATA VALUE =    0.00       RESIDUAL CORE FLUX (%) =    1.00
  VELOCITY LIMIT (sigmas) =    2.00               WATCH PROGRESS =    0.00
  MINIMUM RADIAL VELOCITY =   -1.00      MAXIMUM RADIAL VELOCITY =    1.00
SMALLEST EQUIVALENT WIDTH =   15.00        CREATE OUTPUT SPECTRA =    1.00
   SCALE FWHM WITH LAMBDA =    0.00                     FIX FWHM =    1.00

OPT>
```

Now, you can modify the options on the command line, if you like. You simply have to type the first two letters of each option, followed by an equals sign and the numerical value (see also Section 2.3). After you have changed the desired options, or if you like the options as they are, simply type return. However, please note that these changes will not be saved in `daospec.opt`,

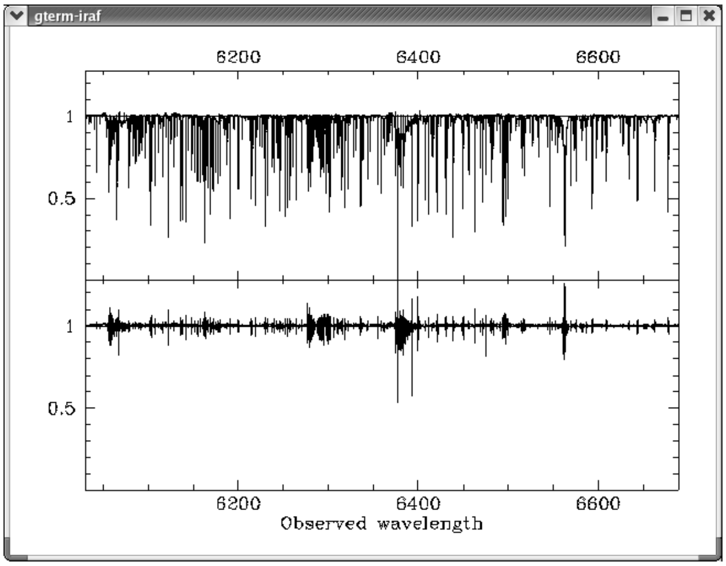

Figure 2.5: The first DAOSPEC display window, showing the continuum normalized spectrum (upper panel) and the line subtracted spectrum (lower panel),within the limits specified by the SHORT and LONG parameters.

and will thus be lost when you exit the program. To change the options permanently, you must edit the file `daospec.opt`.

Now DAOSPEC thinks a while and, if WATCH PROGRESS = 1, it opens the display window (see Section 2.3.8 and Fig. 2.4) containing the DAOSPEC logo. On the command line, you will be asked for the file name: type it in and hit return. If the filename extension is `.fits`, `.fit`, or `.imh`, you can omit it if you like; if it is anything else, you must include it. (If the filename extension is *anything* other than `.imh`, the image file *must* be in FITS format.) The header keyword `OBJECT` is now displayed, together with the spectrum size in pixels and a first estimate of the radial velocity:

```
                        Input spectrum: WFI512938_U.fits
RED_SCIENCE_REDU

                          Picture size:   27889

           Estimated radial velocity:  +218 km/s

FWHM
4.08
```

The FWHM is continuously updated on the screen, while the display win-

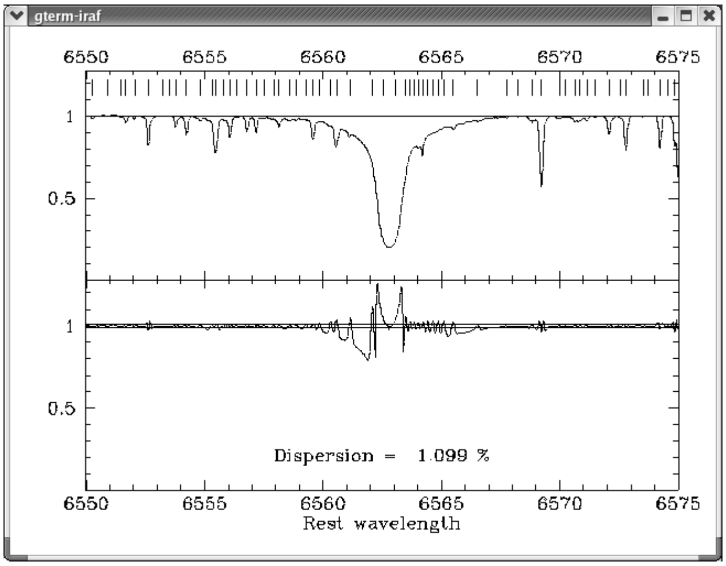

Figure 2.6: The second DAOSPEC display window, showing an enlarged region of the spectrum specified by the LEFT and RIGHT parameters. In this case the H_α line region has been chosen. The continuum normalized spectrum is shown (upper panel), with the identified lines marked by ticks at the top. The subtracted spectrum is also shown (bottom panel) with an estimate of the average residual dispersion in the spectrum as a whole—not just this region. The local continuum level is determined only from pixels with residual values lying between the two horizontal lines.

dow shows alternately the whole spectrum (Fig. 2.5) and an expanded region specified by the user through the LEFT and RIGHT parameters (Fig. 2.6), until the five big loops are completed. When the procedure ends, the final radial velocity estimate appears along with some basic information, and more details will be shown on the display window (Fig. 2.7).

```
              Final radial velocity:  +47.495 km/s
                   Velocity scatter:    2.168 km/s
                           Based on:     2701 lines

       Input spectrum (default EXIT):
```

Finally, the prompt appears again, asking for a new spectrum name. Typing EXIT (or simply hitting RETURN) at this point exits DAOSPEC.

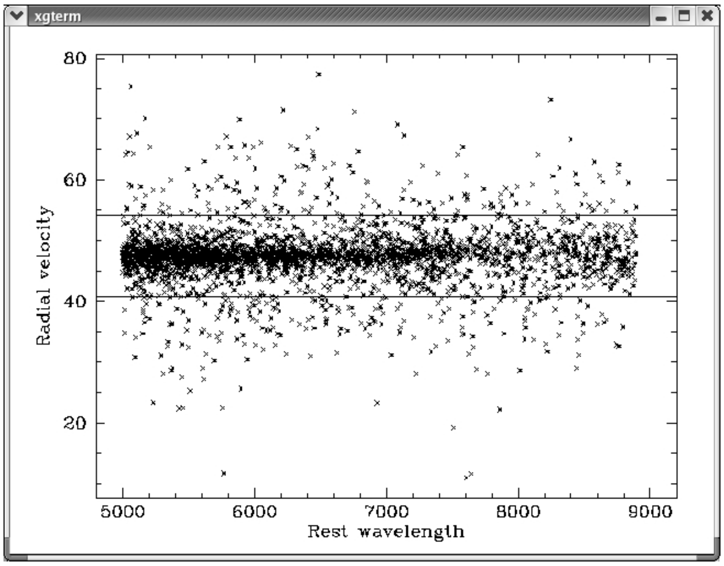

Figure 2.7: Last DAOSPEC display window, showing how the final radial velocity is determined and how discrepant lines are rejected. A 3 σ rejection has been applied in this case.

2.5 Output File

Apart from the continuum and residuals files discussed above, DAOSPEC produces one single ascii file for each spectrum: if the original file was called `test.fits`, the output file will be called `test.daospec`.

Here is an example of DAOSPEC output file:

```
Radial velocity = -0.171   dispersion = 0.429   FWHM = 11.454    158 lines
    matched out of 1070
Relative flux dispersion in residual spectrum = 1.113%
3600.115  3597.494  64.4 20.00 2.088
3600.379  3597.758  82.7 18.86 4.617
3601.731  3599.108  19.0  4.44 0.844
3606.458  3603.832  57.3  7.30 1.587 3603.84 TI1 0.00 1 -3.38 86.9 -blend?
3613.684  3611.053  27.2  5.66 0.839
3613.864  3611.233  19.1  5.59 0.819
3614.289  3611.657  34.5  9.13 1.586
3635.198  3632.551  34.8  8.02 1.644 3632.56 FE1 2.95 2 -1.03 67.2
(1)       (2)       (3)   (4)  (5)   (6)     (7)
```

The file contains two lines of header (with information about the estimated radial velocity, FWHM, number of spectral lines and residuals) and one line of data for each spectral line found. The columns contain:

- *(1)* the line center wavelength observed in the spectrum, in Å;
- *(2)* the line center rest wavelength in Å, corrected for the derived radial velocity;
- *(3)* the measured equivalent width in mÅ;
- *(4)* the estimated error in the measurement, derived in the usual way from the least-square fitting procedure;
- *(5)* a quality parameter, derived from a comparison between the residuals observed in the spectrum in the immediate neighborhood of the line, as compared to the typical residuals in the spectrum as a whole—large values are bad;
- *(6)* if the line has been identified, this column contains the original laboratory wavelength as it appears in `laboratory.dat`;
- *(7)* if the line has been identified, the remaining information from `laboratory.dat` is appended literally here.

2.6 Batch Processing

Like any Fortran program, DAOSPEC can be run from a shell script. This is particularly useful if you have to run it on a number of different spectra with similar characteristics, i.e., sharing the same linelist and most of the DAOSPEC options.

Here is a simple script for reducing multiple spectra (i.e., `star01.fits`, `star02.fits` and so on) with the *same* input parameters (i.e., same `daospec.opt` and `laboratory.dat`) using csh, without the grapics, and storing screen output in a log file called `logfile`.

```
# !/bin/csh                   # Tell the system where's csh
#                             # Comment
rm star??.daospec             # Remove previous output
daospec << DONE >!  logfile   # Run DAOSPEC and save log
FW = 7.5                      # Set FWHM
WA = 0                        # Mandatory for batch processing
                              # Carriage return to exit options
star01                        # Work on first spectrum
star02                        # ...  second spectrum
```

```
star03                        # ...
star04                        # ...
star05                        # ...  last spectrum
                              # Carriage return to exit DAOSPEC
DONE                          # End of input list for DAOSPEC
```

Please note that empty lines are important, because they are actually carriage returns: you need the first one to exit the options and the second one to exit DAOSPEC. Please also note that removing the `*.daospec` is also very important: if you provide DAOSPEC the name of an *existing file*, it will ask if you want to overwrite it, a question that is not normally asked and that will therefore change the order with which commands are given.

If you still are running DAOSPEC on a list of files, but this time would like to change a few parameters from one spectrum to the next, you can first create an ascii file, containing the rootname of each spectrum in each line, followed by one or more parameters that you may want to change. In this example we will change two parameters, FWHM and SCALE. Here is your ascii file called, i.e., `spectra.lst`:

```
star01 7.6 1.0
star02 9.1 1.0
star03 8.3 1.0
star04 2.5 0.0
star05 5.2 0.0
```

This means that we want to run DAOSPEC with FWHM = 7.6 and SCALE = 1 for `star01.fits` and so on. The following shell script (this time using ksh) will read your list, launch DAOSPEC once for each row/spectrum, change the two parameters and save the screen output in a log file for future reference.

```
#!  /bin/ksh -f               # Tell the system where's ksh
#                             # Comment
cat spectra.lst | \           # Read your list containing:
while read spec fwhm scale    # spectrum name, fwhm and scale
do                            # Start do loop on file's rows
rm ${spec}".daospec"          # Remove previous output
rm ${spec}"C.fits"            # ...
rm ${spec}"R.fits"            # ...
./daospec <<END>> ${spec}.log # Start DAOSPEC and save log
   wa=0                       # Mandatory in batch mode
   cr=1                       # Optional to create fits outputs
```

```
   fw=$fwhm                        # Feed variable $fwhm to DAOSPEC
   sc=$scale                       # Feed variable $scale to DAOSPEC
                                   # Carriage return to exit options
   ${spec}.fits                    # Feed spectrum name to DAOSPEC
                                   # Carriage return to exit DAOSPEC
END                                # End of input list for DAOSPEC
done                               # End of loop, repeat on next row
```

As a result of the above script, you will produce a few files for each of your spectra. For example, `star01.fits` will have now: `star01.daospec`, the daospec output file with equivalent witdths; `star01C.fits`, the fitted continuum; `star01R.fits`, the residual spectrum of the line subtraction procedure and `star01.log`, containing the output that is normally printed on screen.