Documentation for SPECTRUM v2.76

Richard O. Gray

May 27, 2010

Contents

1 Introduction 5

2 Quick Installation & Use 7

2.1 Directions for UNIX and Linux 7

2.1.1 Preliminaries 7

2.1.2 Compiling Spectrum 7

2.2 Installing under Windows 9

2.3 Running Spectrum 10

2.4 Finalizing the Installation (UNIX/Linux) 11

3 Detailed Documentation for Spectrum 15

3.1 Spectrum Switches 15

3.2 Stellar Atmosphere Models 18

3.3 Data Files 20

3.3.1 Linelist files 20

3.3.2 Atomic and Molecular data file 22

3.3.3 Isotope file 24

3.4 The Abundance Scale 26

3.5 Atomic and Molecular Species Supported 27

3.6 Isotopes & Hyperfine Structure 28

3.7 More Details on Spectral Line Broadening 31

3.7.1 Normal Lines 31

3.7.2 Autoionizing Lines 32

3.8 How Spectrum Carries out its Calculations 33

3.9 Other Considerations 37

3.9.1 Model Atmosphere Selection 37

3.9.2 The Line-Spread Function and the Comparison of Synthetic and Observed Spectra 38

3.9.3 Further Practical Notes on Computing Synthetic Spectra 39
3.10 What is New in SPECTRUM v2.76? 39

4 SPECTRUM Auxiliary Programs 41
4.1 ABUNDANCE 41
4.2 AVSINI 42
4.3 BFLXSM2 43
4.4 BLACKWEL 43
4.5 BSMOOTH2 45
4.6 CUSTOMSM 45
4.7 GFADJUST 46
4.8 LINES 47
4.9 MACTURB 48
4.10 SELECT 49
4.11 SMOOTH2 49
4.12 VSINI 50

5 Graphical User Interfaces 51
5.1 SPECTRUM GUI 51
5.2 BLACKWEL GUI 53
5.3 A Windows GUI 55

Chapter 1

Introduction

Spectrum is a stellar spectral synthesis program designed for use on a wide range of platforms, including most types of UNIX and Linux. Given a stellar atmosphere model and certain other inputs, Spectrum calculates a synthetic stellar spectrum. Spectrum carries out this computation under the assumptions of local thermodynamic equilibrium (LTE) and a plane-parallel atmosphere. Currently, spectrum is suitable for computing stellar spectra in the spectral-type range B – mid M.

Spectrum may be programmed with command-line switches to give a number of different output formats. For instance, in default mode, spectrum computes the disk-integrated normalized intensity spectrum, but with the appropriate switches, spectrum can be directed to compute either the specific intensity from any point on the stellar surface or the disk-integrated flux. Spectrum currently supports all naturally occurring elements, along with a total of 310 isotopes. Hyperfine structure is also supported. In addition, spectrum can handle 15 diatomic molecules important in stellar atmospheres, including TiO and ZrO. Isotopic versions of most of these molecules are also supported.

Spectrum comes with a suite of supporting programs, including programs for computing elemental abundances from equivalent widths (abundance), a program for simultaneously determining abundances and the microturbulent velocity (blackwel), a program that aids in the identification of stellar spectral lines (lines), and various other auxiliary programs to convert the output of spectrum to a form that can be compared directly with observations.

Note: This documentation has been written specifically for versions 2.76 and later. For version 2.75, please follow the directions on the main spec-

TRUM website for compilation and installation.

Chapter 2

Quick Installation & Use

2.1 Directions for UNIX and Linux

2.1.1 Preliminaries

The following section assumes that there is a C-compiler installed on your system, in particular the GNU project C-compiler, `gcc`, which is a common resource on many Linux and Unix systems. If this compiler is not installed on your system, it may be downloaded from the GNU gcc website `http://gcc.gnu.org/`. If your system has a "native" C-compiler installed (this may be the case for SUN or DEC systems, for instance), then this compiler may be employed instead of `gcc`, but read the entirety of the next section before you attempt compilation. If you have `gcc` on your Mac computer, you should be able to follow the UNIX/Linux instructions.

2.1.2 Compiling Spectrum

SPECTRUM is available for download as a tarred-gzipped file on `http://www.phys.appstate.edu/spectrum/spectrum.html`. On the download page of that website, you will find a link to a file named `spectrumXXX.tar.gz`, where `XXX` stands for the current version. This file should be downloaded and copied to a directory, say, `/spectrum`, and then unzipped with the command:
`tar -xvzf spectrumXXX.tar.gz`.
This command unzips to this directory source code for SPECTRUM in the form of a number of "c" files, such as `spectrum.c`, `opacity6.c`, and so on, along with a number of essential data files, such as `stdatom.dat` which contains basic atomic and molecular data, `isotope.iso`, a listing of sup-

ported isotopes and relative abundances, plus two stellar atmosphere models (`sun.mod` and `vega.mod`) computed with the stellar atmosphere code atlas9 (Kurucz, 1993). You should also see a file named `Makefile`.

The spectrum source files must be compiled first before you can use spectrum. This means that you must have a C-compiler on your system. If not, see the notes above.

Spectrum may be compiled using the provided `Makefile`. Make certain you are in the directory in which you unzipped `spectrumXXX.tar.gz`. At the prompt, issue the command

```
make spectrum
```

and, if all goes well, in about a minute or two spectrum should compile without error. If an error occurs, make certain that you are in the correct directory and that you have write permission for that directory. For other errors, which should be very rare at this point, please contact the author (see contact information in the appendix). The executable for spectrum will be called `spectrum` (as opposed to `spectrum.c` or `spectrum.o`).

If you are interested at this point, all of the auxiliary programs (plus spectrum itself, for that matter) may be compiled with the command

```
make all
```

This will produce the executables `abundance, avsini, bflxsm2, blackwel, bsmooth2, customsm, gfadjust, lines, macturb, selectmod, selectmod9, spectrum, smooth2,` and `vsini`. The use of these programs will be covered later in this document.

If your compiler is not `gcc`, it should be possible invoke it with the command `cc` (try entering `cc` at the prompt; if you get the response: `cc: no input files`, then you can use `cc` instead of `gcc`). However, you must modify the Makefile in order to compile spectrum with `cc`. Use an editor, such as `nano` or `emacs` and modify the first line of the Makefile so that it reads

```
CC = cc
```

and then use the command

```
make spectrum
```

to compile spectrum. You may also have to experiment with the second line

```
CFLAGS = -O3
```

which sets the level of optimization for the compiler. If you experience any problems running spectrum, modify that line to

```
CFLAGS = -O
```

and try again. You may then work up through the various levels of optimization: `O1, O2, O3` and possibly beyond.

To compile for a 64-bit machine, refer to the instructions for your compiler (this applies to `gcc` as well) for the appropriate flags which you may then add to the `CFLAGS` line. The Makefile includes a sample 64-bit optimization line:

```
CFLAGS = -O3 -m64 -mtune=native
```

which can be experimented with. Uncomment it (remove the leading #) and comment the default `CFLAGS` line and compile. The last flag `-mtune=native` may not work on all systems.

You should also download a spectral linelist file from the SPECTRUM distribution site. There are a number available, but `luke.lst` is the best to start with. This can be downloaded as a gzipped file, `luke.lst.gz` which will then need to be unzipped with the command

```
gunzip luke.lst.gz
```

This file, at least for now, may be placed in the `/spectrum` directory; later in this document we will cover placing it in a separate directory, such as `/usr/local/lines`.

2.2 Installing under Windows

SPECTRUM and selected auxiliary programs can be run under Windows using the precompiled binaries available on the distribution page. These binaries were compiled using MinGW (`http://www.mingw.org/`). It is not necessary for you to download and install MinGW to use these binaries, however, as they will run in a DOS (or C prompt) window. The binaries come in the form of a zipfile, `spectrumXXX.zip`, which should be downloaded and unzipped in a folder. I recommend that these binaries be installed in a folder directly under the c: prompt, for instance, `c:\spectrum,` instead of under "My Documents", as the blanks that are permitted in Windows folder and file names are not fully compatible with a program written for a Linux/UNIX environment. You will also have to install the file `stdatom.dat`, the linelist file, and the stellar atmosphere models into the same directory.

These executable binaries all have the extension `.exe`, and were compiled under XP. They should operate under Vista, but I have not had the opportunity to test them. To run SPECTRUM open a C: prompt window, and change to the spectrum directory (for instance, `cd c:\spectrum`). To carry out the test outlined in the next section, simply type `spectrum` at the prompt and respond appropriately to all of the questions.

SPECTRUM may be run under Windows using a GUI. Please see § 5.3 for more information.

2.3 Running Spectrum

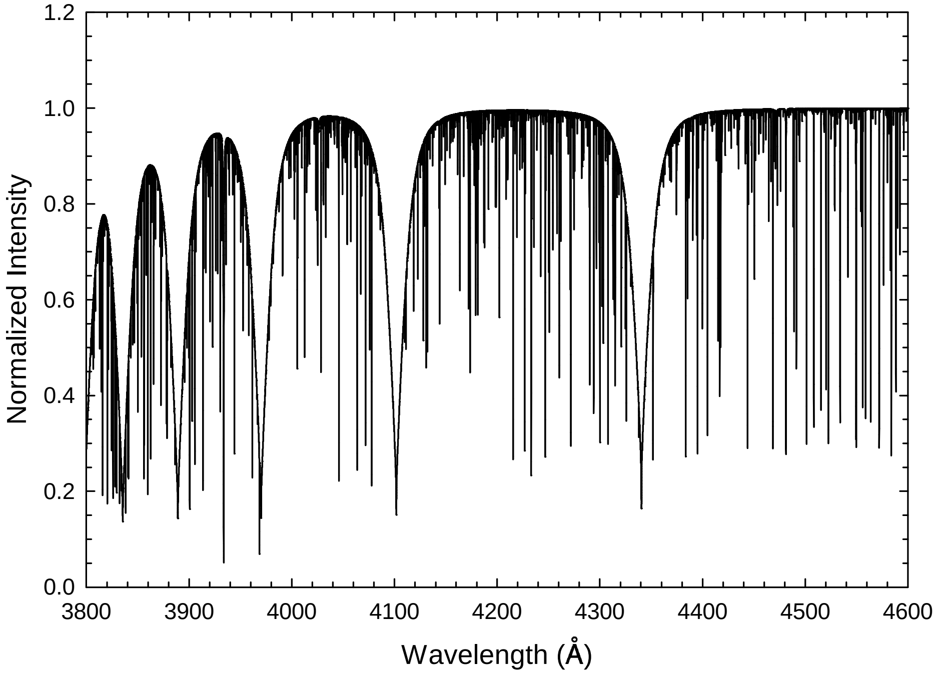


Figure 2.1: The result, `vega.spc`, of carrying out the test run on SPECTRUM described in § 2.3.

This section covers running SPECTRUM in its default mode. To learn how to run SPECTRUM in other modes in order to exploit its full power, please refer to later chapters in this document.

To run SPECTRUM in its default mode, three data files must be available. The first of these is a *stellar atmosphere model.* Two of these models have been provided in the SPECTRUM distribution, `vega.mod` and `sun.mod`. It is also necessary to have an atomic and molecular data file; the one recommended for general use is `stdatom.dat` (provided in the distribution). For this exercise this file *must be in your working directory*, i.e. the directory in which you unzipped SPECTRUM. Lastly, you must have a linelist file. A number of these are available on the SPECTRUM website. For this exercise, download `luke.lst.gz`. Unzip this file with the command `gunzip` luke.lst.gz. This is a large file and contains a few hundred thousand spectral lines. At least for now, place this file in the directory in which you unzipped the SPECTRUM distribution file. Now, confirm that you are in the directory in which you unzipped SPECTRUM and issue the following command at the linux prompt (>) and respond as indicated:

```
> ./spectrum

Enter name of stellar atmosphere data file > vega.mod

Enter name of linelist file: (default = luke.lst) > luke.lst

Enter name of output file > vega.spc

Enter microturbulence (km/s) > 2.0

Enter beginning and ending wavelengths (A) > 3800.0,4600.0

Enter wavelength step (A) > 0.02
```

At this point, SPECTRUM will take off and print to the screen the wavelengths from 3800.0 to 4600.0Å, with a spacing of 0.02Å. When it finishes (in a few minutes, depending on your machine), the output synthetic spectrum will be found in the ascii file vega.spc. This output file has a simple format: two columns with the wavelength in angstroms in the first column, and the normalized intensity in the second. It may be plotted on the screen with any plotting program, such as gnuplot and it should look like Figure 2.1. If your plot looks like Figure 2.1, then SPECTRUM has been compiled correctly and is running properly.

2.4 Finalizing the Installation (UNIX/Linux)

If all has gone well with the above test, you are ready to finalize the installation of SPECTRUM and all of its auxiliary programs (see Chapter 4). First, you need to determine where you are going to place all of the executable binaries. In most cases, you will want to place the executable binaries in /usr/local/bin, but if you are on a multiuser system, and you are the only one who will be using SPECTRUM, it might be better to place the executables in a local "bin" directory. Consult your system administrator. If the executables are to be placed in /usr/local/bin, then no modifications need

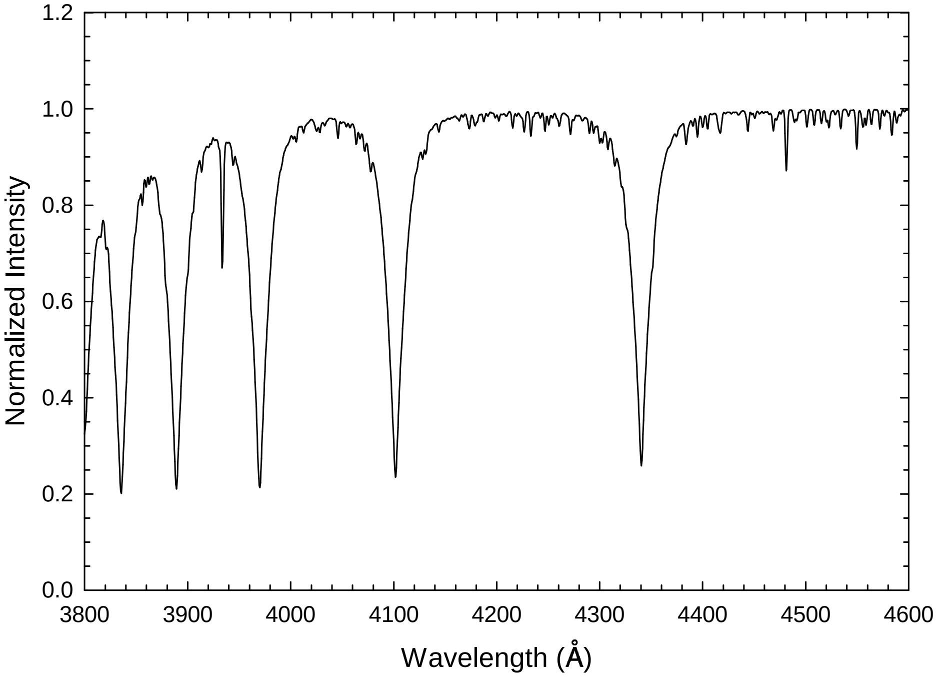


Figure 2.2: The final smoothed spectrum (`vega.sm`) obtained by using the program SMOOTH2 to convolve a Gaussian line spread function (2Å resolution) with the synthetic spectrum `vega.spc` produced in the test described in § 2.3. How to use SMOOTH2 is described briefly in § 2.4.

to be made to `Makefile`. If you want them to go elsewhere, edit `Makefile` and change the line

```
LOCALBIN = /usr/local/bin
```

so that `/usr/local/bin` is replaced with the path where you want the executables to go. Once all is properly set up, issue the command

```
make all
```

which will compile SPECTRUM and all its auxiliary programs. Then issue the command (you will probably need to be `root` or `superuser` to do this)

```
make install
```

which will move all the executables to the `bin` directory. Once this is done, you will be able to invoke SPECTRUM simply by typing `spectrum` at the linux prompt; it will no longer be necessary to include the leading `./` as in the test in § 2.3. Please remember that you will need to have the file `stdatom.dat` in whatever directory you use for your working directory (unless you use switch "a" – see § 3.1 below). SPECTRUM looks in the working directory for `stdatom.dat` unless you use that switch. You will probably also want to move the linelist file `luke.lst` to a special directory, such as `/usr/local/lines`.

Now, as one last test (this assumes you have carried out the test described in § 2.3, and that you are still in the `/spectrum` directory), run the program `smooth2`:

```
> smooth2

Enter name of input file > vega.spc

Enter name of output file > vega.sm

Enter spacing in Angstroms of the input spectrum > 0.02

Enter output resolution in Angstroms > 2.0

Enter spacing in Angstroms of spectrum in output file > 0.5
```

This will smooth the output spectrum produced in § 2.3 to a resolution of 2Å with a wavelength spacing of 0.5Å. The result should look like Figure 2.2. If your computer cannot find the program SMOOTH2 this either means that the install step was not successful in moving the SPECTRUM programs to your `bin` directory, or the `bin` directory is not in your `PATH`. On some systems, you may need to issue the command `rehash` before the operating system knows the location of the newly moved executables.

There is a graphical user interface (GUI) available for SPECTRUM that integrates the operation of SPECTRUM, SMOOTH2 and other auxiliary programs and simplifies the process of computing synthetic spectra. See Chapter 5.

Chapter 3

Detailed Documentation for Spectrum

3.1 Spectrum Switches

The default mode of operation of spectrum calculates the normalized integrated-intensity spectrum using the abundances in the atomic and molecular data file, `stdatom.dat`. In this default mode, spectrum prints to the screen the wavelength currently being computed along with two other numbers. The actual synthetic spectrum is printed to an ascii file. The two numbers printed to the screen refer to the number of spectral lines under computation at the current wavelength and the number of spectral lines in a "look-ahead" list. See § 3.8 for more details. The mode of operation of spectrum can be changed by using one or more of the following "switches".

Spectrum Switches:

a: Prompts the user to supply the name of a custom atomic and molecular data file. This is useful if the user needs to compute a spectrum with a non-standard set of abundances. Without this switch, spectrum will read in the default file `stdatom.dat`. The custom file should have *exactly* the same format as the supplied file `stdatom.dat`. While `stdatom.dat` *must* be in your working directory, this switch gives the user the opportunity to specify the path to the custom data file. The full path must be given. See § 3.3 for details on the information contained in `stdatom.dat`.

b: This switch will cause SPECTRUM to output the synthetic spectrum in a binary format. The advantage of this is that the size of the output file will be about 5 times smaller than the standard ascii output. The disadvantage is that it is much more difficult to work with than the normal ascii output file. This mode *cannot* be used under Windows, and if used under Linux/UNIX the output file permissions must be altered with `chmod` to make it readable by the SPECTRUM auxiliary programs BSMOOTH2 and VSINI. This switch was introduced "in the old days" when disk space was at a premium. I don't recommend using this switch at present.

c: Ignores all spectral lines and outputs only the continuum.

f: This switch causes SPECTRUM to compute the disk integrated absolute flux (in erg/cm^2/s/Å as measured at the stellar surface).

g: An experimental mode that allows the user to input a "velocity gradient" file to study the effects of non-zero velocities on the line profiles. The format is an ascii file with a single column of velocities, one line corresponding to each layer in the stellar atmosphere. This switch should only be used in conjunction with the specific intensity modes (switches m and M) and only for center of disk calculations.

i: Isotope mode. SPECTRUM fully supports the calculation of both atomic and molecular isotopic spectra, but this requires the use of a special format for the linelist file. In addition, in this mode SPECTRUM will prompt the user for an isotope file. The default isotope file is `isotope.iso`. See § 3.6 for more details.

m,M: Both switches invoke the specific intensity mode. This allows the user to calculate, for instance, the center-of-disk solar spectrum, or the emergent specific intensity spectrum at any other point on the disk. The "m" switch causes SPECTRUM to output the normalized specific intensity (i.e. the continuum is normalized to 1.0); the "M" switch outputs the true, non-normalized specific intensity. This mode could be used to build up a spectrum for a star with a non-uniform surface, such as a rapidly rotating star with $T_{\rm eff}$ and $\log g$ which vary from point to point on the surface, or a spotted star, or even the emergent spectrum from a binary system.

n: Silent mode. In this mode, SPECTRUM prints out only a few lines of information and then runs silently. It will not even prompt the user for the various inputs, and thus in this mode SPECTRUM must be used with a *response file*. See § 3.9 for more details. The advantage of running in this mode is that SPECTRUM will run much faster if it doesn't have to print wavelength information to the screen.

p: This switch will cause SPECTRUM to print out number densities of selected species (such as atoms, ions and molecules) at each depth in the atmosphere to a file named `density.out`.

t: This switch signals SPECTRUM that the input stellar atmosphere model is in the ATLAS9/12 format. See § 3.2 for more details.

u: An experimental mode that allows the user to compute a spectrum with a microturbulent velocity that varies with depth. The program will read the microturbulent velocity from the seventh column of the stellar atmosphere model.

x: In this mode, the user is prompted for a "fixed abundance file" which can contain the abundances of any number of elements that the user does not wish to be scaled by the [M/H] factor (see the end of § 3.3.2 for further instructions).

Switches are used on the command line. For example, if we want SPECTRUM to operate in the isotope mode, to read in a stellar atmosphere model in the ATLAS9 format, and to output the absolute flux, we would use the command line:

```
spectrum itf
```

Or, if we wish to operate SPECTRUM in the silent mode with a response file (`spec.rsp`):

```
spectrum n < spec.rsp
```

Notice that the "switch string" should be separated from the `spectrum` command by at least one space, and that the string itself should not contain any spaces. For instance, the following is illegal and will not produce the desired result:

```
spectrum ni tf < spec.rsp
```

Using a Response File with SPECTRUM

As mentioned in the above paragraph, SPECTRUM may be run from the command line using a *response file*. This file contains the responses to all the prompts SPECTRUM prints to the screen. Thus, the file `vega.rsp` (included in the distribution) may be used to run the test example in § 2.3 simply by using the command line:

```
spectrum n < vega.rsp
```

The contents of `vega.rsp` are as follows:

```
vega.mod
luke.lst
vega.spc
2.0
3800.0,4600.0
0.02
```

This assumes that the linelist file `luke.lst` is in the working directory. If it is not, that line should include the full path. Note that the last line of the response file must end in a carriage return. By using response files, shell batch files can be set up that will run multiple SPECTRUM sessions successively. Also note that the use of some switches result in extra SPECTRUM prompts, necessitating extra lines in the response file.

3.2 Stellar Atmosphere Models

One of the required inputs for SPECTRUM is a stellar atmosphere model. Three of these have been provided in the distribution, an ATLAS9 model for the sun, one for Vega, and one for the parameters $T_{\rm eff} = 6720$K, $\log g = 4.40$ and [M/H] = 0.0 (the file `67244k2p00.mod`). These model atmospheres were computed with the ATLAS9 model atmosphere program written by Kurucz (1993). More such models are available on Robert Kurucz's website: `http://kurucz.harvard.edu` under the link "Grids of Model Atmospheres". Another excellent site where individual models may be downloaded is `http://wwwuser.oat.ts.astro.it/castelli/grids.html`. Before we get into extracting a particular model atmosphere for use in SPECTRUM let us take a closer look at the model atmosphere for the sun. The first few lines of this model atmosphere, found in the file `sun.mod` are reproduced below:

```
5777.0  4.43770  0.00  64
2.62383508E-03   3398.7 7.188E+01 6.618E+09 7.359E-04 1.133E-01 1.500E+05
2.62384162E-03   3871.5 7.188E+01 1.301E+10 7.359E-04 8.267E-02 1.500E+05
3.17880896E-03   3901.4 8.709E+01 1.566E+10 8.502E-04 7.839E-02 1.500E+05
4.02509434E-03   3937.8 1.103E+02 1.966E+10 1.019E-03 7.581E-02 1.500E+05
4.97406188E-03   3969.0 1.363E+02 2.409E+10 1.202E-03 7.460E-02 1.500E+05
```

The model continues for 59 more lines; in total, this particular model atmosphere is divided up into 64 layers, each layer characterized by a different temperature, pressure, electron density, etc. The first line in this model is the header which gives the effective temperature ($T_{\rm eff} = 5777.0$K), the $\log g$ (4.43770; logarithm of the surface gravity in units cm/s^2), the overall metallicity in logarithmic terms compared with the sun ([M/H]; 0.00), and the number of layers. SPECTRUM will use the value for [M/H] to scale the abundances of the metals (everything except for H and He) in the atomic and molecular data file `stdatom.dat`. SPECTRUM can handle model atmospheres with up to 100 layers. If you have an application that requires a larger number of layers, please contact me. The single line header "`5777.0 4.43770 0.00 64`" is in the format of the *default* SPECTRUM header.

The next 64 layers in this atmosphere model contain data needed by SPECTRUM for calculating the synthetic spectrum. The first layer represents the surface. The first column is the mass depth. The second column is the temperature, in kelvins, of the layer, the third the gas pressure, the fourth the electron density, the fifth the Rosseland mean absorption coefficient, the sixth the radiation pressure and the seventh the microturbulent velocity in meters/second. The newer Kurucz/Castelli models have three additional columns which give the amount of flux transported by convection, the convective velocity and the sound velocity.

The atmosphere models from the Kurucz and Castelli websites (or as directly produced by the ATLAS9 and ATLAS12 programs) have more elaborate headers than the SPECTRUM default header. You can choose either to edit the atmosphere model to remove these headers and replace them with the single line header detailed above, or use the "`t`" switch which alerts SPECTRUM to expect an ATLAS9 or 12 header. The ATLAS9 and 12 headers contain not only effective temperature, $\log g$ and [M/H] information, but also the specific elemental abundances used by ATLAS9 or 12 to compute the stellar atmosphere. SPECTRUM, in the "`t`" mode, picks up $T_{\rm eff}$, $\log g$ and [M/H] from these headers, but does not, at least as presently constituted, use the elemental abundances in these headers. Rather, the elemental abundances in the `stdatom.dat` file are used (and scaled with the [M/H] value). An option may be added in the future to make it possible to read this information from the stellar atmosphere header. It turns out, however, that for certain applications of SPECTRUM it is very useful to have this information in a separate file.

Even though SPECTRUM was written with the Kurucz atmospheres in mind, this does not mean that other model atmospheres cannot be used. They simply must be cast into the format of the Kurucz models. For instance, the MARCS site (`http://marcs.astro.uu.se/`) distributes stellar atmosphere models. The models with the extension `.krz` are in essentially the same format as the Kurucz models, although the headers are quite different. The present version of SPECTRUM cannot handle the MARCS headers, so if you wish to use these models, remove the MARCS header and add a header in the SPECTRUM default format (see above). Please also note that the MARCS atmospheres come in two versions – *plane-parallel* or *spherical* – according to the geometry used in the computation. SPECTRUM is compatible only with the plane-parallel atmospheres.

The Kurucz atmosphere models from the Kurucz website are distributed in large files in which many models are concatenated (what I refer to in later sections of this document as a "supermodel"). These supermodels may be opened with an ascii editor and a single model extracted by cutting and pasting to another file. However, in Chapter 4, a program called `selectmod` is described which can be used to extract a single model from a supermodel.

As noted above, SPECTRUM is designed to be used with *plane parallel atmospheres* such as the ones produced by ATLAS9 and ATLAS12. SPECTRUM does not yet contain code to handle spherical atmospheres.

3.3 Data Files

SPECTRUM requires, in addition to a stellar atmosphere model two or three more input data files. These data files are 1) a linelist file, 2) an atomic and molecular data file and, optionally, 3) an isotope file. These files are considered in turn below:

3.3.1 Linelist files

A linelist file, suitable for use from 3000 – 6800Å for luke-warm stars (mid B to K-type stars) is available on the download webpage. This file, called `luke.lst`, contains data for over half a million atomic and molecular lines. The format of a typical line is as follows:

```
4045.813   26.0   11976   36686    0.280  1.000 01  NIST
```

The first entry is the wavelength of the spectral line in angstroms. The second is the species code, which gives both the atomic number and the ionization state. The code "26.0" refers to an Fe I (neutral iron) line, as the atomic number of iron is 26, and the neutral ionization state is given as decimal 0. The code for Fe II is 26.1, the code for Ca III is 20.2 and so on. For diatomic molecules, the code specifies the atomic makeup of the molecule. Thus, H_2 is 101.0, the two "1"s referring to the two hydrogens, CH is 106.0, CO 608.0, MgH 112.0, TiO 822.0, etc. The lightest element always comes first in the code, so that 608.0 cannot be confused with NdO, which would be written 860.0. The third entry is the energy of the lower state in cm^{-1}, and the fourth, in the case of atomic species, the energy of the upper state. Since only the energy of the lower state is used in molecular calculations, this entry is sometimes used to encode the molecular band information. The fifth entry is the log(gf) value, that is the logarithm of the product of the statistical weight of the lower level and the oscillator strength for the transition. The sixth entry is a fudge factor which can be used to adjust the calculation for the van der Waals broadening of the line. The seventh entry encodes the type of transition; "01" refers to an s–p transition, "12" a p–d transition, etc. The default coding is "99" which is used when 1) the compiler of the linelist has been too lazy to determine the type of transition, 2) when the transition does not fit into these simple categories (for instance it may involve two electrons) or 3) for a molecular line. Three other possible codings for the seventh entry are "AO", "GA", and "AI". "AO" enables the user to provide SPECTRUM, in a packed format, the σ and α parameters used in the Anstee and O'Mara broadening theory. "GA" can be used to input the individual broadening half-widths for natural broadening, Stark broadening and van der Waals broadening. These two options are discussed in more detail in § 3.7.1 below. "AI" can be used in the computation of autoionizing lines; see § 3.7.2. Finally, the eighth entry gives the source of the data. In the example above, "NIST" obviously stands for the National Institute of Standards and Technology, and their wonderful and very useful atomic line and level website, `http://physics.nist.gov/PhysRefData/ASD/index.html`. If the source begins with a "K", this means it comes from the Kurucz line lists which contain tens of millions of both atomic and molecular lines. Most of the data in luke.lst come from those sources. The Kurucz data may be picked up on his website (`http://kurucz.harvard.edu`).

More linelist files in the format required by SPECTRUM may be picked up from the SPECTRUM distribution site, and others are available on special request. For instance, `luke.lst` does not include lines of TiO or ZrO, but a file called `cool5.lst` which does is available.

If SPECTRUM is operated in the isotope mode, an additional column must be included in the linelist file. See the discussion in § 3.6 below.

3.3.2 Atomic and Molecular data file

Another required input for SPECTRUM is an atomic and molecular data file. One such file, `stdatom.dat`, which includes solar atomic abundances from Grevesse & Sauval (1998), is included in the distribution. The format of that file can be seen from the following sample lines from the beginning

```
code   Abund   Amass   I1/D0   I2/rdmass     I3         I4     maxcharge
  1   -0.0360 1.008  13.5984    0.000       0.000      0.000       0
  2   -1.1060 4.003  24.5874   54.4178      0.000      0.000       1
  3  -10.88    6.941  5.3917   75.6400    122.4543     0.000       1
  4  -10.89    9.012  9.3227   18.2111    153.8966   217.7187      1
  5   -9.25   10.811  8.2980   25.1548     37.9306   259.3752      1
  6   -3.49   12.011 11.2603   24.3833     47.8878    64.4939      3
```

and from the end

```
606    -7.50  24.02    6.15     6.0055      1.00      00.00        0
607    -7.50  26.01    7.66     6.4627      1.00      00.00        0
608    -7.50  28.01   11.108    6.8604      1.00      00.00        0
813    -7.50  42.981   5.27    10.0436      1.00      00.00        0
814    -7.50  44.06    8.26    10.1897      1.00      00.00        0
822    -7.50  63.879   6.87    11.9921      1.00      00.00        0
840    -7.50 107.223   7.85    13.6118      1.00      00.00        0
```

The first column is the atomic or molecular code, described in the previous section. For instance, "6" refers to carbon, and "606" to the molecule C_2. The second column contains, for the atoms, the logarithmic solar abundance in terms of number densities relative to the total number density: $\log(A/N_{\text{total}})$. This is the way that the abundance scale is defined in ATLAS9 and 12, but it is not the standard way of representing elemental abundances. See § 3.4 for more details. For the molecules, this column is set equal to -7.50 for obscure reasons. The third column contains the mean atomic mass for both atoms and molecules. The fourth column contains for atoms the first ionization energy in electron volts (eV); for molecules this column contains the dissociation energy, also in eV. The fifth column contains for atoms the second ionization energy and for molecules the "reduced mass" $(m_1 m_2/(m_1 + m_2))$. The sixth column contains, for the atoms, the third ionization energy, and for the molecules a "fudge" factor by which the "gf" value for each spectral line is multiplied. This gives the user a very rough and blunt instrument for modifying molecular band strengths, for instance, after their appearance in the solar spectrum. The seventh column gives, for the atoms the fourth ionization energy and is identically zero for the molecules. Finally, the eighth column gives the maximum charge (the highest ionization state) supported by SPECTRUM. For instance, a "1" means that SPECTRUM can compute spectral lines only for neutral and singly ionized species (I and II), a "3" means that SPECTRUM has support for ionization states I, II, III and IV. For molecules this entry is 0. All naturally occurring elements are supported (hydrogen – uranium) by SPECTRUM and 15 diatomic molecules. Adding support for another molecule is not as simple as adding another line to `stdatom.dat`. It also involves code changes, addition of the molecular partition function, etc.

If you want to compute a spectrum with non-standard abundances, this can be accomplished by modifying the relevant abundances in `stdatom.dat`. However, a number of points must be kept in mind.

First, the overall metallicity in the header for the stellar atmosphere model (see section above) is used to scale the abundances for the metals in `stdatom.dat`. Thus, if [M/H] = -1.0, SPECTRUM subtracts 1.0 from the logarithmic abundances in `stdatom.dat` for lithium through uranium. Since the opacity due to the iron peak elements is so important in establishing the structure of the stellar atmosphere, it is advisable to use a stellar atmosphere model with [M/H] as close as possible to the [Fe/H] of the star you are analyzing.[1]

Let us suppose that you want to compute a stellar atmosphere model with all but a handful of metals having an abundance one tenth that of the sun. Assuming that iron is not included in that handful, select the appropriate stellar atmosphere model with [M/H] = -1.0 and then individually adjust the abundances in `stdatom.dat` for the handful. For instance, in the sun (and in `stdatom.dat`), the logarithmic abundance of calcium (code = 20) is -5.68. Let us suppose that you actually want to use an abundance of -6.38. If the [M/H] of the model is -1.0, then the [M/H] scaling changes the abundance from `stdatom.dat` to -6.68, so this means that to get an abundance of -6.38 you will need to adjust the abundance in `stdatom.dat` to -5.38. Note that this is equivalent to [Ca/Fe] = +0.30 if [M/H] = [Fe/H].

Another example: Let us suppose that you want to compute a spectrum with iron and all other metals (except for, say, calcium) with abundances 0.15 dex less than found in the sun. Let us say that [Ca/Fe] = +0.30. Select a Kurucz atmosphere model with [M/H] = -0.20 (ATLAS9 models with [M/H] = -0.15 are not available, so -0.20 is the closest). Adjust the abundances of all the metals (except for calcium) in `stdatom.dat` upwards by 0.05 dex, and calcium upwards by 0.35 dex.

[1] Of course, with ATLAS12, it is possible to compute a stellar atmosphere with any mix of abundances you want.

If this all seems to be complicated enough to drive one to tears, v2.76 now has a facility to input a “fixed abundance” file that can be used to specify individual fixed elemental abundances *which will not be adjusted by the [M/H] scaling factor*. Suppose that we want to compute a spectrum with a logarithmic calcium abundance = -6.38, regardless of the value of [M/H] of the model. Create a file (name it anything you like) with the contents:

```
TOTAL
20 -6.38
```

and run spectrum with the “x” switch (see § 3.1). Spectrum will prompt you for a “fixed abundance file”. Supply it with the name of the file you just created. This file can contain abundances for any element except for hydrogen and helium. You may specify abundances on the basis of total number densities (use the keyword `TOTAL` on the top line of the file), or in terms of the number density of hydrogen (use the `HYDROGEN` keyword on the top line) – see § 3.4. The abundances of any number of elements may be specified in the fixed abundance file, each element on a separate line. Note that the element code (in this case 20, signifying calcium) *must be entered as an integer*, i.e. with no decimal points. Specifying `20.0` instead of `20` will result in unpredictable results!

3.3.3 Isotope file

Spectrum fully supports the calculation of atomic and molecular isotopic spectra, although not in the default mode. To enable the isotope mode, use switch “i”. This will cause spectrum to prompt the user for an isotope file; the one provided in the distribution is `isotope.iso`. The types of information contained in this file can be understood from the following sample lines:

```
  1.0   1    1.007825  0.999885
  1.0   2    2.0140    0.000115
  2.0   3    3.016029  0.00000137
  2.0   4    4.002603  0.99999863
  3.0   6    6.015121  0.0759
  3.0   7    7.016003  0.9241
  4.0   9    9.012182  1.000
  4.0  10   10.013534  0.000
  5.0  10   10.012937  0.199
  5.0  11   11.009305  0.801
  6.0  12   12.000000  0.9893
  6.0  13   13.003355  0.0107
```

The first column gives the atomic code; 1.0 is for hydrogen, 2.0 for helium, etc. The second column gives the mass number for the isotope. Thus, `1.0 2` stands for deuterium, `6.0 13` for ^{13}C. The third column gives the molecular weight for the isotope, and the fourth column the relative abundance in the solar system. So, for instance, in the solar system, 98.93% of carbon is ^{12}C and 1.07% is ^{13}C. Note that not all isotopes are represented in isotope.iso; ^{14}C is a radioactive isotope of carbon with a short half-life ($\sim$ 5000 years) and so should not be present with any significant abundance in the atmospheres of stars of any type. Thus, only astrophysically important isotopes are included in `isotope.iso`.

If the user is interested in calculating spectra with unusual isotopic abundances, then the relative abundances should be modified in `isotope.iso`. However, it is up to the user to compile the required spectral linelist. The spectral linelist used in the isotope mode has an additional column that specifies the mass number or "isotope code" for each spectral line. See § 3.6 for more information.

Spectrum supports a total of 310 isotopes, the ones listed in `isotope.iso`. Spectrum also supports molecules that involve isotopes. For instance, there are three astrophysically important versions of the C$_2$ molecule, ^{12}C^{12}C, ^{13}C^{12}C, and ^{13}C^{13}C. Bandheads and individual molecular lines for these three isotopic versions of C$_2$ have slightly different wavelengths. The molecules currently supported in isotope mode, along with their "isotope codes" are listed below:

Molecule	Isotope Code
12C H	12
13C H	13
12C D	2
14N H	14
15N H	15
14N D	2
16O H	16
17O H	17
18O H	18
16O D	2
24Mg H	24
25Mg H	25
26Mg H	26
24Mg D	2
28Si H	28
29Si H	29

```
30Si H          30
28Si D           2
40Ca H          40
42Ca H          42
43Ca H          43
44Ca H          44
46Ca H          46
48Ca H          48
40Ca D           2
12C 12C         12
12C 13C         13
13C 13C         33
12C 14N         12
12C 15N         15
13C 14N         13
12C 16O         12
12C 17O         17
12C 18O         18
13C 16O         13
28Si 16O        16
29Si 16O        29
30Si 16O        30
28Si 17O        17
28Si 18O        18
```

Note: isotopes for H_2, TiO and ZrO are not yet supported. See § 3.6 for more detailed information.

3.4 The Abundance Scale

It was mentioned above that the way the abundance scale is expressed in `stdatom.dat` (or for that matter, Kurucz's model atmosphere programs Atlas9 and 12) is not the standard way that is usually found in the literature. Normally, abundances of elements are expressed with respect to hydrogen, on a scale in which the log of the abundance of hydrogen is set equal to 12.0. Spectrum uses a scale in which the abundances are expressed with respect to the total number density of atoms (and ions). To convert between the two scales is easy; use the following formula

$$\log(A/N_H) = 12.0 + \log(A/N_{\rm Total}) - \log(N_H/N_{\rm Total})$$

For $\log(N_H/N_{\rm Total})$, spectrum uses the hydrogen abundance in the atomic and molecular data file (`stdatom.dat` or a corresponding custom file). The value in the current version of `stdatom.dat` is -0.0360.

It should be noted that SPECTRUM has been used to calculate successfully spectra for hydrogen-poor (helium-rich) models. To introduce a non-standard hydrogen and/or helium abundance, it is necessary to edit `stdatom.dat` AND use a model calculated for those abundances. If you do not use a model with hydrogen and helium abundances consistent with those found in `stdatom.dat`, it is likely that SPECTRUM will fail, as the equilibrium equations will have no solution.

3.5 Atomic and Molecular Species Supported

As mentioned in previous sections, SPECTRUM supports all 92 naturally occurring elements, including a total of 310 isotopes of those elements. The ionization states supported for those elements can be found in the atomic and molecular data file, `stdatom.dat`. The very last column of that file indicates, for a given element, the maximum charge state for which spectra can be computed. For instance, the maximum charge for iron (26) is 3, meaning that spectra can be computed for Fe I, Fe II, Fe III and Fe IV. The ionization states supported depend upon the availability of partition function data in the literature. To calculate accurate number densities for a given ionization state, partition function data for the next higher ion must be built into SPECTRUM.

Molecular species supported by SPECTRUM are also listed in `stdatom.dat`. Currently, only diatomic molecules are supported; those supported are H_2, CH, NH, OH, MgH, SiH, CaH, C_2, CN, CO, SiO and TiO. In v2.76, AlH, AlO, and ZrO were added. Other diatomic molecules will be added in the future, and can be added fairly easily on request, if partition function data are available. Molecules involving different isotopes are also supported. See § 3.3.3 above and § 3.6 below.

It should be noted that spectral lines for certain species are "hard coded" into SPECTRUM. This includes certain hydrogen-line series (Lyman, Balmer, Paschen, Brackett, Pfund and Humphreys), and lines of He I. He II lines are on the "to do list" to be coded into SPECTRUM. All of these lines require special calculations for Stark broadening and various other effects, such as forbidden components in the He I profiles. Certain other very strong lines, such as the Ca II K & H lines have also been hard-coded into SPECTRUM. You can see which lines are in this "strong-line" list by looking at the code for the function `strong6.c`. The point is that the hard-coded lines should not be included in the spectral linelist, or they will be computed twice and appear to either be unusually strong, or have a peculiar profile.

3.6 Isotopes & Hyperfine Structure

Isotopes are fully supported by SPECTRUM in the "isotope mode", which can be invoked with the "i" switch (see § 3.1). It must be understood that SPECTRUM does not automatically calculate isotope shifts; the isotopic components of a given spectral line must be included in the spectral linelist as individual entries. The format of an isotopic spectral linelist is different from the format for the standard linelist; it must include an additional column indicating the mass number or "isotope code" of the isotope in question. As an example, consider the following entries used for the computation of the Li I 6708Å resonance line:

```
6703.568   26.0  0  22249   37163   -3.15   1.000 01  NIST
6704.476   26.0  0  34017   48928   -2.659  1.000 10  NIST
6705.106   26.0  0  37186   52090   -1.17   1.000 99  AGL
6707.44    26.0  0  37186   52090   -2.40   1.000 99  AGL
6707.752   21.0  0  32659   47563   -2.672  1.000 99  K23
6707.754    3.0  7      0   14904   -0.431  1.000 01  AGL
6707.766    3.0  7      0   14904   -0.209  1.000 01  AGL
6707.771   20.0  0  46748   61652   -4.015  1.000 99  K23
6707.824   26.1  0  93488  108392   -3.094  1.000 99  K23
6707.904    3.0  7      0   14904   -0.733  1.000 01  AGL
6707.917    3.0  7      0   14904   -0.510  1.000 01  AGL
6707.921    3.0  6      0   14904   -0.005  1.000 01  AGL
6708.053   26.1  0  93485  108388   -3.614  1.000 99  K23
6708.072    3.0  6      0   14904   -0.306  1.000 01  AGL
6708.094   23.0  0   9825   24728   -3.113  1.000 99  KFE
6708.167   26.1  0  89727  104630   -3.979  1.000 99  K23
6710.316   26.0  0  11976   26875   -4.874  1.000 21  NIST
6712.44    26.0  0  40231   55125   -2.149  1.000 99  NIST
6713.77    26.0  0  38678   53569   -1.602  1.000 12  NIST
```

Compare the format of this linelist with that illustrated in § 3.3.1. The third column contains the mass number of the isotope involved; for instance, the entry for the spectral line with a wavelength of 6707.754Å in the above list is for the isotope ^{7}Li, and the one for 6707.921Å is for ^{6}Li. Note that many of the spectral lines in the above example have isotope codes of "0". The "0" code corresponds to the case in which the isotope shifts are considered to be zero, and thus these entries represent all possible isotopes for that species taken together. So, for instance, since iron (26) has isotopes with mass numbers 54, 56, 57, and 58, the single entry

```
6703.568   26.0  0  22249   37163   -3.15   1.000 01  NIST
```

is equivalent to the entries

```
6703.568   26.0 54  22249   37163   -3.15   1.000 01  NIST
6703.568   26.0 56  22249   37163   -3.15   1.000 01  NIST
6703.568   26.0 57  22249   37163   -3.15   1.000 01  NIST
6703.568   26.0 58  22249   37163   -3.15   1.000 01  NIST
```

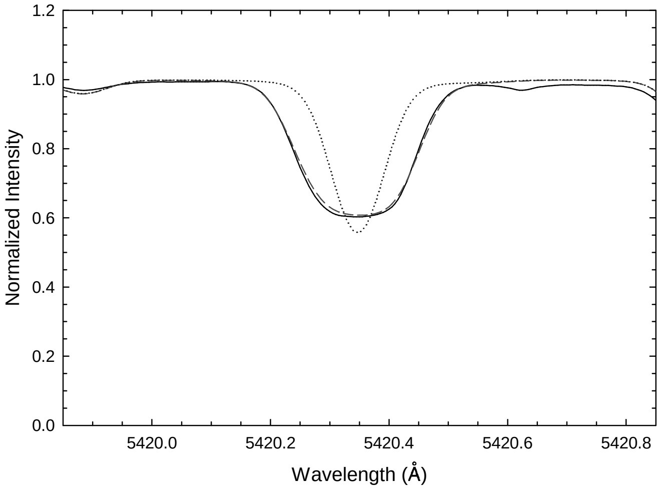


Figure 3.1: The Mn I 5420Å line in the solar spectrum. The observed spectrum is shown as the solid dark line, the synthetic spectrum without hyperfine structure taken into account is shown as the dotted line, and the dashed line is the synthetic spectrum with hyperfine structure taken into account. The $\log gf$ value used in calculating the line was taken from the NIST website and the hyperfine structure constants were taken from Dembczyński et al. (1979) and Luc & Gerstenkorn (1972).

Isotopes with odd nucleon numbers can exhibit hyperfine structure; elements with odd atomic numbers are especially subject to this phenomenon. Thus, lines of vanadium (23), manganese (25), cobalt (27) and copper (29) in the solar spectrum commonly show unusually broad profiles formed from the superposition of a number of hyperfine components (see Figure 3.1). It is quite a frequent occurrence for a single line to show both isotope shifts and hyperfine structure; the Ba II 4554Å resonance line is a case in point, and so is the important Eu II 4129.7Å resonance line, the components of which are listed below:

```
4129.628  -63.1   151       1   24208   -1.512  1.000 01  LWDS
4129.631  -63.1   151       1   24208   -1.035  1.000 01  LWDS
4129.645  -63.1   151       1   24208   -1.316  1.000 01  LWDS
4129.649  -63.1   151       1   24208   -0.977  1.000 01  LWDS
4129.653  -63.1   151       1   24208   -1.512  1.000 01  LWDS
4129.670  -63.1   151       0   24208   -1.257  1.000 01  LWDS
4129.675  -63.1   151       0   24208   -0.847  1.000 01  LWDS
4129.680  -63.1   151       0   24208   -1.316  1.000 01  LWDS
4129.703  -63.1   151       0   24208   -1.294  1.000 01  LWDS
4129.708  -63.1   153       0   24208   -1.512  1.000 01  LWDS
4129.709  -63.1   151       0   24208   -0.696  1.000 01  LWDS
4129.711  -63.1   153       0   24208   -1.035  1.000 01  LWDS
4129.715  -63.1   151       0   24208   -1.257  1.000 01  LWDS
4129.715  -63.1   153       0   24208   -1.316  1.000 01  LWDS
4129.718  -63.1   153       0   24208   -0.977  1.000 01  LWDS
4129.721  -63.1   153       0   24208   -1.512  1.000 01  LWDS
4129.725  -63.1   153       0   24208   -1.257  1.000 01  LWDS
4129.728  -63.1   153       0   24208   -0.847  1.000 01  LWDS
4129.732  -63.1   153       0   24208   -1.316  1.000 01  LWDS
4129.740  -63.1   153       0   24208   -1.294  1.000 01  LWDS
4129.743  -63.1   153       0   24208   -0.696  1.000 01  LWDS
4129.744  -63.1   151       0   24208   -1.480  1.000 01  LWDS
4129.746  -63.1   153       0   24208   -1.257  1.000 01  LWDS
4129.751  -63.1   151       0   24208   -0.545  1.000 01  LWDS
4129.757  -63.1   151       0   24208   -1.294  1.000 01  LWDS
4129.760  -63.1   153       0   24208   -1.480  1.000 01  LWDS
4129.761  -63.1   153       0   24208   -0.545  1.000 01  LWDS
4129.764  -63.1   153       0   24208   -1.294  1.000 01  LWDS
4129.786  -63.1   153       0   24208   -0.401  1.000 01  LWDS
4129.787  -63.1   153       0   24208   -1.480  1.000 01  LWDS
4129.801  -63.1   151       1   24208   -0.401  1.000 01  LWDS
4129.808  -63.1   151       1   24208   -1.480  1.000 01  LWDS
```

Obviously, in this example, SPECTRUM must be used in the isotope mode, as the Eu II 4129.7Å line involves not only hyperfine components but also isotopic components (indicated by the isotope numbers 151 and 153). Note in this example that the species code (63.1 for Eu II) is entered as a negative number. This prompts SPECTRUM to increase the computation radius for such spectral lines by a factor of three. This is done because each hyperfine component taken on its own may be a very weak spectral line (and thus would have a small computation radius assigned to it), but since it overlaps with other closely spaced hyperfine components, if its computation radius were not increased the far wings of the combined spectral line could be truncated. Indeed, this trick can be used if you observe SPECTRUM truncating the far wings of *any* line; enter the species code for that line as a negative number, and the normal computation radius for that line will be tripled, up to a maximum of 20Å (10Å in the optical). A handful of lines, such as the Ca II K and H lines, obviously need a larger computational radius than this, and such lines are actually hard-coded into SPECTRUM. See § 3.8 for more details.

3.7 More Details on Spectral Line Broadening

3.7.1 Normal Lines

SPECTRUM includes code for all the normal spectral line-broadening mechanisms, including natural, Doppler, Stark, resonance and van der Waals. For most metallic lines, van der Waals broadening dominates. SPECTRUM includes code to implement the Anstee & O'Mara van der Waals broadening theory (see Anstee & O'Mara, 1991, 1995). This broadening theory is automatically implemented for neutral atomic species if the transition type is indicated in the linelist file (see § 3.3.1 for how this is done) for s-p, p-s, p-d, d-p, d-f and f-d transitions. For transitions involving higher angular momentum quantum numbers, or for ionized species, the classical van der Waals theory is implemented, with the mean square radii of the states given by the hydrogenic approximation:

$$\langle r^2 \rangle = \frac{n^{*2}}{2}[5n^{*2} + 1 - 3l(l+1)]$$

where n^* is the effective principal quantum number. If the transition type is not indicated, then the classical van der Waals broadening theory is used, but with the term involving the angular momentum quantum number neglected in the expression for the mean square radius.

However, for ionized species, the Anstee–O'Mara theory can be invoked if the "AO" transition type is specified in the linelist. This requires the user to include the σ and α parameters of that theory in a packed format immediately after the "AO" in the linelist file. For instance, Barklem & O'Mara (1998) have calculated these parameters for certain lines of ionized species; for the Ca II λ8498 line, $\sigma = 291$, $\alpha = 0.275$. Below is an example of how those data can be included in the linelist file:

```
8498.023  20.1   13650   25414   -1.416  1.000 AO  291.275 T89
```

Of course, it is also possible to use the "AO" transition type with neutral species if, for some reason, you are not satisfied with the tabulated values of σ and α built into SPECTRUM.

Finally, if the user wants complete control over natural, quadratic Stark and van der Waals broadening, it is possible to input the relevant half-widths using the "GA" transition type. These half-widths may either be calculated by the user with her favorite broadening theory, or extracted from the original Kurucz linelist files (see `http://kurucz.harvard.edu`) or from, for instance, the VALD database (`http://ams.astro.univie.ac.at/~vald/`). As an example, in the Kurucz hyperfine linelist file, for the line Fe I λ4620.128 it is given that $\log(\Gamma_{\rm rad}) = 8.32$, $\log(\Gamma_{\rm Stark}) = -6.20$ and $\log(\Gamma_{\rm vdw}) = -7.78$. These values may be entered into the linelist file in the following way:

```
4620.140   26.0  24772   46410   -3.739  1.000 GA 8.32 -6.20 -7.78  NIST
```

The Stark half width is per electron number (i.e. SPECTRUM multiplies the input half width by the electron number density) and the van der Waals half width is per neutral hydrogen number and is for $T = 10000$K. The van der Waals half width is further modified by SPECTRUM to take into account van der Waals broadening due to neutral helium and H_2 collisions. Bear in mind that these half-widths are *logarithmic* quantities. If you truly want to enter a zero Stark or van der Waals half width, don't enter 0.00! Enter a large negative quantity such as -15.0. SPECTRUM interprets a 0.00 entered for either the Stark or the van der Waals half-width as an indication that the user would like the program to make its own estimate for those half-widths, and that is exactly what it does.

3.7.2 Autoionizing Lines

Autoionizing lines arise from doubly excited states in which the upper level of the transition is above the ionization energy. Autoionizing lines are characterized by a Fano profile instead of a Voigt profile. Fano profiles depend on three parameters: $\Gamma_{\rm Shore}$, $A_{\rm Shore}$, and $B_{\rm Shore}$, where $\Gamma_{\rm Shore}$ is the profile half width, $A_{\rm Shore}$ is the asymmetry parameter, and $B_{\rm Shore}$ is related to the oscillator strength. More information on these parameters may be found in

Shore (1967) and Shore (1968). Most autoionizing lines are in the ultraviolet, but there are a few such lines that appear in the optical. Perhaps the best known of these are the Ca I autoionizing lines in the orange (see below). Beginning with version 2.76, the Fano parameters may be entered into the line list in a way similar to entering the radiation, Stark and van der Waals half widths described in the previous subsection. For instance, for the autoionizing Ca I line at 6343.308Å, enter:

```
6343.308   20.0  0   35819   51579    0.845  1.000 AI 11.28 -30.00 -13.38  AUTO
```

where $\log(\Gamma_{\rm Shore}) = 11.28$, $\log(A_{\rm Shore}) = -30.00$, and $\log(B_{\rm Shore}) = -13.38$. Note, however, if the asymmetry parameter $A_{\rm Shore} < 0$, the entry should be -log($|A_{\rm Shore}|$), which will be a positive quantity. The Fano parameters may be computed from first principles using the references cited above, by fitting resonances in photoionization cross sections (such as those from the Opacity Project[2]), or they may be derived from the Kurucz line lists or other sources. Note the `AI` transition type. Autoionizing lines are often extraordinarily broad. Figure 3.2 shows a synthesis of the region around the three orange Ca I autoionizing lines.

3.8 How Spectrum Carries out its Calculations

Spectrum carries out some preliminary calculations before getting down to the business of calculating the spectrum itself. Spectrum uses as input a stellar atmosphere model with the format of a Kurucz model, but actually only utilizes the columns corresponding to mass depth, temperature and gas pressure. Spectrum then computes for itself the electron number density and the number densities of important species, namely, hydrogen, helium, carbon, nitrogen, oxygen, their ions and the relevant diatomic molecules. Specifically, in this preliminary stage, spectrum solves a system of 7 nonlinear equilibrium equations to determine those number densities. The exact details of these equations depend upon the effective temperature of the star. For $T_{\rm eff} > 8500$K, no molecules, except for H_2 are included in the equilibrium equations, but for lower temperatures molecules are included. The 7 equilibrium equations are:
1) Equilibrium equation for hydrogen, including neutral and ionized hydrogen, the H^- ion, H_2, H_2^+, and, for $T_{\rm eff} < 8500$K, H_2O, CH, NH and OH.
2) Equilibrium equation for Helium, including two stages of ionization.
3) Equilibrium equation for Carbon, including up to four stages of ionization

[2] `http://cdsweb.u-strasbg.fr/topbase/xsections.html`

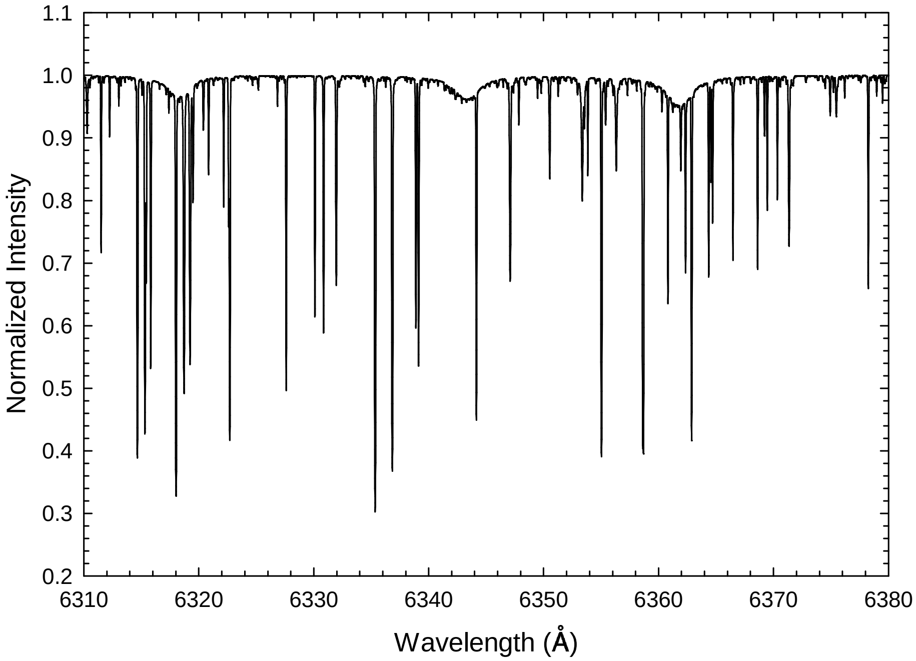


Figure 3.2: The three broad, shallow structures seen in this synthetic spectrum (computed with v2.76b using a solar model and shown at full resolution) are the orange autoionizing lines of Ca I. See § 3.7.2 for how to calculate these structures.

(i.e. C I, C II, C III, C IV and C V), and the molecules CH,CN,CO and C_2 (the number of stages of ionization depends on the effective temperature of the model. For instance, for $T_{\rm eff} > 8500$K, four stages of ionization are considered, but the molecules are left out; likewise for the Nitrogen and Oxygen equations below).
4) Equilibrium equation for Nitrogen, including up to four stages of ionization, and the molecules CN, NH, NO, N_2.
5) Equilibrium equation for Oxygen, including up to four stages of ionization, and the molecules CO, NO, OH, O_2.
6) Charge balance equation, including electron contributions from the ionization of hydrogen, helium, carbon, nitrogen and oxygen and the following species: sodium, magnesium, aluminum, silicon, sulfur, potassium, calcium

and iron.
7) Total number density, including all species mentioned above, which must agree with P_{gas}/kT as read from the stellar atmosphere.

This system of equations is solved iteratively at each of the levels in the stellar atmosphere. Electron number densities so derived are always within 1% of those in the stellar atmosphere model. Convergence is rapid and the abundances of the different species and molecules are stored for each of the atmosphere levels in memory.

Spectrum then moves on to the computation of reference opacities and optical depths at each level in the atmosphere. Opacities currently included in spectrum are: hydrogen bound-free and free-free, He I bound-free and free-free, He II bound-free and free-free, H^- bound-free and free-free, H_2^+ opacity, hydrogen Rayleigh scattering, H_2 Rayleigh scattering, He I Rayleigh scattering, He I^- free-free opacity, low temperature opacities (including continuous opacities due to C I, Mg I, Al I, Si I, Ca I, Fe I, CH, OH and MgH), intermediate temperature opacities (including opacities due to N I, O I, Mg II, Ca II and Si II) and electron scattering. With version 2.75, bound-free opacities for He I, C I, Mg I, Si I and Ca I have been updated with data from the Opacity project, although some of these have been modified by comparison with observations.

With these preliminaries completed, spectrum can now begin to compute the synthetic spectrum. The numbers or words in parentheses correspond to the parameters used in the ultraviolet part of the spectrum. Spectrum basically computes the spectrum in 20 (5) Å blocks, although smaller segments of spectra can be synthesized. For each 20 (5) Å block, spectrum first computes the continuum opacity and the emergent continuum flux at both ends of each block (the exception to this is if the 20 (5) Å block contains a continuous absorption edge. Then spectrum modifies the red end of this block to 0.05 Å before the absorption edge; the next block straddles the absorption edge, and then the new block starts at 0.05Å past the absorption edge). The continuous opacity and continuum flux at every intermediate point is then estimated by interpolation. Spectrum then proceeds to the calculation of the line opacity at each spectrum point. Spectrum must decide, at every spectrum point, which spectral lines to include in the calculation of the line opacity. To do this spectrum maintains two lists of lines. The first list includes all spectral lines in the line file which lie within 20Å (10Å in the optical) of the current spectrum point. For all the lines in this list, spectrum assigns a "computation radius" based upon the distance from line center at which the line depth drops below 0.0001 continuum units. If the computation radius of a line includes the spectrum point being currently computed, it is moved from the first list into a second list which comprises all of the lines which contribute to the line opacity at that point. Once SPECTRUM has stepped entirely through the computation radius of a line, that particular line is dropped from the second list as well.

For each line in the second list, the abundance of the ion in question and the level population for that particular transition are computed using the Saha and Boltzmann equations. The broadening parameter is then computed; broadening mechanisms include natural, van der Waals (see specifics in § 3.7) and quadratic Stark broadening (see, however, hydrogen and helium lines below). The line opacity at that point (for each of the levels in the atmosphere) is then computed using the Voigt function. The line opacity, line optical depth, continuum opacity and continuum optical depth are then included in a "line contribution" calculation which, when ratioed with the continuum flux, leads to the calculation of the normalized residual intensity at that spectral point. Spectrum calculates in its default mode the disk-integrated spectrum. This quantity, along with the wavelength is then written to the output file. As we saw above, this output can be modified by specifying certain SPECTRUM flags.

The hydrogen lines are computed by a routine adopted from Deane Peterson, used in the spectral synthesis code SYNTHE (Kurucz, 1993), which includes Stark and resonance broadening and fine structure in the core, as well as the Lyman-α quasi H_2 satellites. To save time, the full calculation of the hydrogen-line opacity is performed only at 1Å intervals, except within 5Å of the line core. The line opacity at intermediate spectral points is then obtained by 4 point interpolation (performed at each of the layers in the atmosphere). Hydrogen lines are included in the line opacity calculation for all points nominally within 2000Å of the line center (although this is modified in a complicated sort of way for the crowded lines in the vicinity of the Lyman, Balmer and Paschen convergences to avoid spurious line opacity in that region). The code for SPECTRUM includes 130 Lyman lines, 250 Balmer lines, 246 Paschen lines, 30 Brackett lines, 29 Pfund lines and 28 Humphreys lines. The computation for the Balmer jump is quite complex and includes computation of the lowering of the ionization energy using the Debye approximation, overlap of energy levels near to the continuum, etc. This more correct computation of the Balmer jump is also applied to the Paschen and Lyman convergences, but not yet to the Brackett, Pfund and Humphreys convergences.

The He I lines are computed on the basis of new line profile calculations of Beauchamp et al. (private communication & Beauchamp et al., 1997). These line profiles yield very accurate reproductions of the He I lines, including forbidden components. He II lines are currently not included in SPECTRUM, but will likely be included in an upcoming version upgrade. He I lines are computed only for models with $T_{\rm eff} > 8500$K. We note that certain helium lines in the red – especially 5875Å and 6678Å – are strongly affected by non-LTE and are thus not well reproduced by SPECTRUM.

Certain very strong lines (such as the Ca II K and H lines), and a handful of lines for which SPECTRUM's "calculation radius" is not adequate are not included in the `luke.lst` file, as they are calculated in a special routine. This routine allows a larger computation radius for those lines. This routine for strong lines might include, in the future, individualized source functions for certain strong lines, or even a more sophisticated calculation based on a multi-level atom.

3.9 Other Considerations

3.9.1 Model Atmosphere Selection

It is beyond the scope of this document to describe how to select an appropriate stellar atmosphere model for the analysis of a particular star, but the following comments should be of assistance. A stellar atmosphere model is characterized by 4 basic parameters, the effective temperature ($T_{\rm eff}$), the surface gravity ($\log g$), the overall metal abundance, [M/H], and the *microturbulent velocity*, ξ_t. These four parameters must also be specified to enable SPECTRUM to carry out its calculations. Of these four parameters, $T_{\rm eff}$ is certainly the most important, but the others cannot be ignored. There are many techniques that have been used in the literature to estimate the $T_{\rm eff}$. A carefully determined spectral type can yield the $T_{\rm eff}$ to an accuracy of ±100K, at least in the spectral-type range A – M. In addition, the *process* of spectral classification can reveal things about a star which you will need to know before carrying out a fine analysis – for instance, whether or not the star is chemically peculiar or deviates from the norm in any way. The $T_{\rm eff}$ may also be estimated from photometry; the infrared flux method (IRFM) is very useful here; Blackwell & Lynas-Gray (1994) have published a calibration that yields $T_{\rm eff}$ from the $V - K$ index. Other photometric calibrations can be found in the literature. D.F. Gray (Gray, 2008) has pointed out certain line ratios which may be used to estimate $T_{\rm eff}$. The surface gravity is more difficult to determine accurately; in the A and F-type stars the strength of the Balmer jump (measured by the c_1 index in Strömgren photometry, and less accurately by $U - B$ in Johnson photometry) is a sensitive measure of the gravity. When carrying out a fine analysis, an incorrect gravity will yield different abundances from different ionization states – for instance, if the iron abundance is different for Fe I and Fe II lines, the likelihood is that the gravity is wrong. Again, an accurate spectral type can be used to estimate the gravity from the luminosity type. For the A, F and G-type stars the paper by Gray, Graham & Hoyt (2001) may be useful in this regard. The determination of accurate abundances depends critically on the microturbulent velocity. The best way to proceed is to determine the abundance of a major species – say Fe I – simultaneously with the microturbulent velocity via a Blackwell diagram (see § 4.4). Using a model atmosphere with a microturbulent velocity as close as possible to the value determined via the Blackwell diagram is important for precise work, as the microturbulent velocity affects the line strengths, hence the line opacity, which in turn affects the structure of the atmosphere itself.

3.9.2 The Line-Spread Function and the Comparison of Synthetic and Observed Spectra

SPECTRUM computes the synthetic spectrum at very high resolution (for instance, a spacing of 0.01Å at 5000Å corresponds to a resolution of $R = 500,000$). Very few spectrographs yield this sort of resolution, and thus the very high resolution synthetic spectrum must be "smoothed" or convolved with a line-spread function (usually a Gaussian or near-Gaussian) to match the resolution of the spectrograph. The line-spread function is generally a function of the wavelength. This line-spread function can be determined from careful measurement of the emission lines from a comparison lamp, but often is adequately determined by trial and error. The SPECTRUM auxiliary program SMOOTH2 may be used to convolve a Gaussian of specified width with the output of SPECTRUM. Some spectrographs have a line-spread function with more extensive wings than a Gaussian. The program CUSTOMSM provides a line-spread function that is a hybrid of a Gaussian and a Lorentzian profile. These two programs are discussed more extensively in § 4.11 and § 4.6.

Spectral line profiles may also be affected by stellar rotation, *macroturbulence* and other effects. The programs VSINI and AVSINI may be used to apply rotational broadening to a synthetic spectrum. Like rotation, macroturbulence broadens a spectral line but, unlike microturbulence, does not strengthen it. Thus rotational broadening and macroturbulent broadening can be applied, at least to first order, *after* the synthetic spectrum has been computed by convolving the spectrum with the appropriate line-spread functions. The program MACTURB may be used to apply macroturbulent broadening. In practice, Gaussian smoothing, rotational broadening *and* macroturbulent broadening together will probably be needed to adequately match the line profiles in an observed spectrum. The exact parameters to use in these convolutions must generally be determined by trial and error. The macroturbulent velocity and the rotational velocity should be independent of the wavelength. As mentioned above, the resolution is generally dependent on the wavelength. Logically, macroturbulent and rotational broadening should be applied first, and Gaussian smoothing last.

Many stars, including the sun, show somewhat asymmetrical line profiles due to velocity fields in the photosphere. Such asymmetries can only be accounted for in a fundamental way by carrying out detailed 3D hydrodynamical calculations of the stellar atmosphere.

3.9.3 Further Practical Notes on Computing Synthetic Spectra

What wavelength spacing should be used in computing a synthetic spectrum? For detailed comparison with a high-resolution spectrum, it is advisable to compute the synthetic spectrum with a very fine spacing, 0.01 or even 0.005Å. For synthetic spectra which will be smoothed to match medium or classification-resolution spectra (∼ 1Å resolution), a spacing of 0.02Å is adequate. Finally, for computing flux spectra over a broad wavelength range, to be smoothed to a low resolution, a spacing of 0.05Å may be used without noticeable degradation in the final smoothed spectrum.

The linelist file `luke.lst` is adequate for computing spectra which will be smoothed to medium or low resolutions. However, for more detailed work, it is necessary to take into account isotope shifts and hyperfine structure. The linelist file `luke.iso.lst`, available on the distribution website, is thus recommended as a starting point for detailed work.

3.10 What is New in SPECTRUM v2.76?

There are four major improvements in v2.76 compared with the last major release, v2.75d. These improvements are:

1 The addition of AlH and AlO to the list of molecules that can be computed with SPECTRUM. Note that this has required a revision of `stdatom.dat`, which is now incompatible with earlier versions.

2 The addition of a facility to compute autoionizing lines (see § 3.7.2).

3 The addition of a facility to input a “fixed abundance” file containing elemental abundances that will not be scaled by the [M/H] of the model.

4 A significant improvement in the calculation of the “computation radius” for spectral lines (see § 3.8). Earlier versions used a rough and ready empirical estimate for the computation radius, which occasionally resulted in line wings being truncated. Various work arounds were used in these earlier versions to avoid these truncations, but in this version, the line profile is calculated until the depth falls below 0.0001 continuum units. Weak lines with central depths less than 0.0001 continuum units are thus not computed. This strategy has negligible effects on the line blanketing and thus the flux, and has the advantage that the computational speed is greatly enhanced. In regions of high line density, such as the ultraviolet, speed increases up to factors of 2.5 and more can be seen.

Chapter 4

Spectrum Auxiliary Programs

4.1 Abundance

The program ABUNDANCE can be used to calculate elemental abundances from equivalent widths. The necessary inputs are a stellar atmosphere model for the star being analyzed, the atomic and molecular datafile `stdatom.dat` and a linelist file with equivalent widths. The linelist file has the following format:

```
6127.895   26.0   33413   49727   -1.368  1.000 99 39.6 KFE
6165.361   26.0   33413   49628   -1.45   1.000 21 37.4 NIST
6232.635   26.0   29469   45509   -1.100  1.300 99 72.6 KFE
```

This linelist file has the same format as the linelist file for SPECTRUM except for one extra column – the penultimate column – which contains the equivalent width of the line in units of mÅ. Note that the seventh column contains the transition type; all the transition types valid for SPECTRUM are also valid here (with the exception of `AI`), including the `AO` and the `GA` types. If the `AO` type is used, then the next column is used for the σ and α parameters, in the packed format as explained in the documentation for SPECTRUM; the equivalent width will still be entered in the next-to-last column. Likewise for the `GA` transition type – see § 3.7.

Only the "t" switch may be used with ABUNDANCE (see § 3.1). This implies that isotopes are not supported. Likewise, while hyperfine components may technically be analyzed with ABUNDANCE it would be necessary to measure equivalent widths for each hyperfine component. Abundances based on lines that show significant isotope splitting or broad profiles due to hyperfine structure are best analyzed with full spectral synthesis.

The output of ABUNDANCE has the following format:

```
6127.895   26.0  1.08   -4.581    7.459  -0.041
6165.361   26.0  1.08   -4.550    7.490  -0.010
6232.635   26.0  1.08   -4.581    7.460  -0.041
```

The meaning of the first and second columns are obvious, the third column is the microturbulent velocity (in km/s) input into ABUNDANCE (it prompts the user for this quantity). The fourth column gives the abundance on the scale native to SPECTRUM (see § 3.4), and the fifth on the normal scale, in which the logarithmic abundance of hydrogen is equal to 12.0. The final column is the abundance *relative* to the *unscaled* abundances in `stdatom.dat`; since the `stdatom.dat` abundances are solar abundances, the final column gives the quantity [X/H] where X is the species in question. Note that if you are using a custom atomic and molecular data file, this may not necessarily be the case.

4.2 Avsini

The program AVSINI may be used to apply rotational broadening to a synthetic spectrum. Avsini will prompt the user for the required inputs, or it may be used in a command line mode. The format for invoking AVSINI in the command line mode is

```
avsini input output vsini u dlambda
```

where `input` is the filename of the synthetic spectrum you wish to rotationally broaden, `output` is the name of the rotationally broadened spectrum, `vsini` is the rotational velocity, in km/s, `u` the limb-darkening coefficient (0.6 is a good value to use here, but see Gray, 2008) and `dlambda` is the wavelength spacing in the input file. The spacing in the output file will be the same as the spacing in the input file. If used in the prompt mode, the AVSINI prompts are in the same order. Rotational broadening should be carried out before the spectrum is smoothed to a lower resolution with SMOOTH2. Avsini may be used either before or after macroturbulent broadening is applied (see below). Avsini is designed to be used with ascii synthetic spectra. The program VSINI is the equivalent to be used with binary synthetic spectra. Both programs use David Gray's (Gray, 2008) implementation of rotational broadening.

4.3 Bflxsm2

The program bflxsm2 convolves a Gaussian line-spread function with a synthetic spectrum and is designed to work with a binary format synthetic spectra computed with the “f” switch (i.e. in flux format). The equivalent for ascii files is smooth2. The purpose of this program is to match the resolution of the synthetic spectrum to an observed spectrum so that detailed comparisons can be made. A similar program is Bsmooth2, which is designed to work with a binary spectrum computed in the normalized intensity mode. They aren’t very different. Bflxsm2 can be used either in the prompt mode or in the command line mode. In the command line mode, the format is

```
bflxsm2 input output resolution dlambda
```

where `input` is the name of the input binary synthetic spectrum, `output` is the name of the smoothed synthetic spectrum, `resolution` is the resolution in Å of the smoothed spectrum, and `dlambda` is the wavelength spacing of the *output* spectrum; bflxsm2 picks up the wavelength spacing of the input spectrum from the header of the binary file. An important note: the spacing of the output spectrum should be an *integer multiple* of the spacing of the input spectrum, or weird things will happen!

4.4 Blackwel

Blackwel computes the data required for the plotting of a “Blackwell diagram” (see Blackwell & Shallis, 1979) for a set of spectral lines of a given species with measured equivalent widths. Blackwell diagrams are used to determine the abundance of an element simultaneously with the microturbulent velocity for a star. That is to say, Blackwell diagrams give the same information as the classical curve of growth method, but without the debilitating effects of the many invalid assumptions inherent to that method. A Blackwell diagram is a plot of the abundance versus the microturbulent velocity (see Figure 4.1); for each line in the set, the abundance required to yield the measured equivalent width is computed as a function of the microturbulent velocity. Each spectral line is thus represented as a curve in the diagram; for weak lines on the linear part of the curve of growth, the curve is nearly horizontal, for stronger lines, the curves slope more steeply. Ideally, all of these curves should intersect at a single point which would then give the elemental abundance and microturbulent velocity simultaneously.

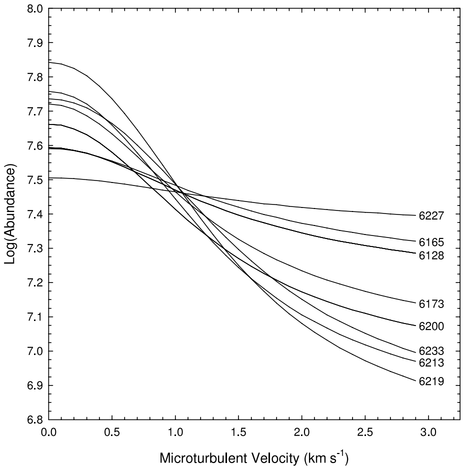


Figure 4.1: A Blackwell diagram computed with BLACKWEL for a set of Fe I lines in the star HD 26367 (Gray & Griffin, 2007).

In practice, there is a region of "least confusion", from which the elemental abundance, the microturbulent velocity and the associated errors may be estimated. The microturbulent velocity so determined should be the same, within the errors, for all species. So, a practical way to proceed would be to construct a Blackwell diagram for a species well represented by a large number of measurable lines (say, Fe I), and the resulting microturbulent velocity considered as a fixed quantity in the calculation of other elemental abundances using a program such as ABUNDANCE.

The set of spectral lines used to construct the Blackwell diagram should be placed in a file in which each line has the following format (identical to the format used for ABUNDANCE and GFADJUST):

```
6165.361   26.0   33413   49628   -1.45   1.000 21 37.4 NIST
```

Here the wavelength, code, E(lower), E(higher), log(gf), the damping factor, and the transition type are in exactly the same format as in luke.lst. The equivalent width (the second to last entry), derived from a measurement in a stellar spectrum, is expressed in milli-angstroms (mÅ). All of the broadening codes used in SPECTRUM are also valid in BLACKWEL. An example file `feI.eqw` is provided in the distribution. It can be used in conjunction with the model 67244k2p00.mod to test BLACKWEL. Note that BLACKWEL does not support isotopes. A line with measurable hyperfine structure is not suitable for use in constructing a Blackwell diagram.

The output from BLACKWEL has the format:

```
wavelength  code  vt  Atot  AH
```

where `vt` is the microturbulent velocity, `Atot` is the abundance of the element represented by `code` on the abundance scale used by SPECTRUM, whereas `AH` is the abundance scale based on hydrogen. See § 3.4 for more information on these abundance scales. BLACKWEL will prompt the user for input and output files, for a range of microturbulent velocities and for a microturbulent velocity stepsize. Thus, each spectral line in the input file results in a number of lines in the output file. This output file can be used to plot a Blackwell diagram. The best lines to use to construct a Blackwell diagram are isolated (non-blended), weak to moderate-strength lines on the linear and flat parts of the curve of growth. Strong lines, especially those on the damping part of the curve of growth, are not appropriate for a Blackwell diagram.

The only “switch” that can be used with BLACKWEL is the “t” switch which alerts the program to the presence of an ATLAS9 header in the stellar atmosphere model. See § 3.1.

The distribution for SPECTRUM comes with a GUI for use with BLACKWEL (`blackwell_sh.tcl`). See § 5.2 for more information on this GUI and more practical information on how to construct and use Blackwell diagrams.

4.5 BSMOOTH2

The program BSMOOTH2 convolves a Gaussian line spread function with a binary synthetic spectrum computed in the normalized intensity mode. Its operation is identical to that of BFLXSM2 (see above).

4.6 CUSTOMSM

Some spectrographs do not have Gaussian line-spread functions, and so the programs BSMOOTH2, BFLXSM2 and SMOOTH2 will not give satisfactory results. In some cases, there is no substitute for writing your own smoothing program incorporating the measured linespread function of your spectrograph, especially if the function is asymmetrical. However, if you need a linespread function that has more “wing” than a Gaussian, you might find CUSTOMSM useful. This program convolves the synthetic spectrum with a hybrid linespread function made from a weighted mean of a Gaussian and a Lorentzian profile. The Lorentzian profile has much broader wings & a narrower core than a Gaussian. In addition to inputing the information required with SMOOTH2, CUSTOMSM needs a "line-shape parameter" which runs from 0 → 1. When that parameter is 0, the linespread function is a pure Gaussian; when it is 1 it is a pure Lorentzian. Experimentation is required to get the best fit. CUSTOMSM, like SMOOTH2, is only for ascii spectra. It can be used in both the prompt mode and the command-line mode. Use the command-line mode format:

```
customsm input output input_spacing resolution output_spacing line_shape
```

4.7 GFADJUST

GFADJUST is a program, based on SPECTRUM code, that can be used to *invert* a stellar spectrum to determine oscillator strengths (*gf values*) for individual spectral lines. The *gf* values used in the linelist file `luke.lst` come from a number of sources. Some have been determined in the laboratory, others have been derived from theoretical calculations. The accuracy of these *gf* values varies widely from line to line; some are known to accuracies of 1% or better, others are off by an order of magnitude. The accuracy of the *gf* values is certainly one of the most important factors in determining whether or not a spectral synthesis calculation reproduces faithfully the spectrum of a star. Other factors include the accuracy with which the basic physical parameters have been determined for the star in question, the quality of the broadening parameters and theory, the validity of the LTE approximation, etc.

Within the limitations of those other factors, GFADJUST may be used to determine *gf* values from measurement of the spectrum of a well-studied star such as the sun. To determine the *gf* value for a particular spectral line, its *equivalent width* must be measured and included in the input linelist for GFADJUST. The format of this line list is identical to the ones used for both ABUNDANCE and BLACKWEL. GFADJUST also requires as input a stellar atmosphere model for the star in question and a microturbulent velocity. The output from GFADJUST has the following format:

```
Wavelength    Code    Log10(gf)
```

In addition to the caveats above, care must be taken to ensure that the spectral lines for which you are determining the *gf* values are unblended, or that you have adequately separated out the components of the blend. GFADJUST does not support isotopes, nor should it be used with lines with measurable hyperfine structure. Likewise, GFADJUST should not be used to determine the *gf* values for strong lines. Only the "t" switch may be used with GFADJUST(see § 3.1).

4.8 Lines

The program LINES may be used to calculate the equivalent width and the central depth of the lines in a linelist file. This is useful for identifying lines, and also for figuring out the components in a complex blend of lines. But it can also be used for creating a line list file based on the subset of lines in the input linelist file that have equivalent widths greater than a specified value. This can be useful for speeding up a spectral synthesis by eliminating those lines that do not contribute significantly to the spectrum. Lines fully supports isotopes and hyperfine structure, and the full range of line broadening theory in SPECTRUM. The format of the input linelist file is identical to that of SPECTRUM. The inputs are also the same as SPECTRUM (a stellar atmosphere model, the atomic and molecular data file `stdatom.dat`, and optionally the isotope file `isotope.iso`). Lines will also prompt the user for the spectral range, and the minimum equivalent width for a line to be included in the output. The default output has the following format:

```
4190.243   22.1    22.387  0.5531
4190.408   23.1     9.332  0.7928
4190.459  106.0     3.026  0.9610
```

where the first column is the wavelength, the second the species code, the third the computed equivalent width (units mÅ), the fourth the line depth (with the continuum at 1.0, and assuming infinite resolution with no macroturbulence or rotation). Thus, the core of the third line is only 4% below the continuum. The default (and only mode) is to calculate the equivalent width and line depth for the integrated disk.

The following switches may be used with LINES:

C: Will cause LINES to print out a file `cd.out` which contains the "contribution function" of the lines in the linelist (note – it is best to invoke this for a linelist that has only a few lines in it). The output file has three columns; the first is the layer number in the stellar atmosphere, beginning with 1 for the top layer, the second is the reference continuous optical depth (at 5000Å) and the third is the contribution function used in calculating the line profile.

i: Invokes the isotope mode – see § 3.1.

s: Causes LINES to output a linelist file (in exactly the format of the input linelist file) containing those lines that have equivalent widths greater than the minimum equivalent width entered as a prompt to the program. This option is useful for culling linelists of weak lines which do not contribute to the spectrum and would simply slow down a spectral synthesis calculation.

t: Informs LINES that the stellar atmosphere model is in the ATLAS9/ATLAS12 format – see § 3.1.

v: Verbose mode – the default mode of LINES is to be silent – i.e. to print nothing to the screen beyond a few initial messages. This switch causes LINES to print to screen everything it prints out to the default output file.

4.9 MACTURB

Macroturbulence is defined as a scale of turbulence in the stellar atmosphere in which the size of the turbulent cell is greater than the mean-free path of the photon. This is in contrast to *microturbulence* in which the size of a turbulent cell is smaller than the mean-free path. With microturbulence, a photon will tend to be emitted in one turbulent cell and absorbed in another; relative motions between these turbulent cells will cause a photon emitted in line center to be absorbed in the line wing. As a consequence, microturbulence both broadens and strengthens the line, as it leads to a desaturation of the line core. The effects of microturbulence are fully taken into account in the spectral synthesis calculation. This is not the case with macroturbulence. Macroturbulence will broaden the spectral line, but not desaturate it, and so it has no effect on the strength of the line. However, it is necessary to apply macroturbulent broadening to the computed spectrum when making a detailed comparison between the synthetic spectrum and a high-resolution observed spectrum. The program MACTURB implements the macroturbulent broadening formulation of David Gray (Gray, 2008). MACTURB requires as input the name of the spectral synthesis file, an output file name, the wavelength spacing in the input file and the macroturbulent velocity. The output wavelength spacing will be the same as the spacing in the input file. MACTURB may be used in the command line mode thus:

```
macturb input_file output_file spacing velocity
```

MACTURB operates only on ascii synthetic spectra. There is no equivalent (yet) for SPECTRUM binary files. Macroturbulent and rotational broadening should be applied before the spectrum is smoothed with SMOOTH2. The macroturbulent velocity must be determined by trial and error, or via the techniques outlined in David Gray's book. Typically, the macroturbulent velocity is on the order of a few km/s.

4.10 SELECT

The program SELECTMOD may be used to extract individual models from Kurucz "supermodels" such as the file `ap00k2odfnew.dat` which can be obtained from Robert Kurucz's website `http://kurucz.harvard.edu`. SELECT extracts a single model from a supermodel for a given T_{eff} and $\log g$. The model grid in the more recent supermodels has a T_{eff} spacing of 250K (beginning with 3500K) up to 13,000K and a spacing of 1000K after that (up to 50,000K), and a $\log g$ spacing of 0.5, beginning on the high end at 5.0. The program SELECTMOD will extract one of these models and export it to a file with the default SPECTRUM header. The program SELECTMOD9 does the same thing, but preserves the ATLAS9 header. Both can be invoked in the command line mode:

```
selectmod supermodel.dat output.mod teff logg
```

4.11 SMOOTH2

SMOOTH2 is a program similar in action to BSMOOTH2, except that it operates on ascii spectra (the normal, default output of SPECTRUM). It convolves a Gaussian linespread function with the synthetic spectrum to smooth the spectrum to a desired resolution. It can be used either in the prompt mode or in the command-line mode. In the command line mode the format is:

```
smooth2 input_file output_file input_spacing resolution output_spacing
```

where `input_spacing` is the wavelength spacing in angstroms of the input spectrum, and `output_spacing` is the wavelength spacing of the output smoothed spectrum. Note that `output_spacing` must be an integer multiple of `input_spacing` or funny things will happen! The resolution should be the resolution in angstroms desired for the output spectrum. Sometimes a more versatile linespread function is required to help match the linespread function of a given spectrograph. In that case, the program CUSTOMSM can be helpful.

4.12 VSINI

VSINI is the binary-file equivalent of AVSINI. It may be used in both the prompt mode and the command-line mode. The command-line mode format is

```
vsini input_file output_file vsini(km/s) limb_darkening_coeff output_spacing
```

Chapter 5

Graphical User Interfaces

Graphical user interfaces using the Tcl/Tk Tool Command Language and graphical interface have been designed for both SPECTRUM and BLACKWEL. These GUIs are designed to make using SPECTRUM and BLACKWEL more easy and to integrate SPECTRUM with some of its auxiliary programs. The Tcl/Tk GUIs require that the Tcl/Tk toolkit be installed on your UNIX/Linux computer. To check to see if it is installed, type `wish` at the prompt. If a small, grey empty window appears on your screen, it is installed. If not, you should be able to install it via a package manager, or from the website `http://www.tcl.tk/software/tcltk/`. A Windows-based GUI has also been designed for SPECTRUM.

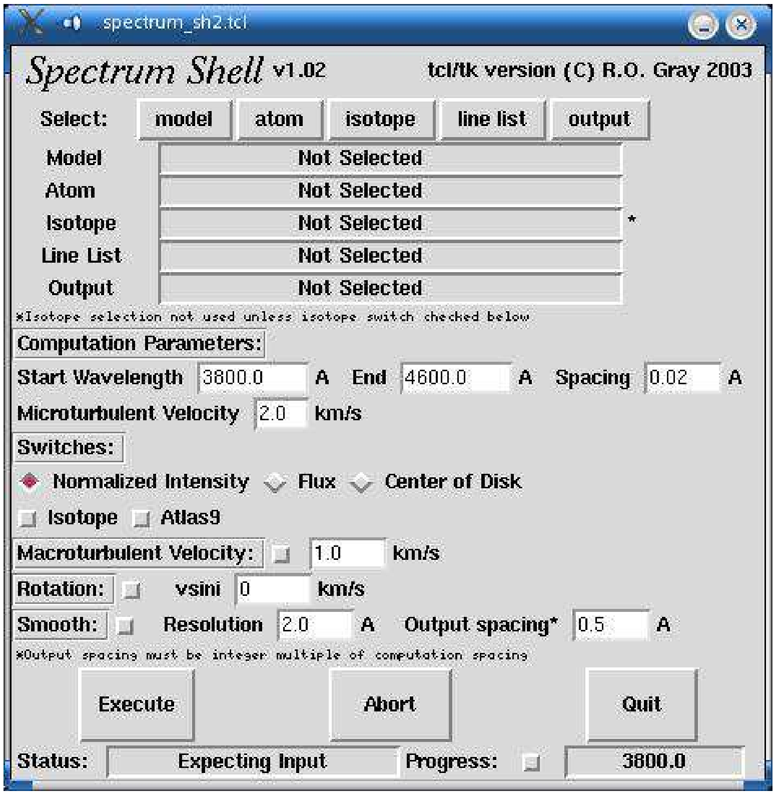
5.1 SPECTRUM GUI

The script `spectrum_sh.tcl`, which should be in your SPECTRUM distribution, is a Tcl/Tk graphical user interface for SPECTRUM. If you ran the `make install` script in § 2.4 the script spectrum_sh.tcl should have been placed in `/usr/local/bin`. If you can't run it, it may be necessary to get root permission and then to change the mode so that the file is executable.

This shell may be invoked by typing at the prompt:

```
> spectrum_sh.tcl
```

The GUI will come up and should look like Figure 5.1. The buttons on the top row can be used to select the atmosphere model, the atomic data file (`stdatom.dat`), the isotope definition file (`isotope.iso`), the spectral linelist and the output file. The computation parameters are self-explanatory and must be supplied. The shell enables the user to apply a number of



Spectrum Shell v1.02 tcl/tk version (C) R.O. Gray 2003

Model: Not Selected

Atom: Not Selected

Isotope: Not Selected *

Line List: Not Selected

Output: Not Selected

*Isotope selection not used unless isotope switch checked below

Computation Parameters:

Start Wavelength 3800.0 A End 4600.0 A Spacing 0.02 A

Microturbulent Velocity 2.0 km/s

Switches:

● Normalized Intensity ○ Flux ○ Center of Disk

□ Isotope □ Atlas9

Macroturbulent Velocity: □ 1.0 km/s

Rotation: □ vsini 0 km/s

Smooth: □ Resolution 2.0 A Output spacing* 0.5 A

*Output spacing must be integer multiple of computation spacing

Status: Expecting Input Progress: □ 3800.0

Figure 5.1: The Tcl/Tk GUI for SPECTRUM.

switches which control the operation of SPECTRUM. Please read § 3.1 to understand the meaning of these switches. On the first row, the default output (Normalized Intensity), absolute flux, or center of disk switches can be set (corresponding to: no switch, f and m, respectively). On the second row, the isotope mode may be turned on (the isotope definition file must have been selected above) and/or the facility to read ATLAS9 default headers (switch t) can be selected. If selected, the output spectrum may then be smoothed using MACTURB (which convolves the spectrum with a macroturbulent velocity line profile), AVSINI, which rotationally broadens the output spectrum, and SMOOTH2, which convolves the output spectrum with a Gaussian function. To execute, press the Execute button. To see the progress as SPECTRUM steps through the wavelength region, click the Progress check box. Pressing Quit will exit the shell.

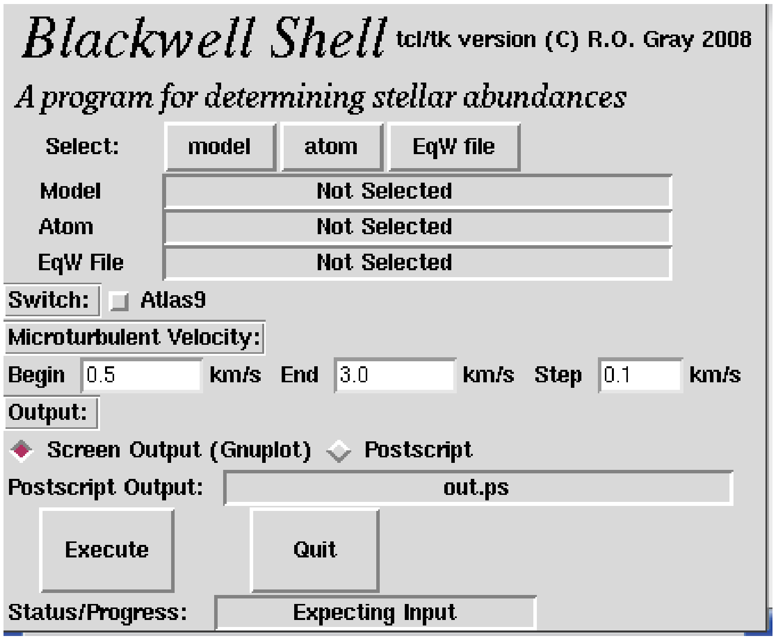
Please note that this shell is not absolutely foolproof. Please pay very close attention to the “*”’ed boxes - i.e. the isotope box and the output spacing box for the Smooth option.

5.2 Blackwel GUI

Before attempting to use this shell, please read over the documentation for BLACKWEL. This shell is included in the SPECTRUM distribution and is called `blackwell_sh.tcl`. It likewise will have been moved to `/usr/local/bin` if you executed the `make install` script discussed in § 2.4. To invoke this shell, type at the prompt:

```
> blackwell_sh.tcl
```

The GUI illustrated in Figure 5.2 should appear. To run this shell, you must have installed on your machine the latest version (4.0 or later) of gnuplot (see `http://www.gnuplot.info/`).



Blackwell Shell tcl/tk version (C) R.O. Gray 2008

A program for determining stellar abundances

Model Not Selected

Atom Not Selected

EqW File Not Selected

Switch: □ Atlas9

Microturbulent Velocity:

Begin 0.5 km/s End 3.0 km/s Step 0.1 km/s

Output:

● Screen Output (Gnuplot) ◇ Postscript

Postscript Output: out.ps

Status/Progress: Expecting Input

Figure 5.2: The Tcl/Tk GUI for BLACKWEL.

To test this shell, select the stellar atmosphere model `67244k2p00.mod` provided in the distribution, the atomic data file (`stdatom.dat`) and an equivalent width file for a single species (the file `feI.eqw` is provided for this test). Select a range of microturbulent velocities (say 0.5 to 3 km/s) over which you want BLACKWEL to do its computations and then press Execute. For this set of files, you will get the graph in Figure 5.3, courtesy of `gnuplot`.

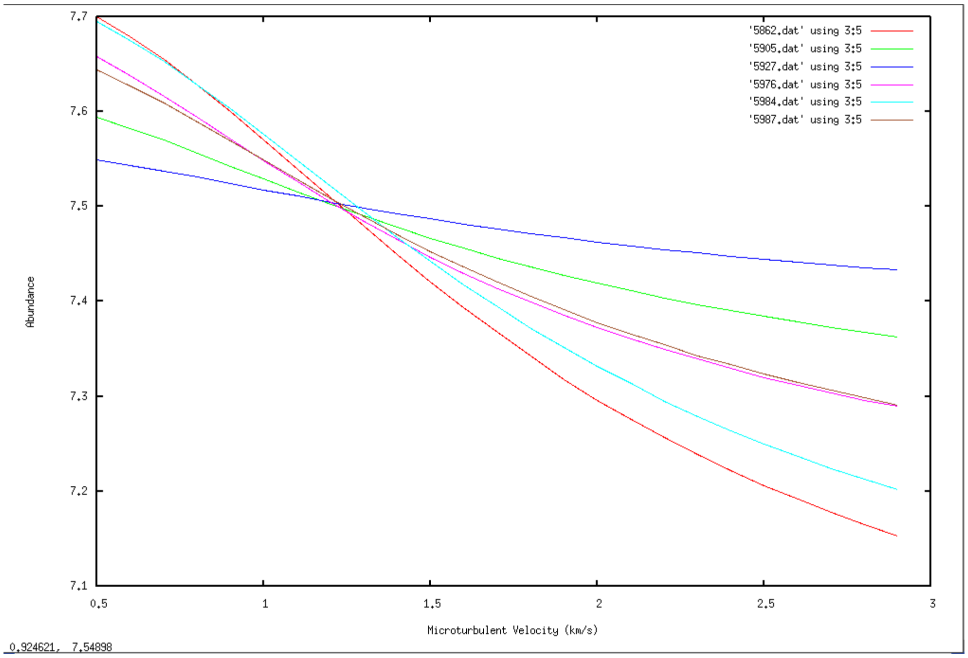


Figure 5.3: Sample output from the BLACKWEL GUI.

Each sloping line in this graph corresponds to a spectral line in the file `feI.eqw`. This is a plot of the abundance of iron as determined from Fe I lines against the microturbulent velocity. Notice that (somewhat ideally!) all of the lines seem to converge at a point in the plot. This gives the optimal value for the abundance of iron and the characteristic microturbulent velocity in the atmosphere of the star under consideration. If you have chosen the temperature and gravity of the star correctly, you will get nearly the same answer for Fe II lines. And, for other species, you should get, within the errors, the same value of the microturbulent velocity. The figure above is an example of a "Blackwell Diagram", which is a useful tool for determining microturbulent velocities and abundances in a consistent way in stellar atmospheres.

Please note that the lines can fail to converge in a Blackwell diagram if 1) one or more of your lines is in an undetected blend, 2) the *gf* values for one or more of your lines are not good, 3) your choice of a stellar atmosphere to model your star is not good, or 4) your line is strongly affected by non-LTE effects. You should use a stellar atmosphere computed with a microturbulent velocity which is as close as possible to the value indicated by the Blackwell diagram, for the sake of consistency. The model atmosphere `67244k2p00.mod` was computed with a microturbulent velocity of 2km/s. Since the diagram above indicates that the microturbulent velocity is closer to 1.2 km/s, it would be better to select a model computed with a microturbulent velocity of 1 km/s.

Please note that BLACKWEL cannot be used with hydrogen or helium lines (both He I and He II). Ideally, lines used to compute Blackwell diagrams should be weak to medium strength lines. The cores of strong lines are often not well reproduced in the LTE approximation. A range of line strengths is required in a Blackwell diagram. The weak lines (which are not strongly affected by microturbulence) are nearly horizontal in a Blackwell diagram. Stronger lines have a steeper slope.

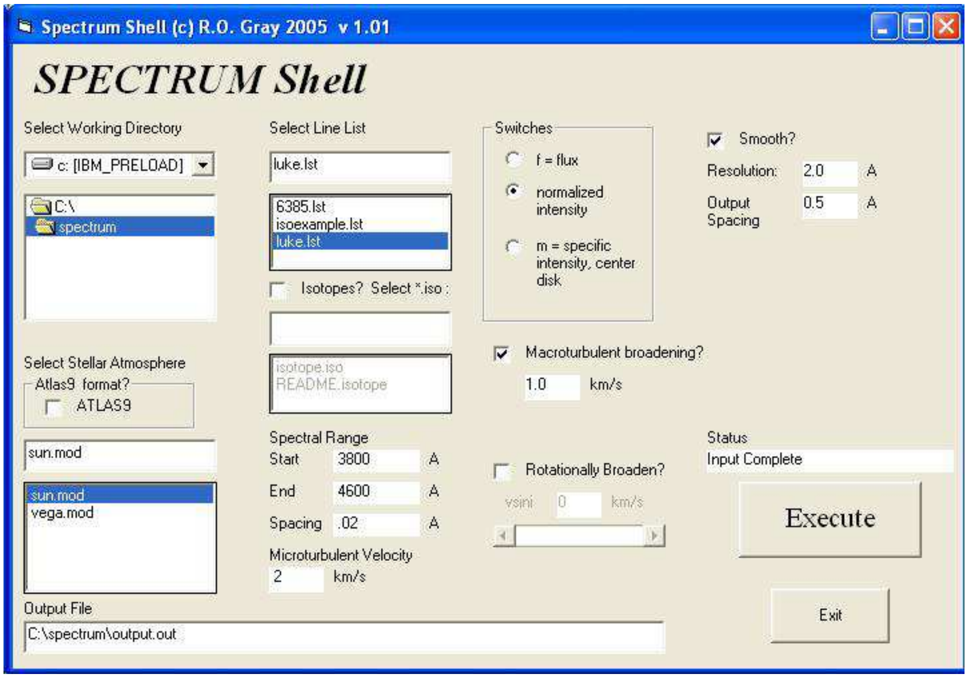
Note that `blackwell_sh.tcl` outputs data you can use to reconstruct the above graph in individual data files labeled with the wavelength of the line.

The shell `blackwell_sh.tcl` can be directed to give the above plot in postscript format.

5.3 A Windows GUI

To make SPECTRUM easier to use for those Windows users who are not conversant with command-line computing, and with pedagogical purposes in mind, I have written a GUI shell for SPECTRUM using Visual Basic. It is illustrated in Figure 5.4. With this shell, it is possible to compute a synthetic spectrum, apply macroturbulent and rotational broadening and smooth the resulting spectrum with just a few mouse clicks.

To install this shell, download the file spectrum101.zip and place it in the same folder you used to install SPECTRUM in § 2.2. I recommend c:\spectrum; I have had problems if I installed this under “My Documents”, as the path name was apparently too long for Visual Basic. Unzip the file into this directory; you will find the files `setup.lst, setup.exe` and `spectrum101.cab`. Double click on `setup.exe`. This will run the install program for the shell; it may be necessary to reboot Windows before the program will completely install. When prompted, specify the directory where you have copied the other files (e.g. c:\spectrum) as the location for in-



Spectrum Shell (c) R.O. Gray 2005 v 1.01

SPECTRUM Shell

Select Working Directory: c: [IBM_PRELOAD]

C:\
spectrum

Select Line List: luke.lst

6385.lst
isoexample.lst
luke.lst

□ Isotopes? Select *.iso :

isotope.iso
README.isotope

Switches
○ f = flux
● normalized intensity
○ m = specific intensity, center disk

☑ Smooth?
Resolution: 2.0 A
Output Spacing 0.5 A

Select Stellar Atmosphere
Atlas9 format? □ ATLAS9

sun.mod

sun.mod
vega.mod

Spectral Range
Start 3800 A
End 4600 A
Spacing .02 A

Microturbulent Velocity 2 km/s

☑ Macroturbulent broadening? 1.0 km/s

□ Rotationally Broaden? vsini 0 km/s

Status: Input Complete

Output File: C:\spectrum\output.out

Figure 5.4: The Visual Basic GUI for the Windows version of SPECTRUM.

stalling the shell. When the install is completed, the shell can be run from the Run menu (click on the Windows Start button). You must first select the working directory (the folder where all the other files reside). If you have carried out the above instructions, the proper working directory should already be selected. Then, select the stellar atmosphere from the list in the file list box, the linelist (e.g. `luke.lst`) and specify the spectral range, computational spacing, and, if you wish, whether the output spectrum will be broadened by rotation and/or macroturbulence and smoothed. Once all the program parameters have been specified, click on EXECUTE and sit back and relax. SPECTRUM will be launched in a minimized DOS window, which will disappear when all the calculations have completed. The output file will be in the working directory. It is an ascii file with wavelength in the first column and intensity or flux in the second. It can be displayed in any plotting program, such as EXCEL, SIGMAPLOT, etc.

Appendix

The SPECTRUM package may be picked up on the web at `http://www.phys.appstate.edu/spectrum/spectrum.html`. Be certain to check this site periodically to see if a new version has been released. Please also consider registering your use of SPECTRUM on that website.

The author of this document and SPECTRUM, Richard Gray, may be contacted at the following address
Department of Physics and Astronomy
Appalachian State University
Boone, NC 28608
`grayro@appstate.edu`

If you use SPECTRUM in your publications, please reference it by the paper Gray & Corbally (1994) and, if possible, by including the URL in a footnote.

Bibliography

Anstee, S.D. & O'Mara, B.J. 1991, MNRAS, 253, 549

Anstee, S.D. & O'Mara, B.J. 1995, MNRAS, 276, 859

Barklem, P.S. & O'Mara, B.J. 1998, MNRAS, 300, 863

Beauchamp, A., Wesemael, F. & Bergeron, P. 1997, ApJS, 108, 559

Blackwell, D.E. & Lynas-Gray 1994, AA, 282, 899

Blackwell, D.E. & Shallis, M.J. 1979, MNRAS, 186, 673

Dembczyński, J., Ertmer, W., Johann, U., Penselin, S., & Stinner, P. 1979, Z. Phys. A, 291, 207

Gray, D.F. 2008, "The Observation and Analysis of Stellar Photospheres", 3rd edition, (Cambridge: Cambridge University Press).

Gray, R.O. & Corbally, C.J. 1994, AJ, 107, 742

Gray, R.O., Graham, P.W. & Hoyt, S.R. 2001, AJ, 121, 2159

Gray, R.O. & Griffin, R.E.M. 2007, AJ, 134, 96

Grevesse, N. & Sauval, A.J. 1998 Space Science Reviews 85, 161

Kurucz, R. L. 1993, CD-ROM 13, ATLAS9 Stellar Atmosphere Programs and 2 km/s Grid (Cambridge: Smithsonian Astrophys. Obs.)

Luc, P. & Gerstenkorn, S. 1972, AA, 18, 209

Shore, B.W. 1967, Journal of the Optical Society of America, 57, 881

Shore, B.W. 1968, Phys. Rev., 171, 43