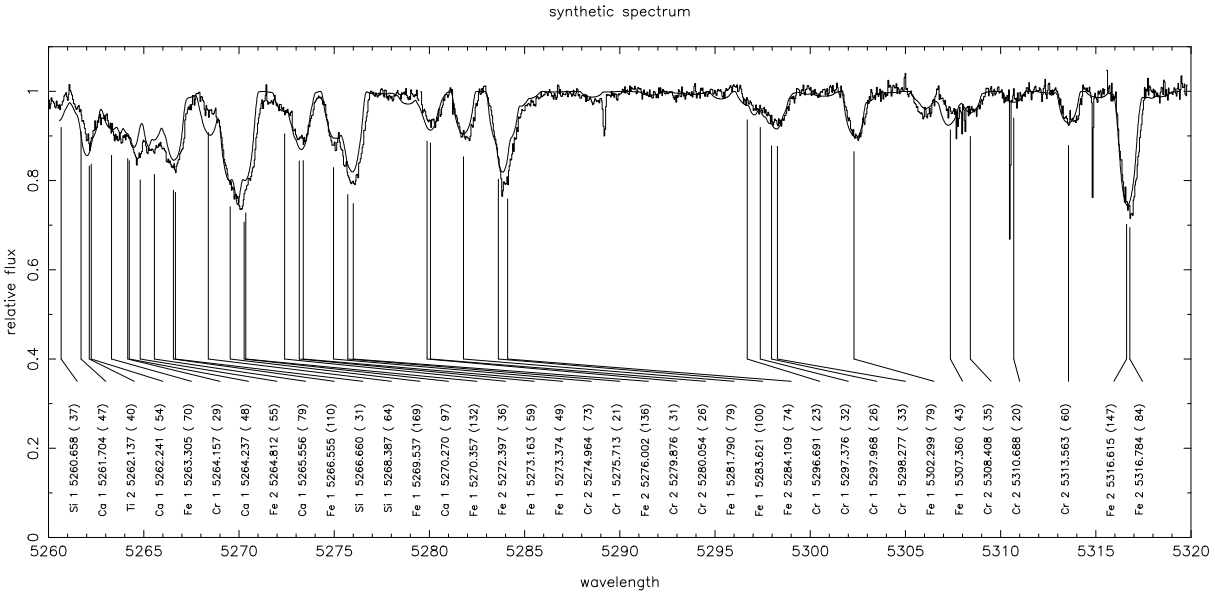
UCLSYN User Guide (v 4.0)
incorporating BINSYN and TELSYN

B. Smalley[1], K.C. Smith[2] & M. M. Dworetsky[2]

[1] *Department of Physics, Keele University, Staffordshire, ST5 5BG*

[2] *Department of Physics & Astronomy, University College London, Gower Street, London WC1E 6BT*




```
> *list 1 14
  no  id   lambda     ep  log(gf)  g(rad)   g(vdw) g(stark)   ew  log(A)
   1 Fe 2 5260.259 10.419   1.069  1.01E+1 -1.82E-8 -4.32E-6 0.016  7.540
   2 Ca 1 5260.387  2.521  -1.720  9.93E-1 -1.89E-8 -1.75E-6 0.007  6.360
   3 Si 1 5260.658  5.616  -0.920  1.00E+0 -2.10E-7 -9.99E-5 0.039  7.550
   4 Ca 1 5261.704  2.521  -0.591  9.96E-1 -1.89E-8 -1.75E-6 0.043  6.360
   5 Cr 1 5261.763  3.698  -0.542  6.61E-1 -3.72E-8 -9.96E-6 0.003  5.670
   6 Ti 2 5262.137  1.582  -2.106  2.06E+0 -1.12E-8 -1.94E-7 0.038  4.990
   7 Ca 1 5262.241  2.521  -0.491  9.98E-1 -1.88E-8 -1.75E-6 0.047  6.360
   8 Fe 1 5262.608  4.320  -2.028  2.35E+0 -1.70E-8 -1.72E-5 0.003  7.540
   9 Fe 1 5262.882  3.251  -3.173  7.60E-1 -1.47E-8 -4.90E-7 0.001  7.540
  10 Fe 1 5263.305  3.266  -0.915  9.36E-1 -2.41E-8 -3.07E-6 0.050  7.540
  11 Fe 1 5263.394  3.640  -2.774  1.73E+0 -1.47E-8 -9.92E-7 0.002  7.540
  12 Fe 1 5263.862  3.573  -2.111  2.04E-1 -1.36E-8 -4.99E-7 0.008  7.540
  13 V  2 5263.999  2.510  -2.368  2.87E+0 -1.22E-8 -2.54E-7 0.001  4.000
  14 Cr 1 5264.157  0.968  -1.288  6.58E-1 -1.53E-8 -7.59E-7 0.031  5.670
>
```

Contents

1 Introduction

UCLSYN is a converted version of the J.E. Ross Stellar Synthesis Code, which runs interactively on Starlink (see Smith, 1992). The program allows for the curve-of-growth determination of abundances or the evaluation of equivalent widths. The program can also be used to synthesize line profiles.

2 Setting up UCLSYN

2.1 Starting the code

To invoke UCLSYN the following statement must be set up:

- On a UNIX system add the following to the .login file:
  `source /packages/uclsyn/uclsyn_setup`
- On a LINUX system follow the instructions in the file `Install.Readme`, and take note of the requirement to have a legal version of the Numerical Recipes source codes. The program files must also be sourced in a similar manner but the name of the directory with the files may be different.

Upon logging in the messages `UCLSYN available` will be displayed. Any errors or problems with the startup procedure will be displayed. The programs can now be invoked by typing `uclsyn`. On running the code the following will appear on the screen:

```
 welcome to UCLSYN 4.0 (17 Dec 2003)
 incorporating BINSYN 1.0 and TELSYN 0.2-1
 reading initialisation files ...

  type "help" for list of commands available
  type "help summary" for one-line descriptions

  [other recent program notes will appear here]

>
```

The prompt '`>`' indicates that the program is waiting for a command. Section 4 gives a full list of available commands. Typing a '`?`' straight after a command, with no space in between, will cause all parameters for that command to be prompted for. A '`%`' as the first character of a line indicates that the line is a comment statement, which can be useful when writing command files (see `rdcmd`).

2.2 What's new?

Version 4.0 sees the incorporation of the extended VCS tables of Lemke ([31] and the Anstee, Barklem & O'Mara (ABO) VDW broadening parameters.

Now incorporates BINSYN which provides for two arrays (called "current" and "shadow") for two stars, permitting the curve-of-growth or synthesis options to be applied to binary star component spectra with designated doppler shifts, to simulate the blending and dilution effects found in binaries.

Inclusion of developmental version of telluric line synthesis facilities (TELSYN). Not fully integrated and only really a demonstration version.

Details of changes to UCLSYN can be found in the `CHANGES` file in the `docs` subdirectory. In the same directory, a `TODO` file indicates topics that should be addressed in future releases.

3 Basic outline of code and its internal buffers

3.1 line buffer

Contains absorption line data entered via `*atom` commands.

3.2 results buffer

Contains the results from `exact` and/or `error`.

3.3 errors buffer

Contains values of error estimates.

3.4 cloud buffer

Contains any current abundance cloud.

4 Description of commands

There now follows a description of the commands available. *Note that no "," should be typed in between parameters; these are given here to separate the parameter names.* Parameters in brackets are optional. If commands are executed out of sequence (e.g. `synth` before `grid`, `pgplot` before `pgdev`, etc.), an error message is displayed and the command aborted.

When a command is excuting, various messages are displayed informing the user of what the program is doing and if any errors are encountered. The following symbols have special meaning:

`?` indicates that the program requires the user to input a value for a parameter.

`!` informs the user of an error which caused the command to be aborted or a warning that part of the command could not be executed for some reason.

`...` indicates that the program is performing certain operations that may take a few seconds or more to execute.

`*absum`

Produces a table of mean abudances and errors for all elements within the current results buffer.

If a plotting device is available, a plot of the abundance pattern (relative to solar) is generated.

Under Development

`*abund` element, log A

Sets the abundance of all the lines of the given element to log A. The convention in UCLSYN is that abundances are given on a log scale where log A(H) = 12.00.

Clears results and errors.

`*atom` element, ion stage, wavelength, ep (eV), log gf, gamma(nat), gamma(vdw), gamma(stk), log A, ew (Å)

Enters atomic data for a transition into the line data buffer, requires the following parameters:

element Two-letter name of element.

ion stage Ionization stage (1 = I, 2 = II, 3 = III, 4 = IV, 5 = V).

wavelength Line wavelength in Å.

ep Lower-level excitation potential in eV.

log gf Oscillator strength.

gamma(nat) Radiative damping constant. See Section 7.1.

gamma(vdw) Van der Waals damping constant. See Section 7.2.

gamma(stk) Stark broadening factor. See Section 7.3.

log A Log_{10} elemental abundance $(\log_{10}(\frac{\mathrm{N_{EL}}}{\mathrm{N_H}}) + 12)$.

ew Equivalent width of line in Å.

There are special procedures for the He I lines $\lambda\lambda$ 4026, 4388, 4471, and 4922. The profiles are taken from the same routines as in Hubeny's SYNSPEC code (Hubeny, Lanz & Jeffery, 1994). The user must insert the correct (or desired) wavelength and log *gf* for each of the above lines according to Table 1. To invoke one of these special He profiles the user must set the wavelength within 1.0 Å of the tabulated values and must set the Stark broadening factor g3 = 0 for the line concerned, otherwise the value supplied will be assumed. The wavelength provided by the user will be the wavelength used, so one can experiment with small wavelength shifts (e.g., to simulate isotope shifts).

The Anstee, Barklem & O'Mara (ABO) VDW broadening parameters are discussed in Section 7.2.

Clears results and errors.

`*clear`

Clears all the lines currently stored in the line data buffer. Clears results and errors.

`*copy`

Copy the values of log A or ew obtained from `exact` into the line data buffer. Maintains results and errors.

`*del` line no., [line no.1]

Deletes the given line from the line data buffer. If a second line number is given all the lines in the range line no. – line no.1 are deleted. The remaining lines are renumbered. Use `*list` to find the line number before using `*del`! Results and errors are re-sequenced as well.

`*dispr` [line no., line no.1]

Displays results from `exact`. If one line number is given, only that line is displayed. If two line numbers are given, then the lines in that range are displayed. Displays the values of d(log A) as well, which are zero unless `errors` has been used.

`*ewcor` factor(%)

Increases the equivalent widths of all the lines in the line data buffer by a percentage factor. The value can be negative to reduce the equivalent widths. Useful when analysing the abundance sensitivity of equivalent width scale factor errors.

`*g3cor` factor(%)

Increases the Stark broadening factor of all the lines in the line data buffer by a percentage factor. The value can be negative to reduce the Stark broadening.

`*list` [line no., line no.1]

Displays lines currently stored in the line data buffer. If one line number is given, only that line is displayed. If two line numbers are given, then the lines in that range are displayed.

`*mean` element, [ion stage]

Calculates the mean log A and standard deviation for a given element. If no ionization stage is given the mean and standard deviation are for *all* available stages. Uses the results from `exact`. Also, gives a weighted mean if `errors` has been used.

`*setab` line no., log A

Sets the abundance of the specified line number.

Clears results and errors.

`*setew` line no., ew (A)

Sets the equivalent width of the specified line number. If the line number is 0, then all the lines in the line buffer are set to the same equivalent width.

Clears results and errors.

`*setgf` line no., log gf

Sets the log gf of the specified line no.

Clears results and errors.

`*sort` [mode]

Sorts the lines in the line data buffer into element order for mode =0 (the default). If mode = 1, then the line data buffer is sorted into wavelength order. Results and errors are maintained.

`abclou` element, log(tau)lo, log(tau)hi, d(Log A)

Enters an abundance scaling factor for this region of the model. Can be used to investigate effects of element stratification. When using `abclou`, it is a good idea to use the n_layer option in `zmodel` to map the atmosphere to a finer mesh of depth points and to use `qualty` option n_emerge = 1 to integrate with linear flux quadrature within strong line profiles. The user should experiment to ensure that the choices do not affect the result.

`alecia` element, log A

Produces a plot of a curve-of-growth. Used for investigating the effects of abundance clouds. Details of this method can be found in Alecian (1982)

Not for general use.

`balfit` Teff, d(Teff), log(g), d(log g)

Determines Teff and/or log(g) by performing a least squares synthesis fit to observational data. Selecting either d(Teff) or d(log g) to 0.0 fixes the corresponding Teff or log(g).

an experimental command

`centrd`

Calculates the centroid of a synthetic spectrum. Useful in the study of blends and complicated profiles of lines with hyperfine or isotope structure.

`chifit` element, log A, [element(2), log A (2), element(3), log A (3), element(4), log A (4), element(5), log A (5)]

Performs least squares synthesis fit to observational spectrum. The log A values are the starting abundances for each element. All other elements in the line buffer retain their abundances as set by `*abund, *atom, *copy,` or `rdlist`.

`chisqu`

Calculates the chi-squared value between the current observations and synthesis.

`cogsyn` element, log A(lo), log A(hi), d(log A)

Performs a synthesis over current `grid` for each abundance. Plots the resulting curve-of-growth. For each abundance, values of the total equivalent width for the element and the centroid value are displayed. Note that none of the log A values in the line data buffer are changed. Any lines from other elements just add a background to the total equivalent width (i.e. a zero-point offset). Useful when studying the curves-of-growth of blends, and lines affected by hfs. If there is just one line of the element in the current grid, a straightforward curve-of-growth is generated for one line, equivalent to the command `growth`.

`contav` wavelength, [mode]

Computes contribution function average. If Mode = 0 gives total, while Mode = 1 line average. For a discussion of the contribution functions refer to Gray (1992; p. 275.) and Achmad, de Jager & Nieuwenhuijzen (1991).

Not for general use

`contfn` wavelength, [mode]

Computes contribution function fluxes. If Mode = 0 gives total, while Mode = 1 line flux. For a discussion of the contribution functions refer to Gray (1992; p. 275.) and Achmad, de Jager & Nieuwenhuijzen (1991).

Not for general use

`contpk` wavelength, [mode]

Computes contribution function peak spectrum. If Mode = 0 gives total, while Mode = 1 line spectrum. For a discussion of the contribution functions refer to Gray (1992; p. 275.) and Achmad, de Jager & Nieuwenhuijzen (1991).

Not for general use

`csynth` [mode]

Performs a continuum-only synthesis over current `grid` range. If mode = 0 then hydrogen-lines are treated as continuous opacity (default). If mode = 1 then the hydrogen-lines are not treated as continuous opacity.

`dump` [mode]

Select test dump prior to reading a model:

(0) Reset all dumps.

(1) Continuous opacity: X, K_λ, σ_λ/K_λ, H, H^-, H_2^+, H_2^-, He, He^+, He^-, Mg, Si.

(2) Flux: X_0, τ_0, τ_ν, B_ν, S_ν, $J_\nu - S_\nu$.

(3) Line opacity at line centre: X_0, ℓ_ν, N(R,S), α, A, V, Γ, ξ^2.

(4) Abundance Fractions: X, R1, R2, R3, R4, N(R)/N(EL), N(R,S), U1, U2, U3, U4, U5. X = monochromatic $\log \tau$ at 5000Å.

(5) Source function.

(6) Line opacity at all frequencies: X_0, ℓ_ν, N(R,S), α, A, V, Γ, ξ^2.

(7) Damping constants.

(8) Drift curve: μ, I(μ)/I(0).

(9) Model atmosphere: log τ, θ, $\log P_g$, $\log P_e$, $\log K_0$, log KS, N(HI)/gm, N(MGI)/gm, N(E)/gm, ΔE. Dump must be on before model is read in.

(10) n/a

All dumps output to session log file (uclsyn.log).

`errors` line no., Teff, d(Teff), log(g), d(log g), xi (km/s), d(xi)

Perform error analysis; determine effect of errors (d) in stellar parameters.

Not for general use.

`errsyn` element, Teff, d(Teff), log(g), d(log g), xi (km/s), (xi)

Calculates error estimate on abundance of selected element using synthesis over current `grid` range.

Not for general use.

`exacld` element, ew (A), log(tau)lo, log(tau)hi, d(log A)

Speciallized version of `exasyn` for use with abundance clouds.

Not for general use.

`exact` mode

Performs curve-of-growth analysis on all the lines in the line data buffer:

mode 0 Determines the abundance of each line from its equivalent width. Uses the value of log A as an initial guessed abundance.

mode 1 Determines the synthetic equivalent width for each line from their log A values.

The results are stored internally and can, for example, be displayed by `*dispr`. Details of the calculations are written to the session log file.

`exasyn` element, log A, ew (A)

Performs an iterative synthesis of element until computed total equivalent width agrees with that given. Log A is used as starting abundance. On completion the lines in the line buffer for the element are adjusted to the new abundance and a synthesis is available for plotting. Any other lines from different elements are not adjusted, but they are included in the synthesis and total equivalent width calculations. This command is useful for working with blended lines.

`fwhm` fwhm (A)

Enters instrumental broadening as full-width half-maximum in Å. Default is 0. Clears any empirical instrumental profiles that may have been read in using `rdinst`. Assumes gaussian profile. See also `iwidth`.

`gfsfit` lambda(lo), lambda(hi), ew_min (A), ew_max (A)

Determines "astrophysical" log gf values by fitting synthesis to observations over given wavelength range for lines within the specified equivalent width range.

`grid` lambda(lo), lambda(hi), d(lambda)

Wavelength range and step size over which synthesis is to be performed.

`growth` line no., log A(lo), log A(hi), d(log A)

Computes a curve-of-growth (plot of log ew versus log A) for the given line no. Produces a plot if `pgdev` is available. Compare `cogsyn`.

`halcor` factor(%)

Apply a halation correction to *IUE* data.

Not for general use.

`help` [topic]

Enter online help utility. If no topic is given a list of commands is displayed.

`ionbal` element, Teff, d(Teff), log(g), d(log g)

Determines ionization balance for given element. Calculates mean abundances f ionization stages I and II for given range of Teff and log g. A contour plot the differences is generated on selected pgplot plotting device.

`iwidth` full width (Å) at (1/e) of max

Enters instrumental broadening width in Å at the level where the profile has fallen to 1/e of maximum. Default is 0. Clears any empirical instrumental profile that may have been read in using `rdinst`. Assumes gaussian profile with s.d. σ. The following relationships hold (see `fwhm`):

$$\mathrm{fw}(1/\mathrm{e}) = 2\sqrt{2}\,\sigma$$

$$\mathrm{fwhm} = 2\sqrt{2\ln 2}\,\sigma$$

so fw (1/e) = 1.20112 fwhm, and fwhm = 2.355 σ.

`lineid` ew (A), [mode, factor(%)]

Identifies absorption lines in the observed spectrum. Requires a model atmosphere to have been read in. The identified lines are stored in the line buffer. Uses the elemental abundances from file `uclsyn_abs.init` or read in via `rdabun`.

ew (A) is the minimum equivalent width allowable in the line identification process, for both observed and synthetic equivalent widths.

mode changes the method of derivative calculation used in the identification of absorption lines. Mode = 0 is the default. If mode is between 1 and 10 (for broader profiles) a more elaborate calculation is made.

factor(%) is the noise estimate given as a percentage. It also doubles as the minimum percentage contribution allowed in blends. The default is 1.0%.

Under development

`magain` xi (min), xi (max), d(xi)

Computes the 'optimal' value of the microturbulence parameter for a sample of spectral lines of the same species by using the null-correlation criterion between inferred abundance and *synthetic* equivalent width (see Magain, 1984). If a graphics device is enabled by `pgdev` prior to using the command a series of plots log A versus ew will be displayed. The original value of `mictrb` is unchanged. Compare `turbul`. The line buffer should contain only the species to be analysed (e.g. Fe 2) when `magain` is used. Note that UCLSYN does not check for this; it is up to the user.

`mictrb` xi (km/s)

Enters depth-independent microturbulent velocity in km/s. Default is 0.

`mkgrid` tolerance

A rather speciallized command for making a non-uniformly spaced wavelength grid, with points centred around lines within the line buffer.

Not for general use

`mu` cos(theta)

Sets the value of the disk position $\mu = \cos\theta$. If $0 < \mu \leq 1$ then the specific intensity is calculated for that value of μ. If $\mu = 0$ then the flux is calculated. Default value is $\mu = 0$ (i.e. flux is calculated).

`noise` factor(%)

Adds percentage Gaussian noise to broadened spectrum.

`pdepth` d(tau)

Speciallized command that labels depth points on a synthetic spectrum. Requires use of `synth -1` option.

Not for general use.

`pgadv`

Plotting page advance disable/enable toggle. With page advance disabled `pgplot` will plot over the previous plot without drawing a new box. With page advance enabled `pgplot` will start a new plot. When UCLSYN starts `pgadv` is enabled by default.

`pgdev` device, [x_pages, y_pages, colour]

Enters graphics device. x_pages and y_pages are used to specify the number of plots in x and y directions that are to be plotted on one graphics page. If no x_pages and y_pages values are given, they are set to 1. Optional colour parameter: if 0 then monochrome plotting is performed; if non-zero then colour plotting is performed. Default = 0.

Examples of graphics devices are `/GWM` (Unix or Linux X-window) or `/PS` (send output to a postscript file). A list of available devices can be obtained by typing `pgdev ?`.

`pgew`

Interactively measure the equivalent width of an absorption line in the observed spectrum. Requires the observed spectrum to be displayed on the screen with `pgplot`. Starts cursor. Typing (A) marks a continuum point where integration starts, another (A) marks the end point, (D) deletes a point, and (X) exits and displays the measurement (and sends it to the log file). If more than two values of A are entered, the extra ones are ignored. The equivalent width is normalised to the straight line joining the two points. A search is made for plausible identifications in the line buffer.

`pgnorm`

Rectify observational data displayed on the screen by `pgplot,` using straight line or Hermite spline based on the cursor positions of up to 50 points. Move cursor to observational data. Typing (A) adds a point, (D) deletes a point, (X) exits and rectifies observational data.

`pgplot` [mode]

Plot synthetic and/or observational data to current graphics device. The modes are:

mode 0 Plot both synthetic and observational spectrum (default when `pgadv` enabled). If available, the synthetic spectrum is combined with a telluric spectrum.

mode 1 Plot synthetic spectrum only (default when `pgadv` disabled).

mode 2 Plot observational spectrum only.

mode 3 Plot telluric spectrum only.

The current rotational velocity and instrumental profile are convolved with the synthetic spectrum prior to plotting.

`pgrect`

Rectify observational data using straight line or Hermite spline based on the cursor positions of up to 50 points. Move cursor to observational data to locate wavelength points, which are then used to locate the corresponding points on synthetic spectrum. The observations are then normalized such that the observations and synthesis agree at the selected wavelength points. Typing (A) adds point, (D) deletes point, (X) exit and rectify observational data.

`pgxran` [x_lower, x_upper]

Select x-range over which synthetic and observational data are to be plotted. If no parameters are given, the x-range is cleared and set automatically by `pgplot`.

`pgyran` [y_lower, y_upper]

Select y-range over which synthetic and observational data are to be plotted. If no parameters are given, the y range is cleared and set automatically by `pgplot`.

`plines` [ew (A), mode]

Superposes line identifications on a synthetic/observed spectrum plot. The optional parameter ew (A) allows a minimum equivalent width to be specified, in which case results must be available (i.e., there must be lines in the line buffer and `exact` must have been run). It maybe necessary to set `pgyran` (E.g. 0.0 to 1.1) to produce reasonable results without overcrowding the plot.

mode 0 (default) annotates synthetic plots.

mode 1 annotates observational plots.

Note that when in BINSYN BINARY MODE it labels only the current star.

`qualty` n_emerg, n_tau, n_saha

The `qualty` command specifies three parameters that control the solution integration methods, and the total number of ionization stages to be considered. The input parameters and value ranges are:

n_emerg Code for Flux/Intensity Quadrature Type.

1 = Linear.

2 = Quadratic (default).

3 = Four Point Lagrange.

4 = Simpson.

n_tau τ_ν Integration Type.

1 = Simpson's Rule (default)

2 = Bode's Rule

n_saha Maximum Stages of Ionization for Saha Ionization Equilibrium Calculations

1 thru 5 = possible range (default = 5)

In the majority of cases (solar type stars) this command may be omitted.

`quit`

Ends session, closing the session log file, `uclsyn.log`, in current directory.

`rdabun` filename

Read elemental abundances for continuous opacity from a file. By default, the program automatically reads in a set of Anders & Grevesse (1989) abundances defined by the file `uclsyn_abs.init` (slightly modified by updates). These abundances are also used by `lineid` and `rdlist`. The easiest way to make a new file to be read is to copy `uclsyn_abs.init` to another file in a local directory and modify the appropriate abundances; for example, by increasing the Mn abundance for HgMn stars, the Eu abundance for Eu stars, etc. The file has list-directed input with 23 lines having four elements each, in order of atomic number.

Note: Using non-default `uclsyn_abs.init` will affect the results of global abundances.

`rdblis` ew (A), lambda(lo), lambda(hi)

A variant of `rdlist`, which uses a binary file rather than plain text to improve performance.

Not for general use.

`rdclou` element, filename

Read in an abundance cloud model for given element. Allows cloud to have gradients. The file must have at least two layers; each layer must contain two numbers:

1) $X_0 = \log(\tau_0)$, log optical depth at 5000 Å.

2) D(log A), the logarithmic increase factor of the abundance at that layer.

The same precautions used for `abclou` regarding `qualty` options should be observed. Used in conjunction with `alecia.`

Not for general use.

`rdcmd` filename

A very important command! Reads in a file of UCLSYN commands and executes them. When the last command has been executed, the command prompt returns.

`rdelmt` element, ew(min), lambda(lo), lambda(hi)

Reads in all lines from the Kurucz CD23 list and finds all lines of the specified element in all ionization stages with synthetic ew greater than ew(min) in the range lambda(lo) to lambda (hi) inclusive. Use with caution to avoid excessive execution time.

`rdinst` filename

Read in a tabulated empirical instrumental profile. Used when the instrumental profile is asymmetric or non-gaussian. Must be a two column list-directed file of $\Delta\lambda$ (Å) and intensity points. Causes any previous value of `iwidth` or `fwhm` to be ignored.

To avoid wavelength shift problems, the profile should have the value of $\Delta\lambda = 0$ exactly at the centroid. UCLSYN interpolates and renormalises the profile (as it does for all instrumental profiles) when its tabulated spacing is not the same as that for a synthesis.

`rdkurz` filename, [n_layer]

Reads in a file containing a Kurucz ATLAS model atmosphere.

Not fully tested.

`rdlist` ew (A), lambda(lo), lambda(hi)

Reads in lines from the Kurucz CD23 linelist; ew (A) is the minimum allowable synthetic equivalent widths. If ew (A) = 0, then all lines in the specified wavelength range are read in. Uses the elemental abundances from `uclsyn_abs.init` or read in via `rdabun`. The default Kurucz `CD23 gfall.dat` linelist contains 504 800 lines in the range 1000 – 100 000 Å.

`rdmod` filename, [n_layer]

Reads in a file containing a model atmosphere of five columns;

1) $X_0 = \log(\tau_0)$ or τ_0 : Monochromatic optical depth at 5000Å.

2) $T(\tau_0)$ or $\theta(\tau_0) = 5040/T(\tau_0)$: Temperature or 5040/T.

3) $\log P_g(\tau_0)$: Total gas pressure (atoms + electrons, cgs).

4) $\log N_e(\tau_0)$ or $\log P_e(\tau_0)$: Electron density or pressure (cgs).

5) $\log \kappa_0(\tau_0)$: Mass absorption coefficient at 5000Å.

This should be given in constant steps in $\log \tau_0$, but UCLSYN will map the model onto a regular $\log \tau_0$ grid, if necessary.

`rdobs` filename, [filetype]

Read in observational data from a file. Filetype is as follows:

−1	Starlink sdf or ndf type files (default)
0	Dipso's original Spectrum 0 unformatted filetype
1	Standard two-column formatted ASCII file
2	Dipso's Spectrum 2 formatted filetype

`rdsyns` filename, [filetype]

Read in a spectrum into synthesis arrays, as if it had been generated using `synth`. Filetype is as follows:

−1	Starlink sdf type files (default)
0	Dipso's original Spectrum 0 unformatted filetype
1	Standard two-column formatted ASCII file
2	Dipso's Spectrum 2 formatted filetype

This is useful if you want to perform rotational or instrumental broadening operations on a spectrum, for example an observed template spectrum or someone else's synthetic spectrum.

`rmarcs` filename, [n_layer]

Reads in a file containing a MARCS-format model atmosphere of six columns;

1) $X_R oss = \tau_R oss$: Rosseland mean optical depth.

2) $X_0 = \tau_5 000$: Monochromatic optical depth at 5000Å.

3) $T(\tau_0)$ or $\theta(\tau_0) = 5040/T(\tau_0)$: Temperature or 5040/T.

4) P_e : Electron pressure (cgs).

5) P_g : Total gass pressure (cgs).

6) $\kappa_R oss$: Rosseland absorption coefficient.

Not for general use.

`rotfit` [element, log A]

Determines vsini by performing a least squares synthesis fit to observational data.

`rxrang` lambda(lo),lambda(hi)

Restricts x-range of the observed spectrum.

`scater` mode

Continuum scattering switch. mode = 1 to enable, mode = 0 to disable. Initially mode = 0. Needed for hotter stars ($T_{\rm eff} >$ 15-16000 K).

`select` ewmin (A), [lambda(lo),lambda(hi)]

Deletes spectral lines in the line buffer which have a synthetic equivalent width below the user-specified minimum value ewmin. Optional wavelength range can be entered to *delete* spectral lines *outside* that range. Results are stored in arrays and can be examined by `*dispr`, etc.

`settol` tolerance

Sets the fractional tollerance used in least squares fitting. Default = 0.001

`spawn`

Create and enter a VMS DCL subprocess or a UNIX or LINUX shell without `quit`ting UCLSYN. With X-windows it is probably easier just to have another nxterm open. To return to UCLSYN prompts, use <ctrl>-D.

`spcbrd`

Applies rotational and instrumental broadening to the synthetic spectrum by convolution using user-specified values of the instrumental profile and $v \sin i$. Used when writing a synthesis to a file without the need to plot the synthesis.

`spcint`

Integrates synthetic spectrum and gives the total equivalent width.

`status`

Displays current UCLSYN status.

`synth` [mode]

Computes a synthetic flux or intensity spectrum over the current `grid` range. mode = 0 (default) gives a synthesis in which Hydrogen lines are treated as continuous opacity and normalized out. Mode = 1 gives a synthesis in which Hydrogen lines are treated as true line opacity and present in the synthesis (i.e., the model continuum will not be 1.000). The profiles are normalized at ±100Å from the core.

`tgplot` Teff, d(Teff), log(g), d(log g)

Determines the Chi^2 for a rectangular range of teff and logg values.

A contour plot is generated on selected pgplot plotting device.

`turbul` xi (min), xi (max), d(xi)

Computes the 'optimal' value of the microturbulence parameter for a sample of spectral lines of the same species by using the null-correlation criterion between inferred abundance and *observed* equivalent width. If a graphics device is enabled by `pgdev` prior to using the command, a series of plots of log A versus ew will be displayed. Restores `mictrb` to original value. Compare `magain`.

`turfit` [element, log A]

Determines microturbulence by performing a least squares synthesis fit to observational data.

`vshift` rv (km/s)

Enter radial velocity of the observed spectrum; it will be shifted to a rest wavelength scale in the plot. The shifted spectrum can be written using `wrobs`. Compare `xshift`.

`vsini` ve*sin i

Enters rotational velocity in km/s (default = 0). UCLSYN calculates the rotational profile using limb-darkening appropriate to the model atmosphere and wavelength.

`wrlist` filename

Writes current line buffer into a file containing a set of `*atom` commands. Can be read in again using `rdcmd`. Useful for saving the day's work for continuing another time.

`wrobs` filename, [filetype]

Writes observational spectrum to a file. filetype is as follows:

−1	Starlink sdf type files (default)
0	Dipso's original Spectrum 0 unformatted filetype
1	Standard two-column formatted ASCII file
2	Dipso's Spectrum 2 formatted filetype

Useful when it is necessary to generate an observed spectrum with a doppler shift or revised continuum normalisation generated using UCLSYN.

`wrres` filename

Writes results from `exact` to a formatted file.

`wrsyns` filename, [filetype]

Write synthetic spectrum to a file. filetype is as follows:

−1	Starlink sdf type files (default)
0	Dipso's original Spectrum 0 unformatted filetype
1	Standard two-column formatted ASCII file
2	Dipso's Spectrum 2 formatted filetype

`xshift` shift (A)

Apply an x-shift to the observed spectrum. Compare `vshift`.

`zmodel` Teff, log(g), [d(log A), n_layer]

Reads in a model from a grid of Kurucz (1993) model atmospheres combined with the COLK95 (Castelli et al, 1997) grid for cooler stars. Model interpolation is performed for any sensible $T_{\rm eff}$

and log g values in range 3500 – 50000 K, 0.0 – 5.0 dex. Other available grids (e.g., Kurucz, 1979) can be chosen by changing the 'makemake' file's setenv command for the appropriate directory for UCLSYN_MODDIRE, but for most purposes the current models95 directory is the best one to use.

Optional parameter n_layer can be used to change the number of depth points in the atmosphere; see `abclou` and `alecia`. Maximum number of layers is 200.

`zscale` d(log A)

Applies a metallicity scaling to lines within the line buffer.

Not for general use.

`zscatr` factor(%)

Apply scattered light correction to observational spectrum (see Gray, 1976 pp. 283–290; 1992 pp. 256–262).

5 BINSYN–A binary star version of UCLSYN

5.1 Introduction

The BINSYN BINARY MODE allows UCLSYN to be used in the analysis of double-lined spectroscopic binary systems. The program can mostly be used in the same way as for normal SINGLE-STAR MODE, but there are a few commands which are not available or which have been modified, and there are new commands to input necessary parameters such as the light ratio and radial velocities of the component stars. The BINSYN BINARY MODE uses the concept of current and shadow arrays; UCLSYN commands will operate on the current star (1) only; special BINSYN commands operate on both arrays.

While BINSYN is primarily intended for use in the investigation of double-lined spectroscopic binary systems, it can be used in 1) single-lined system where an 'unseen' companion can be allowed for by an appropriate light ratio and 2) in two-component analyses with differing models and light ratios, but no radial velocity separation (e.g. in the study of star spots, convection streams, etc.).

`star` mode

Invoking `star` with N = 1 makes the original choice for star 1 the current array and star 2 the shadow array, so that all subsequent commands operate on star 1. In this way, for example, the line list can be read in for star 1. Invoking `star` 2 brings star 2 into the current arrays and moves star 1 into shadow arrays so that commands can be used on star 2 data.

Note that `rvals` and `lratio` always refer to star 1 and star 2 in that order.

Using `star` 0 resets UCLSYN to "classic" SINGLE-STAR MODE.

`rvals` RV (1), RV (2)

Enters radial velocities of stars 1 and 2 within BINSYN BINARY MODE.

`lratio` L Ratio(1), L Ratio(2)

Specifies the monochromatic relative light contributions for the two stars within BINSYN.

`binfit` element, log A, [element(2), log A (2), element(3), log A (3), element(4), log A (4), element(5), log A (5)]

Performs least squares synthesis fit to observational data. This is a variant of chifit, which adjusts the abundances of both current and shadow arrays while in BINSYN BINARY MODE. Thus, the final abundances obtained a

Only available in BINSYN BINARY MODE.

Not for general use.

5.2 `Equivalent widths`

Observed equivalent widths for the components should not be corrected for dilution effects, but should be given as measured in the observed spectrum. The calculated equivalent widths are adjusted for dilution effects, so that ew (Å) is multiplied by $L_{\rm current}/(L_{\rm shadow} + L_{\rm current})$. Any chage to the light ratio will require the equivalent width to be re-calculated.

6 TELSYN commands

Some intro comments... HITRAN, earth atmosphere model...
EXPERIMENTAL COMMAND...

`telsyn` Altitude, Airmass

Calculates a telluric spectrum over the current `grid` range for the specified observatory altitude and observational airmass. Uses Nicholls (1988) 6-layer model of the Earth's atmosphere and the HITRAN molecular database. Currently only uses O2, O3 and H2O molecules.

`rdtell` filename, [filetype]

Read in a telluric spectrum into telluric arrays, as if it had been generated using `telsyn`. Allows for observational telluric spectra to be read in. Filetype is as follows:

−1 Starlink sdf or ndf type files (default)
0 Dipso's original Spectrum 0 unformatted filetype
1 Standard two-column formatted ASCII file
2 Dipso's Spectrum 2 formatted filetype

`wrtell` filename, [filetype]

Write telluric spectrum to a file. Filetype is as follows:

−1 Starlink sdf or ndf type files (default)
0 Dipso's original Spectrum 0 unformatted filetype
1 Standard two-column formatted ASCII file
2 Dipso's Spectrum 2 formatted filetype

7 Damping constants

The damping constant input values are now discussed.

7.1 Radiative damping constant gamma(rad)

If the entry is $< 10^3$ it is used to multiply the classical damping constant:

$$\gamma_{\rm rad} = \text{entry} \times \gamma_{\rm clas}$$

where $\gamma_{\rm clas} = 0.2223 \times 10^{16}/\lambda^2$.

If the entry is $> 10^3$ it is used directly as the damping constant (in units $\rm s^{-1}$).

If the entry is 0, then $\gamma_{\rm rad} = 0$, i.e. no radiative damping.

7.2 Van der Waals damping constant gamma(vdw)

The Van der Waals damping constant arises from collisions with neutral atoms (mainly H and He). Here, it is treated as temperature and density dependent, and is computed using the asymptotic dipole-dipole (Lindholm-Foley) theory. The difference in mean-square-radii of the radiating electron ($\Delta\bar{r}^2$) in Bohr atomic units is computed using one of the following:

1. user-supplied value (coded as a negative quantity) taken as $3.667 \times 10^{-9}(\Delta\bar{r}^2)^{0.4}$ at 10000 K, where $\Delta\bar{r}^2$ comes from a quantum mechanical calculation (e.g. from literature);
2. use approximate hydrogenic $\Delta\bar{r}^2$ with principal quantum number dependence only.
3. use precise hydrogenic $\Delta\bar{r}^2$ with principal and orbital quantum number dependences.
4. user-supplied value for the Anstee, Barklem & O'Mara (ABO) theory for neutral lines.

If the entry is < 0 then the user-supplied value yields the quantity $|entry|/Z_\theta$, where Z_θ is a constant (1.228205) which converts the user-supplied value at T = 10000 K to a value per unit temperature (assumes $\theta^{-0.3}$ dependence). (see for example Kurucz, 1988).

If the entry is > 0 then a hydrogenic approximation is used to obtain $\Delta\bar{r}^2$. The entry is of the form

$$10 \times L(\text{upper}) + L(\text{lower})$$

and used in the precise hydrogenic formula. If the orbital quantum numbers are not known then using 19 as the entry will cause the approximate hydrogenic formula to be used. Finally,

$$\gamma_{\rm VDW} = 3.667 \times 10^{-9}(\Delta\bar{r}^2)^{0.4}\theta^{-0.3}(N_H I + N_{HeI})$$

See, e.g., Warner, 1967. If the entry is 0 then $\gamma_{\rm VDW} = 0$. i.e. no Van der Waals damping.

If entry is > 100 and Zeffec = 1 (i.e. neutral line), then the Anstee, Barklem & O'Mara (ABO) theory is used. The integer part is SIGMA and decimal part is ALPHA. Uses the routines from Paul Barklem ([7, 6, 5]).

7.3 Stark broadening factor gamma(stk)

The Stark quadratic damping constant is treated as temperature and density dependent and is computed as follows:

1. The user-supplied value is taken as $\gamma_{\rm Stk}/n_{\rm e}$ and is used to compute the damping constant directly.
2. Lindholm-Foley formula: the user-supplied value is taken as the level displacement (in units of $\rm cm^{-1}$ per 100 kV/cm electric field) and the damping constant is computed.
3. Griem formula for ions: the user-supplied value is taken as a coding of the orbital quantum numbers for the transition energy levels and used to compute the precise hydrogenic mean square-radii for the transition states. The damping constant is computed using Griem's (1968) formula for ions.
4. Fraudenstein & Cooper formula for neutrals: the user-supplied value is taken as a coding of the orbital quantum numbers for the transition energy levels and used to compute the precise hydrogenic mean square-radii for the transition states. The damping constant is computed using Fraudenstein & Cooper's (1978) formula for neutrals.

If the entry is $< -10^{-4}$, e.g. -1.53, then the Lindholm-Foley formula is used, d = |entry|.

If the entry is $> -10^{-4}$ and < 0, e.g. -2.8×10^{-6}, then $\gamma_{\rm Stk}/n_{\rm e} = |\text{entry}|$.

If the entry is > 0 then the entry is of the form

$$10 \times L(\text{upper}) + L(\text{lower})$$

If $Z_{\rm eff} = 1$ (i.e. a neutral atom) then the Fraudenstein & Cooper formula is used, otherwise the Griem formula for ions is used. The mean-square-radius of the upper level is computed using the Hydrogenic approximation in either its precise (principal and orbital quantum number dependence) or approximate (principal quantum number dependence only) form. If the orbital quantum numbers are not known then using 19 as the entry will cause the approximate hydrogenic formula to be used.

If the entry = 0 then $\gamma_{\rm Stk}/n_{\rm e} = 0$, i.e. no Stark broadening.

7.4 Special Helium Lines

Table 1 lists the input data needed in a `*atom` command to invoke detailed He I profile calculations. These are taken from the SYNSPEC code of Hubeny et al (1994), who based the profiles on the work of Barnard et al (1969, 1974, 1975) for λ4471 and 4921, and on the PhD thesis of L. J. Shamey (1969, University of Colorado) for the other lines. The log gf values come from the critical compilation of Wiese et al (1966).

Remember, to invoke the special He I profiles for these four lines, the value of gamma3 must be set to 0, and the user can adjust the wavelength within 1.0 Å of the nominal value. The wavelengths listed below should, of course, work reasonably well for most applications.

Table 1: He I data for detailed profiles of four lines

Wavelength, Å	lower ep (eV)	log gf
4026.20	20.963	-0.370
4387.93	21.217	-0.883
4471.50	20.963	+0.052
4921.93	21.217	-0.435

8 UCLSYN Example

Below is an example of UCLSYN commands and usage:

```
  welcome to uclsyn 3.1
  reading initialisation files ...

   type "help" for list of commands available
   type "help summary" for one-line descriptions

> pgdev xwindows 1 2
> *atom fe 2 6147.735 3.889 -2.721 1 21 21 7.55 0.035
> *atom fe 2 6149.238 3.889 -2.724 1 21 21 7.55 0.035
> *list
  no  id   lambda     ep  log(gf)  g(rad)   g(vdw) g(stark)   ew  log(A)
   1 Fe 2 6147.735  3.889 -2.721  1.00E+0  2.10E+1  2.10E+1 0.035  7.550
   2 Fe 2 6149.238  3.889 -2.724  1.00E+0  2.10E+1  2.10E+1 0.035  7.550
> exact 0
 ! no prior execution of rdmod
 ! command aborted
> zmodel 11250 4.13
  remapping model onto a regular grid
> exact 0
  calculating abundances ...
  results available
> *dispr
  results from exact 0

  no  id   lambda     ep  log(gf)  g(rad)   g(vdw) g(stark)   ew  log(A) d(logA)
   1 Fe 2 6147.735  3.889 -2.721  1.00E+0  2.10E+1  2.10E+1 0.035  8.064  0.000
   2 Fe 2 6149.238  3.889 -2.724  1.00E+0  2.10E+1  2.10E+1 0.035  8.067  0.000
> mic 2
> exact 0
  calculating abundances ...
  results available
> *dispr
  results from exact 0

  no  id   lambda     ep  log(gf)  g(rad)   g(vdw) g(stark)   ew  log(A) d(logA)
   1 Fe 2 6147.735  3.889 -2.721  1.00E+0  2.10E+1  2.10E+1 0.035  7.879  0.000
   2 Fe 2 6149.238  3.889 -2.724  1.00E+0  2.10E+1  2.10E+1 0.035  7.882  0.000
> *mean fe
  unweighted mean for    2 lines, log A =  7.881 +/-  0.002  [s.d. =  0.002]
! no weighted mean possible
> mic 0
> exact 1
  calculating equivalent widths ...
  results available
> *dispr
  results from exact 1
```

```
  no  id   lambda     ep  log(gf)  g(rad)   g(vdw) g(stark)   ew  log(A) d(logA)
   1 Fe 2 6147.735  3.889 -2.721  1.00E+0  2.10E+1  2.10E+1 0.019  7.550  0.000
   2 Fe 2 6149.238  3.889 -2.724  1.00E+0  2.10E+1  2.10E+1 0.019  7.550  0.000
> grid 6146 6151 0.001
> synth
  calculating synthetic spectrum ...
> pgplot
! no observational data to plot
> vsini 20
> pgadv
  page advance disabled
> pgplot
  applying rotational broadening ...
> pgadv
  page advance enabled
> *atom hg 2 6149.475 11.867 +0.328 1 01 01 3.00 0.01
> exact 1
  calculating equivalent widths ...
  results available
> *dispr
  results from exact 1

  no  id   lambda     ep  log(gf)  g(rad)   g(vdw) g(stark)   ew  log(A) d(logA)
   1 Fe 2 6147.735  3.889 -2.721  1.00E+0  2.10E+1  2.10E+1 0.019  7.550  0.000
   2 Fe 2 6149.238  3.889 -2.724  1.00E+0  2.10E+1  2.10E+1 0.019  7.550  0.000
   3 Hg 2 6149.475 11.867  0.328  1.00E+0  1.00E+0  1.00E+0 0.000  3.000  0.000
> synth
  calculating synthetic spectrum ...
> vsini 0
> pgplot
! no observational data to plot
> *abund hg 6.3
> *list
  no  id   lambda     ep  log(gf)  g(rad)   g(vdw) g(stark)   ew  log(A)
   1 Fe 2 6147.735  3.889 -2.721  1.00E+0  2.10E+1  2.10E+1 0.035  7.550
   2 Fe 2 6149.238  3.889 -2.724  1.00E+0  2.10E+1  2.10E+1 0.035  7.550
   3 Hg 2 6149.475 11.867  0.328  1.00E+0  1.00E+0  1.00E+0 0.010  6.300
> synth
  calculating synthetic spectrum ...
> pgadv
  page advance disabled
> pgplot
> iwid 0.1
> pgplot
  applying instrumental broadening ...
> vsini 5
> pgplot
  applying rotational broadening ...
  applying instrumental broadening ...
> quit
```

Figure 1 shows what the screen should now look like.

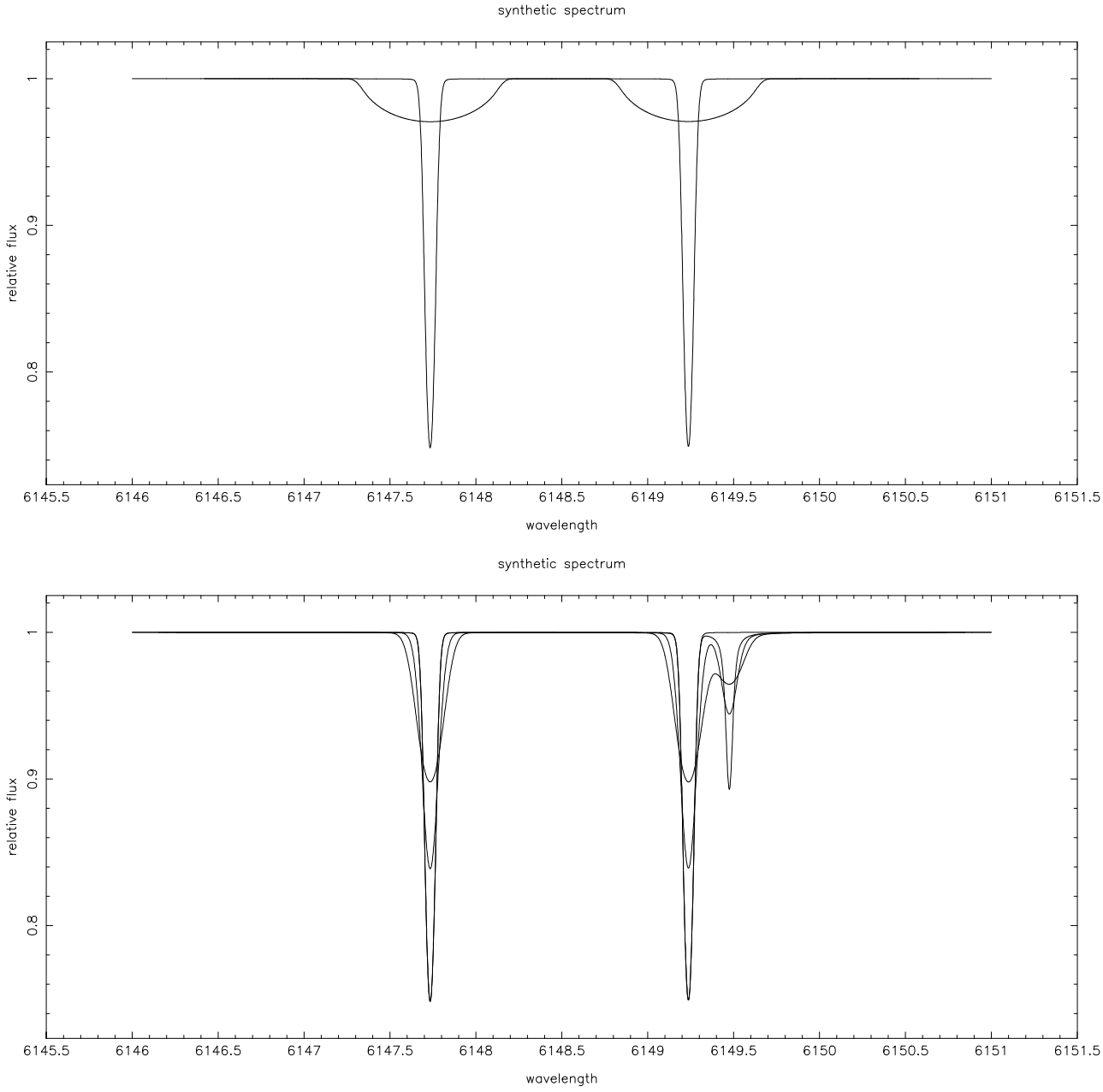


Figure 1: Display from the UCLSYN example

9 BINSYN Example

For all but the most simple syntheses, it is advisable to work with command files rather than line-by-line commands. The following example is for a synthesis performed for the binary HgMn star ι CrB for high-resolution spectra from the CFHT. The binary components are not fully resolved. At about observed wavelength 4045.3 Å there is an unidentified very sharp absorption line in the primary star.

The command file starts with `*clear` commands to clear any entries in the buffers which might have been left by a previous synthesis. These can be omitted *if* the user is sure that the file will be run before any other command file. The only result of using `star` and `*clear` as the first commands is the generation of harmless error messages.

```
star 2
*clear
star 1
*clear
%****  data and device commands  ****
rdobs 4055icrb10an.txt 1
pgdev xwindows 1 1 1
%****  model parameters  ****
models 11000 4.0 9000 4.3
rvals -23.6 -16
lratio 0.73 0.27
mictrb 0 0.95
vsini 1 2.1
%****  grid parameters  ****
pgxran 4043.5 4046.5
pgyran 0.3 1.02
grid 4042.5 4047 0.001
star 1
%****  line list  ****
%****  star 1  ****
*atom Fe 2  4044.012  5.571 -2.414  3.69E+0 -1.29E-8 -2.95E-7 7.77 0.000
*atom Fe 1  4044.492  4.103 -1.137  2.17E+0 -3.02E-8 -9.33E-6 7.6 0.000
*atom P  2  4044.576 13.311  0.481  1.00E+0 -3.47E-8 -1.07E-5 6.5 0.000
*atom Fe 1  4044.610  2.832 -1.080  5.98E-1 -1.62E-8 -6.31E-7 6.9 0.000
*atom Fe 1  4045.593  3.211 -1.271  1.47E+0 -2.09E-8 -3.98E-6 7.6 0.000
*atom Zr 2  4045.638  0.710 -0.603  1.00E+0 -1.52E-8 -8.15E-7 3.17 0.000
*atom Fe 1  4045.813  1.485  0.280  7.53E-1 -1.59E-8 -6.17E-7 7.48 0.000
*atom Fe 1  4046.060  3.266 -1.486  1.57E+0 -2.04E-8 -5.37E-6 8.1 0.000
%Platinum isotopes and hyperfine components
%192
%*atom pt 2 4046.4047 4.52  -1.19 5.82 23 -2.0E-6 3.5  0.000
%194
*atom pt 2 4046.4429 4.52  -1.19 5.82 23 -2.0E-6 3.95  0.000
%195_b
*atom pt 2 4046.4522 4.52  -1.60 5.82 23 -2.0E-6 4.95  0.000
%195_a
*atom pt 2 4046.4656 4.52  -1.45 5.82 23 -2.0E-6 4.95  0.000
%196
*atom pt 2 4046.4824 4.52  -1.19 5.82 23 -2.0E-6 5.1  0.000
%198
```

```
*atom pt 2 4046.5251 4.52  -1.19 5.82 23 -2.0E-6 4.89  0.000
%195_c
%*atom pt 2 4046.463 4.52  -2.75 5.82 23 -2.0E-6 4.80  0.000
*atom Hg 1 4046.556 4.667 -1.17 5.82 23 -2.0E-6 6.05  0.000
*atom Hg 1 4046.562 4.667 -1.17 5.82 23 -2.0E-6 6.05  0.000
exact 1
%****  star 2  ****
star 2
*atom Fe 2  4044.012  5.571 -2.414  3.69E+0 -1.29E-8 -2.95E-7 7.77 0.000
*atom Fe 1  4044.492  4.103 -1.137  2.17E+0 -3.02E-8 -9.33E-6 7.6 0.000
*atom P  2  4044.576 13.311  0.481  1.00E+0 -3.47E-8 -1.07E-5 5.450 0.000
*atom Fe 1  4044.610  2.832 -1.080  5.98E-1 -1.62E-8 -6.31E-7 6.9 0.000
*atom Fe 1  4045.593  3.211 -1.271  1.47E+0 -2.09E-8 -3.98E-6 7.6 0.000
*atom Zr 2  4045.638  0.710 -0.603  1.00E+0 -1.52E-8 -8.15E-7 2.6 0.000
*atom Fe 1  4045.813  1.485  0.280  7.53E-1 -1.59E-8 -6.17E-7 7.48 0.000
*atom Fe 1  4046.060  3.266 -1.486  1.57E+0 -2.04E-8 -5.37E-6 7.2 0.000
exact 1
%****  synthesis commands  ****
star 1
iwidth 0.045
bsynth
pgplot 0
plines
star 2
plines
```

The result can be seen in Fig 2. The two components are partially resolved in wavelength (this is about as large a separation as they ever achieve). The strong Fe I line is accurately reproduced, as is the Pt II line. The (diluted) equivalent widths are given together with the identifications, because `plines` takes them from the line buffers. The user did not identify some lines which were not crucial to the research result, hence these are unsynthesised.

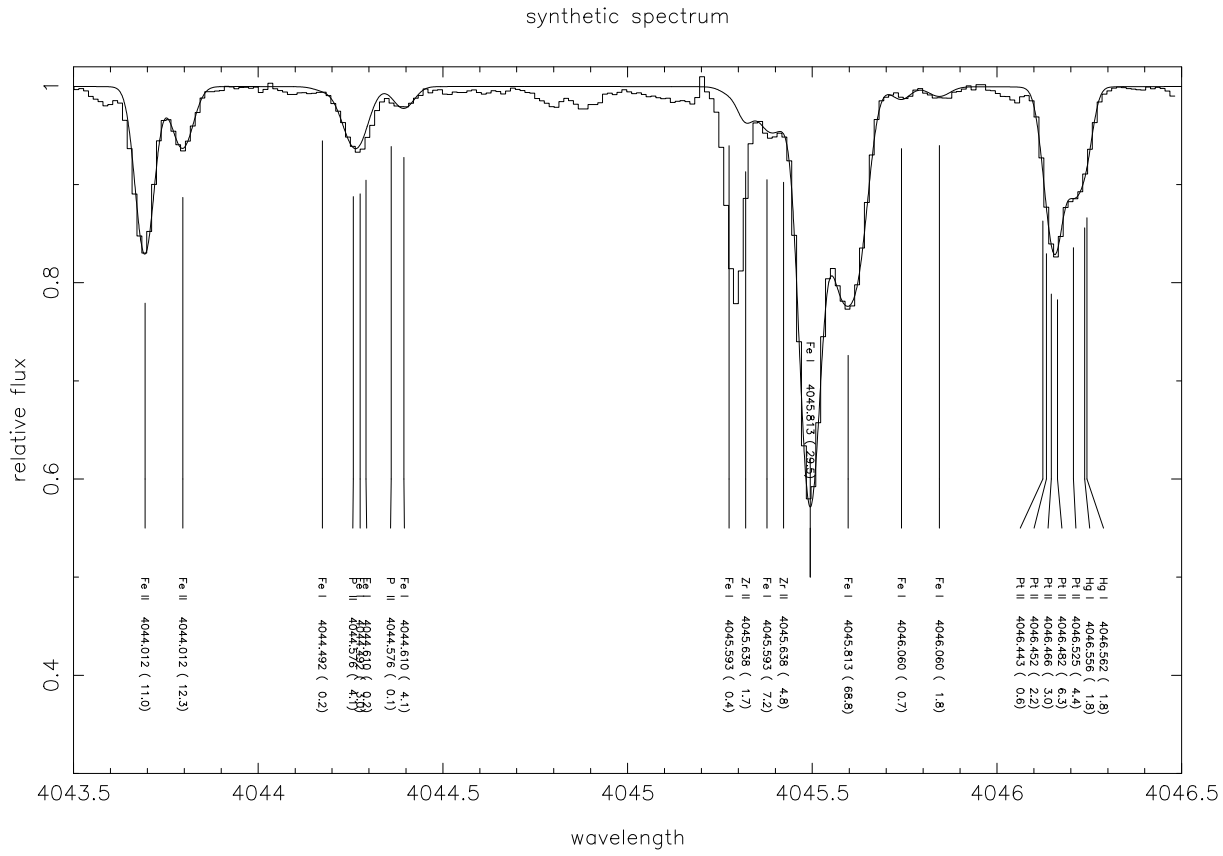


Figure 2: Display from the BINSYN example; the region of the Fe I and Pt II lines in ι CrB

10 Sources of atomic data

10.1 Sources of data for partition functions

Primary sources are Drawin & Felenbok (1965) and Kurucz (1970). For low temperatures, the *Atomic Energy Levels* were used in the calculation of U values to extend the tabulated values to 0 K (i.e. ground state g value). Where no tabulated U values were available, values were calculated from the *Atomic Energy Levels* and are generally not complete. If no data could be found a value of $U = 1$ was assumed. The sources for the various elements are given in Table 2.

For the part of the partition function above the explicit sum, the function is extended by an integral up to the energy of the cut-off potential due to collisions with electrons. A separate code is available for tabulating the partition functions.

10.2 Continuous opacity

Peach's (1970) tables are used for the neutral atoms of Mg I, Si I, Al I and C I and the singly-ionized atoms of Mg II, Si II, Al II and C II.

10.3 Hydrogen line opacity

VCS theory is used for H_α to H_δ (Vidal et al., 1973), and ESW theory for H_ϵ to H_{16} (Edmonds et al., 1967). The VCS subroutine is based on Peterson's (1969) BALMER program.

Table 2: Sources of data for partition functions

	I	II	III	IV	V
H	MMD	U=1	U=1	U=1	U=1
He	D&F	D&F	U=1	U=1	U=1
Li	MMD	D&F	D&F	U=1	U=1
Be	MMD	MMD	KUR	KUR	U=1
B	MMD	MMD	MMD	AEL	AEL
C	MMD	MMD	D&F	D&F	AEL
N	MMD	MMD	MMD	D&F	D&F
O	MMD	NIST	MMD	D&F	D&F
F	D&F	D&F	D&F	D&F	D&F
Ne	D&F	D&F	D&F	D&F	KUR
Na	NIST	NIST	NIST	NIST	NIST
Mg	NIST	NIST	NIST	NIST	NIST
Al	NIST	NIST	NIST	NIST	NIST
Si	NIST	NIST	NIST	NIST	NIST
P	NIST	NIST	NIST	NIST	NIST
S	NIST	NIST	NIST	NIST	NIST
Cl	D&F	D&F	D&F	D&F	D&F
Ar	D&F	D&F	D&F	D&F	D&F
K	NIST	NIST	NIST	NIST	NIST
Ca	NIST	NIST	NIST	NIST	NIST
Sc	NIST	NIST	NIST	NIST	NIST
Ti	NIST	NIST	NIST	NIST	NIST
V	NIST	NIST	NIST	NIST	NIST
Cr	NIST	NIST	NIST	NIST	NIST
Mn	NIST	NIST	NIST	NIST	NIST
Fe	NIST	NIST	NIST	NIST	NIST
Co	NIST	NIST	NIST	NIST	NIST
Ni	NIST	NIST	NIST	NIST	NIST
Cu	NIST	NIST	NIST	NIST	NIST
Zn	D&F	D&F	D&F	G	U=1
Ga	AEL	AEL	AEL	AEL	G
Ge	NIST	NIST	NIST	NIST	NIST
As	MMD	MMD	KUR	AEL	AEL
Se	KUR	KUR	KUR	AEL	AEL
Br	D&F	D&F	D&F	AEL	AEL
Kr	NIST	NIST	NIST	NIST	NIST
Rb	D&F	D&F	D&F	G	U=1
Sr	MMD	MMD	FAK	D&F	G
Y	MMD	MMD	MMD	G	AEL
Zr	D&F	MMD	D&F	D&F	G
Nb	KUR	KUR	KUR	AEL	AEL
Mo	NIST	NIST	NIST	NIST	NIST
Tc	KUR	KUR	KUR	U=1	U=1
Ru	KUR	KUR	KUR	G	U=1
Rh	KUR	KUR	KUR	G	U=1
Pd	KUR	KUR	KUR	G	U=1

	I	II	III	IV	V
Ag	D&F	D&F	D&F	G	U=1
Cd	D&F	D&F	D&F	AEL	U=1
In	D&F	D&F	D&F	D&F	AEL
Sn	KUR	KUR	KUR	AEL	AEL
Sb	KUR	KUR	KUR	AEL	AEL
Te	KUR	KUR	KUR	AEL	AEL
I	D&F	D&F	KUR	G	U=1
Xe	D&F	D&F	D&F	D&F	U=1
Cs	D&F	D&F	D&F	U=1	U=1
Ba	D&F	D&F	FAK	D&F	U=1
La	NIST	NIST	COW	NIST	NIST
Ce	NIST	NIST	COW	NIST	NIST
Pr	NIST	NIST	COW	NIST	NIST
Nd	NIST	NIST	COW	NIST	G
Pm	NIST	NIST	COW	NIST	G
Sm	NIST	NIST	COW	NIST	G
Eu	NIST	NIST	COW	NIST	G
Gd	NIST	NIST	COW	NIST	G
Tb	NIST	NIST	COW	NIST	G
Dy	NIST	NIST	COW	NIST	G
Ho	NIST	NIST	COW	NIST	G
Er	NIST	NIST	COW	NIST	G
Tm	NIST	NIST	COW	NIST	G
Yb	NIST	NIST	COW	NIST	G
Lu	NIST	NIST	COW	NIST	NIST
Hf	KUR	KUR	KUR	U=1	U=1
Ta	KUR	KUR	FAK	U=1	U=1
W	KUR	KUR	FAK	U=1	U=1
Re	KUR	KUR	FAK	U=1	U=1
Os	KUR	KUR	FAK	U=1	U=1
Ir	KUR	FAK	FAK	U=1	U=1
Pt	KUR	KUR	FAK	U=1	U=1
Au	KUR	KUR	KUR	U=1	U=1
Hg	D&F	D&F	D&F	D&F	U=1
Tl	KUR	KUR	KUR	AEL	G
Pb	D&F	D&F	D&F	D&F	D&F
Bi	MMD	MMD	MMD	AEL	AEL
Po	KUR	FAK	FAK	U=1	U=1
At	FAK	FAK	FAK	U=1	U=1
Rn	KUR	FAK	FAK	U=1	U=1
Fr	FAK	FAK	FAK	U=1	U=1
Ra	KUR	KUR	FAK	U=1	U=1
Ac	KUR	KUR	KUR	U=1	U=1
Th	KUR	FAK	FAK	U=1	U=1
Pa	KUR	FAK	FAK	U=1	U=1
U	KUR	FAK	FAK	U=1	U=1

Key:

NIST	Calculated from NIST compilations (see Table 3).
MMD	*Atomic Energy Level* calculation of Dworetsky (priv. comm.)
D+F	Drawin & Felenbok (1965)
KUR	*Atomic Energy Level* calculation of Kurucz (1970)
FAK	Fake partition functions of Kurucz (1970)
AEL	Approximate values based on *Atomic Energy Levels* (see also Table 3).
G	Ground state value from *Atomic Energy Levels* or Allen (1973)
U=1	No data found, assumed U=1.

Table 3: Additional Atomic Energy Level Sources

		Reference
B	IV	Baskin & Stoner (1978b)
Si	V	Martin & Zalubas (1983)
Sc	IV	Baskin & Stoner (1978a)
Sc	V	Baskin & Stoner (1978a)
Ti	V	Baskin & Stoner (1978a)
Co	IV	Sugar & Corliss (1981)
Co	V	Sugar & Corliss (1981)
Cu	IV	Sugar & Musgrove (1990)
Cu	V	Sugar & Musgrove (1990)
Ga	I	Johansson & Litzen (1967)
Ga	II	Isberg & Litzen (1985)
Ga	III	Isberg & Litzen (1986)
Ga	IV	Joshi et al. (1973)
Kr	IV	Sugar & Musgrove (1991)
Kr	V	Sugar & Musgrove (1991)

11 Adopted solar abundances

The table below gives the adopted solar abundances used by the code for continuous opacity and are available in `and_grev.cmd` located in the `uclsyn_dir` directory. Most of the values are from Anders & Grevesse (1989), with a few updated as indicated in the table.

Table 4: adopted solar abundances

EL	log A		EL	log A		EL	log A	EL	log A	
H	12.00		Cr	5.67		Ag	0.94	Yb	1.08	
He	10.99		Mn	5.39		Cd	1.77	Lu	0.76	
Li	1.16		Fe	7.54	[2]	In	1.66	Hf	0.88	
Be	1.15		Co	4.92		Sn	2.00	Ta	0.00	
B	2.60		Ni	6.25		Sb	1.00	W	1.11	
C	8.60	[1]	Cu	4.21		Te	2.25	Re	0.26	
N	8.00	[1]	Zn	4.60		I	1.51	Os	1.45	
O	8.94	[1]	Ga	2.88		Xe	2.19	Ir	1.35	
F	4.56		Ge	3.41		Cs	1.12	Pt	1.80	
Ne	8.09		As	2.39		Ba	2.13	Au	1.01	
Na	6.33		Se	3.35		La	1.22	Hg	1.27	
Mg	7.58		Br	2.63		Ce	1.55	Tl	0.90	
Al	6.47		Kr	3.21		Pr	0.71	Pb	1.85	
Si	7.55		Rb	2.60		Nd	1.50	Bi	0.71	
P	5.45		Sr	2.90		Pm	1.50	Po	1.50	[3]
S	7.21		Y	2.24		Sm	1.01	At	1.50	[3]
Cl	5.50		Zr	2.60		Eu	0.51	Rn	1.50	[3]
Ar	6.56		Nb	1.42		Gd	1.12	Fr	1.50	[3]
K	5.12		Mo	1.92		Tb	0.00	Ra	1.50	[3]
Ca	6.36		Tc	1.50	[3]	Dy	1.10	Ac	1.50	[3]
Sc	3.10		Ru	1.84		Ho	0.26	Th	0.23	
Ti	4.99		Rh	1.12		Er	0.93	Pa	1.50	[3]
V	4.00		Pd	1.69		Tm	0.00	U	0.00	

Key:
[1] Grevesse (1991)
[2] Biémont et al. (1991)
[3] No value listed.

References

[1] Achmad, L., de Jager, C. & Nieuwenhuijzen, H., 1991, *Astron. Astrophys.,* **250,** 445.

[2] Alecian, G., 1982, *Astron. Astrophys.,* **107,** 61.

[3] Allen, C.W., 1973, *Astrophysical Quantities,* 3rd Edition, Athlone Press, London.

[4] Anders, E. & Grevesse, N., 1989, *Geochimica et Cosmochimica Acta,* **53,** 197.

[5] Barklem, P.S., Anstee, S.D & O'Mara, B.J., 1998, Publications of the Astronomical Society of Australia, 15, 336.

[6] Barklem, P.S., O'Mara, P.J., & Ross, J.E., 1998, *Mon. Not. R. Astron. Soc.,* **296,** 1057.

[7] Barklem, P.S. & O'Mara, B.J., 1997, *Mon. Not. R. Astron. Soc.,* **290,** 102.

[8] Barnard, A.J., Cooper, J., Shamey, L.J., 1969, *Astron. Astrophys.,* **1,** 28.

[9] Barnard, A.J., Cooper, J., Smith, E.W., 1974, *J. Quant. Spectr. & Rad. Transfer,* **14,** 1025.

[10] Barnard, A.J., Cooper, J., Smith, E.W., 1975, *J. Quant. Spectr. & Rad. Transfer,* **15,** 429.

[11] Bashkin, S. & Stoner, J.O., 1978a, *Atomic Energy Level & Grotrian Diagrams, Vol. 1: Hydrogen I - Phosphorous XV, Addenda*, North-Holland Publishing Company, Amsterdam.

[12] Bashkin, S. & Stoner, J.O., 1978b, *Atomic Energy Level & Grotrian Diagrams, Vol. 1: Sulfur I - Titanium XXII*, North-Holland Publishing Company, Amsterdam.

[13] Biémont, E., Baudoux, M., Kurucz, R.L., Ansbacher, W. & Pinnington, E.H., 1991, *Astron. Astrophys.,* **249,** 539.

[14] Castelli, F., Gratton, R.G., Kurucz, R.L., 1997, *Astron. Astrophys.,* **318,** 841.

[15] Drawin, H.W. & Felenbok, P., 1965, *Data for Plasmas in Local Thermodynamic Equilibrium,* Gauthier-Villars, Paris.

[16] Edmonds, F.N., Schlüter, H. & Wells, D.C., 1967, *Mem. Roy. Astron. Soc.,* **71,** 271.

[17] Foley, H.M., 1946, *Phys. Rev.,* **69,** 616.

[18] Fraudenstein, S.J. & Cooper, J., 1978, *Astrophys. J.,* **224,** 1079.

[19] Gray, D.F., 1976, *The Observation and Analysis of Stellar Photospheres,* Wiley.

[20] Gray, D.F., 1992, *The Observation and Analysis of Stellar Photospheres,* 2nd edition, Cambridge University Press.

[21] Grevesse, N., 1991, In: *Evolution of Stars: The Photospheric Abundance Connection, IAU Symp. 145,* Eds. Michaud, G. & Tutukov, A., Kluwer, Dordecht, p.63.

[22] Griem, H.R., 1968, *Phys. Rev.,* **165,** 258.

[23] Hubeny, I., Lanz, T. & Jeffery, C.S., 1994, *Newslet. Anal. Astron. Spectra (CCP7),* **20,** 30.

[24] Isberg, B. & Litzen, U., 1985, *Physica Scripta,* **31,** 533.

[25] Isberg, B. & Litzen, U., 1986, *Physica Scripta,* **33,** 420.

[26] Johansson, I. & Litzen, U., 1967, *Arkiv Fur Fysik,* **34,** 573.

[27] Joshi, Y.N., Bhatia, K.S. & Jones, W.E., 1973, *Spectrochimica Acta,* **28B,** 149.

[28] Kurucz, R.L., 1970, *ATLAS: A Computer Program for Calculating Model Stellar Atmospheres,* SAO Special Report 309.

[29] Kurucz, R.L., 1979, *Astrophys. J. Suppl.,* **40,** 1.

[30] Kurucz, R.L., 1988, *Semiempirical Calculation of gf Values for the Iron Group, Commission 14 on Atomic and Molecular Spectroscopy, IAU General Assembly, Baltimore.*

[31] Lemke, M., 1997, *Astron. Astrophys. Suppl.,* **122,** 285.

[32] Lindholm, E., 1945, *Ark. Mat. Astron. Fys.,* **32A,** No. 17.

[33] Magain, P., 1984, *Astron. Astrophys.,* **134,** 189.

[34] Martin, W.C. & Zalubas, R., 1983, *J. Phys. Chem. Ref. Data,* **12,** 232.

[35] Martin, W.C., Zalubas, R. & Hagan, L., 1978, *Atomic Energy Levels - The Rare-Earth Elements,* NSRDS-NBS 60.

[36] Moore, C.E., 1949, *Atomic Energy Levels,* NSRDS-NBS 35 Vol. 1, Reissued December 1971.

[37] Moore, C.E., 1952, *Atomic Energy Levels,* NSRDS-NBS 35 Vol. 2, Reissued December 1971.

[38] Moore, C.E., 1958, *Atomic Energy Levels,* NSRDS-NBS 35 Vol. 3, Reissued December 1971.

[39] Peach, G., 1970, *Mem. Roy. Astron. Soc.,* **73,** 1.

[40] Peterson, D.M., 1969, SAO Special Report 293.

[41] Smith, K.C., 1992, PhD Thesis, University of London, Chapter 5.

[42] Sugar, J. & Corliss, C., 1981, *J. Phys. Chem. Ref. Data,* **10,** 1097.

[43] Sugar, J. & Musgrove, A., 1990, *J. Phys. Chem. Ref. Data,* **19,** 527.

[44] Sugar, J. & Musgrove, A., 1991, *J. Phys. Chem. Ref. Data,* **20,** 829.

[45] Vidal, C.R., Cooper, J. & Smith, E.W., 1973, *Astrophys. J. Suppl.,* **25,** 37.

[46] Warner, B., 1967, *Mon. Not. R. Astron. Soc.,* **136,** 81.

[47] Wiese, W.L., Smith, M.W., Glennon, B.M., 1966, *Atomic Transition Probabilities I, H-Ne* (NSRDS-NBS 4).