CORONAL DIAGNOSTIC SPECTROMETER
SoHO

CDS SOFTWARE NOTE No. 52

Version 3 23 April 1999

A Set of Diagnostic Routines for use with the CHIANTI Database

Peter Young
DAMTP, Cambridge University
Silver Street
Cambridge CB3 9EW
UK

P.R.Young@damtp.cam.ac.uk

1 Overview

The primary aim of the CHIANTI project was to place in an easily accessible database all the atomic data required for the interpretation of emission lines in UV and X-ray spectra. The database was distributed with a few simple routines for the analysis of density and temperature diagnostics, and the creation of synthetic spectra.

This software note describes a new set of routines that extract extra information from the CHIANTI database that may prove useful in the analysis of spectra. Amongst other things, the routines can do the following:

- A density diagnostic analysis routine that allows the input of observed intensities with error bars, to yield densities with error estimates (routine: dens_plotter.pro).
- The extraction of the contribution function (or "G-of-T") for any transition (routine: g_of_t.pro)
- Compute the level balance for any ion (routine: show_pops.pro).
- Allow photoexcitation and stimulated emission to be included in the level balance (all routines).
- Compute the atomic parameters for a simple emission measure analysis (routine: integral_calc.pro).

The routines are described in more detail in Sect. 4. First of all, I give in Sect. 2 the theory behind the interpretation of optically-thin emission lines which serves to set out the notation that I use. Sect. 3 describes how to set up the routines to read the CHIANTI database.

2 Theory and definitions

In the 'standard model' for interpreting line intensities from the solar atmosphere, there are three fundamental assumptions that serve to simplify the problem considerably:

1. the plasma is in a steady state;
2. atomic processes affecting the ionisation state of an element can be separated from those affecting the level balance within an ion;
3. all lines are optically thin.

The atomic data contained in the CHIANTI database are particularly suited to the analysis of emission lines via this model, and the following discussion outlines this approach. No attempt is made to discuss non-equilibrium conditions.

With the first of the assumptions, the population of ions lying in a given state is constant and so the number of ions leaving this state per unit time must exactly balance the number arriving into that state. If we denote the number of transitions leaving the state i to a state j taking place per unit time per unit volume by $\alpha_{\rm ij}$, then steady state implies

$$N_{\rm i} \sum_{\rm j \neq i} \alpha_{\rm ij} = \sum_{\rm j \neq i} N_{\rm j} \alpha_{\rm ji}. \tag{1}$$

Setting

$$\alpha_{\rm ii} = -\sum_{\rm j\neq i} \alpha_{\rm ij} \tag{2}$$

we have

$$\sum_{\rm j} N_{\rm j}\alpha_{\rm ji} = 0 \tag{3}$$

for each state i and, as the coefficients $\alpha_{\rm ji}$ are independent of the state populations, we have a set of linear equations to solve for the $N_{\rm i}$.

To proceed, we first write

$$N_{\rm i} = n_{\rm i}.\frac{N({\rm X}^{+m})}{N({\rm X})}.\frac{N({\rm X})}{N({\rm H})}.\frac{N({\rm H})}{N_{\rm e}}.N_{\rm e} \tag{4}$$

where $n_{\rm i}$ is the fraction of ions $\rm X^{+m}$ lying in the state i, $N({\rm X^{+m}})/N({\rm X})$ is the fraction of the element X in the ionisation state $\rm X^{+m}$; $N({\rm X})/N({\rm H})$ is the abundance of the element X relative to hydrogen; and $N({\rm H})/N_{\rm e}$ is the ratio of hydrogen to free electrons.

Now our second assumption means that the processes that affect the ionisation state of the plasma do not affect the quantity $n_{\rm i}$. Eq. 3 thus becomes

$$\sum_{\rm j} n_{\rm j}\alpha_{\rm ji} = 0 \tag{5}$$

where the $\alpha_{\rm ji}$ only include those processes that affect the level balance of the ion. For the basic CHIANTI model these processes are simply electron excitation and de-excitation, and radiative decay, and so

$$\alpha_{\rm ij} = {\rm C}_{\rm ij} + A_{\rm ij} \tag{6}$$

where C is the generalised electron excitation–de-excitation rate, and $A_{\rm ij}$ is the radiative decay rate which is zero for i<j (the 'A-values' are contained in the CHIANTI `.wgfa` files). C is given explicitly by

$${\rm C}_{\rm ij} = \begin{cases} N_{\rm e}q_{\rm ij} & {\rm i < j} \\ N_{\rm e}\frac{\omega_{\rm j}}{\omega_{\rm i}}\exp\left(\frac{\Delta E}{kT}\right) q_{\rm ji} & {\rm i > j} \end{cases} \tag{7}$$

where $\omega_{\rm i}$ is the statistical weight of level i, k is Boltzmann's constant, T the electron temperature, and $q_{\rm ij}$ the electron excitation rate coefficient which is given by:

$$q_{\rm ij} = 2.172\times 10^{-8}\left(\frac{I_\infty}{kT}\right)^{1/2}\exp\left(-\frac{\Delta E}{kT}\right)\frac{\Upsilon_{\rm ij}}{\omega_{\rm i}} \qquad [{\rm cm^3\ s^{-1}}] \tag{8}$$

where I_∞ is the Rydberg energy (13.61 eV), and $\Upsilon_{\rm ij}$ is the thermally-averaged collision strength for the $\rm i \rightarrow j$ excitation. The upsilons are derived from the data in the CHIANTI `.splups` files.

Solving Eq. 5 is performed by the CHIANTI routine `populate.pro`, and gives the fraction of ions in the state i. Going back to Eq. 4, we write

$$N_{\rm i} = 0.83\,F(T)\,Ab({\rm X})\,N_{\rm e}\,n_{\rm i}, \tag{9}$$

where $F(T)$ is the ionisation fraction (independent of $N_{\rm e}$ in current ion balance calculations), Ab(X) the abundance of the element relative to hydrogen, and the ratio of hydrogen to free electrons has been taken as 0.83, as hydrogen and helium are completely ionised for temperatures $T \gtrsim 10^4$ K.

The *emissivity* of the emission line resulting from a j-to-i radiative decay is defined as

$$\epsilon_{\mathrm{ij}} = \Delta E \, N_{\mathrm{j}} \, A_{\mathrm{ji}} \tag{10}$$

and has units of erg $\mathrm{cm}^{-3}\,\mathrm{s}^{-1}$. Often the alternative notation ϵ_λ will be used where λ is the wavelength of the emitted radiation in Angstroms (Å), and $\lambda = 1.986 \times 10^{-8}/\Delta E$ for ΔE in ergs. We will also define the *ion emissivity* as

$$\varepsilon_{\mathrm{ij}} = \Delta E \, n_{\mathrm{j}} \, A_{\mathrm{ji}}. \tag{11}$$

In order to relate the emissivity to the actual observed intensity of a line, we make use of the third assumption, which tells us that the intensity is proportional to the emissivity of the plasma, and so

$$P_\lambda = \int \epsilon_\lambda \, \mathrm{d}V, \tag{12}$$

where P_λ is the power in an observed line (units: erg s^{-1}), and $\mathrm{d}V$ is a volume of plasma with temperature T and density N_{e}.

Expanding ϵ_λ using Eqs 9 and 10 gives

$$P_\lambda = 0.83 \, \Delta E \, Ab(\mathrm{X}) \int F(T) \, n_{\mathrm{j}} A_{\mathrm{ji}} N_{\mathrm{e}} \, \mathrm{d}V. \tag{13}$$

An important feature of emission measure studies is to isolate those lines for which $n_{\mathrm{j}} A_{\mathrm{ji}} \sim N_{\mathrm{e}}$. By analysing only such lines, we are essentially separating the determination of the emission measure from the determination of the plasma density. If the lines all had different density dependencies, then it would be necessary to determine the density variation with temperature before finding the emission measure. If the $n_{\mathrm{j}} A_{\mathrm{ji}} \sim N_{\mathrm{e}}$ relation is assumed then we write

$$P_\lambda = \Delta E \, Ab(\mathrm{X}) \int G_\lambda(T) \, N_{\mathrm{e}}^2 \, \mathrm{d}V \tag{14}$$

where

$$G_\lambda(T) = 0.83 \, F(T) \, \frac{n_{\mathrm{j}} A_{\mathrm{ji}}}{N_{\mathrm{e}}} \tag{15}$$

which is the so-called "G-of-T" function.

On account of the ionisation fraction $F(T)$ this function is sharply peaked, and a common approximation (e.g., Pottasch [3], Jordan & Wilson [1]) is to assume that $G(T)$ has a constant value over a narrow temperature interval around $G(T_{\mathrm{max}})$, where T_{max} is the temperature of maximum ionisation for the ion. Here we will use the temperature of maximum emission or T_{mem} which is the temperature at which G_λ has its maximum. Defining

$$G_{\lambda,0}(T) = \begin{cases} C_\lambda & |\log T - \log T_{\mathrm{mem}}| < 0.15 \\ 0 & |\log T - \log T_{\mathrm{mem}}| > 0.15 \end{cases} \tag{16}$$

we require that

$$\int G_\lambda(T) dT = \int G_{\lambda,0}(T) dT \tag{17}$$

so

$$C_\lambda = \frac{\int G_\lambda(T) dT}{T_{\mathrm{mem}}(10^{0.15} - 10^{-0.15})}. \tag{18}$$

Our expression for P_λ thus becomes

$$P_\lambda = \Delta E\, Ab(\mathrm{X})\, C_\lambda\, EM(V) \tag{19}$$

where

$$EM(V) = \sum_{\mathrm{i}} \left(\int_{V_\mathrm{i}} N_\mathrm{e}^2 \mathrm{d}V \right) \tag{20}$$

is the *volume emission measure*. Each volume V_i contains plasma with temperatures such that $|\log T - \log T_\mathrm{mem}| < 0.15$, and the sum over i is required in case there are distinct regions along the line of sight that satisfy this condition on T.

To derive $EM(V)$ we need simply the measured radiated power, P_λ, and compute the atomic parameters that make up C_λ. In practice, the measured quantity will typically be in energy flux, F, units (erg cm^{-2} s^{-1}) or intensity, I, units (erg cm^{-2} sr^{-1} s^{-1}) and so

$$P_\lambda = \Phi_\mathrm{em} \mathcal{F} = 4\pi \Phi_\mathrm{em} I \tag{21}$$

where Φ_em is the emitting area projected onto a plane perpendicular to the observation axis. We then have

$$EM(V) = \int N_\mathrm{e}^2 \mathrm{d}V = \int N_\mathrm{e}^2 \Phi_\mathrm{em} \mathrm{d}s \tag{22}$$

and so

$$\mathcal{F} = 4\pi I = \Delta E\, Ab(\mathrm{X})\, C_\lambda\, EM(s) \tag{23}$$

where $EM(s) = \int N_\mathrm{e}^2 \mathrm{d}s$ is the *column emission measure*.

Another definition useful in the analysis of solar full-disk data or stellar data comes through treating the volume element $\mathrm{d}V$ as a spherical shell and so $\mathrm{d}V = 4\pi R^2 \mathrm{d}h$ where R is the distance from the centre of the emitting object. If the spherical shell lies close to the star's surface then only one half of the emitted radiation will leave the star and so we have

$$2 \times F = 2 \times 4\pi I = \Delta E\, Ab(\mathrm{X})\, C_\lambda\, EM(h) \tag{24}$$

where $EM(h)$ is the *emission measure over height*. Jordan and co-workers (see, e.g., Jordan & Wilson [1], Jordan & Brown [2]) utilise this definition and an assumption of spherical symmetry to deduce energy balance relations in solar and stellar atmospheres.

3 Setting up the routines

Currently there are three ways of interfacing with the CHIANTI database, and for each I describe how to set up the new routines.

3.1 Through the CDS software

The routines are already in the CDS software tree, and so you can start using them immediately. *However*, after starting a CDS idl session, you *must* type in `use_dere_chianti`, otherwise an error will be generated.

3.2 Through the solarsoft software

If you do not have the CDS software as part of your solarsoft tree, then you will not have the extra routines described here and so you will have to download them into a separate directory. To do this, use a web browser to access:

```
ftp://ftp.damtp.cam.ac.uk/pub/for_pry10/chianti/
```

and click on the file:

```
extra_routines.tar.gz
```

after downloading it, place the file in a suitable directory (e.g., `/home/extra_routines`), and then do the following operations:

```
gunzip extra_routines.tar.gz
tar xvf extra_routines.tar
```

You now have the routines, but you need to tell IDL where it can find them. To do this, find out where your IDL_PATH is defined (usually the .login or .cshrc (or equivalent) file), and add the following to the definition:

```
:\+~/extra_routines
```

Once this has been done, you will be able to run the routines from a solarsoft idl session.

If you do have the CDS section of the solarsoft tree installed, then you will have access to all the routines. Unlike with the CDS software, you do not need to type `use_dere_chianti` in order to use the routines.

3.3 Normal IDL

If you have the CHIANTI database, and have been using the routines `density_ratios.pro`, `temperature_ratios.pro` or `synthetic.pro` through standard idl, then you simply have to follow the steps outlined in the previous section to download the routines into a directory in your account, and then add the directory to your IDL_PATH.

4 The primary routines

I've divided the routines into *primary* and *secondary* routines. The secondary ones are called by some of the primary routines, and chances are that you won't have to use them too often. They are described in Sect. 5.

All of the routines have headers which give more detailed information about how they work. This header can be read in the normal IDL way through, e.g., `doc_library,'dens_plotter'`.

4.1 pop_plot.pro

This routine plots the values of

$$10^{20}\,\Delta E\,n_{\mathrm{j}}A_{\mathrm{ji}}/N_{\mathrm{e}} \tag{25}$$

against N_{e}. As discussed in Sect. 2, if we only study lines in the emission measure analysis for which this quantity is independent of density, then the derived emission measure is independent of the plasma density.

Example: For Fe XIII, select a line/blend from lines in the range 200 to 205 Å

```
pop_plot,26,13,wrange=[200,205]
```

Note how no single line shows zero density dependence, and so care should be taking in using Fe XIII in emission measure analyses. Compare with Fe XVI:

```
pop_plot,26,13,wrange=[330,370]
```

where both the 335 and 360 lines are OK.

4.2 integral_calc.pro

This routine calculates C_λ, defined in Eq. 18. It displays both this value and the values of $\Delta E\ C_\lambda$ and $4\pi/\Delta E\ C_\lambda$. For lines for which $n_{\mathrm{j}}A_{\mathrm{ji}} \sim N_{\mathrm{e}}$, C_λ is insensitive to N_{e}, but for other lines N_{e} should be specified. Note that for blended lines only $\sum \Delta E\ C_\lambda$ and $4\pi/\sum \Delta E\ C_\lambda$ are output. The routine also outputs the T_{mem} of the lines, accurate to 0.02 dex.

Example: Work out C_λ for the Fe XIII lines between 200 and 205 Å at a density of 10^9 cm^{-3}.

```
integral_calc,26,13,wrange=[200,205],dens=9.
```

From Eq. 23, an observed line intensity of 100 erg cm^{-2}s^{-1}sr^{-1} for the 202.044 line implies a column emission measure of $EM(s) = 100 \times 1.305 \times 10^{20}/Ab(Fe)$, where 1.305×10^{20} is taken from `4pi/DE*C_lambda` column of the output.

For Fe XIV, one can do:

```
integral_calc,26,14,wrange=[210,220],dens=9.
```

and so to get the same column emission measure for Fe XIV, an intensity of $100\times1.305\times10^{20}/2.490\times 10^{20} = 52.4$ erg cm^{-2}s^{-1}sr^{-1} is required.

4.3 dens_plotter.pro

A widget-based routine that allows the thorough investigation of density sensitive ratios. Observed line intensities can be input and densities are derived. Excitation by a background radiation field is allowed for.

Example: to study density sensitive ratios of Fe XIII, simply type in

```
dens_plotter,26,13
```

Try inputting some line intensities and errors from the SERTS-89 spectrum (Thomas & Neupert [4]), and comparing the derived densities with those found by Young, Landi & Thomas [5] in Table 20.

4.4 show_pops.pro

Gives percentage level populations for all levels within the specified ion that have populations greater than 0.01%.

Example: Compute level populations for Fe XIII at a density of 10^{10} cm^{-3}:

```
show_pops,26,13,dens=10
```

4.5 processes.pro

For a specified level within an ion, this routine gives the contributions of the different processes that populate and de-populate the level.

Example: Level 13 of Fe XIII at a density of 10^{10} cm^{-3}:

```
processes,26,13,dens=10,level=13
```

4.6 g_of_t.pro

Eq. 15 gives my definition of the "G-of-T" or *contribution* function. In it's default setting the `g_of_t` routine actually calculates:

$$\Delta E \; G_\lambda(T) = 0.83 \, \Delta E \; F(T) \, \frac{n_\mathrm{j} A_\mathrm{ji}}{N_\mathrm{e}}$$

which is more useful when considering blends of lines at different wavelengths. The ΔE can be 'disabled' with the `/no_de` keyword. It is also useful to multiply the above function by the element abundance, and this is accomplished with the `/abund` keyword. The output function is tabulated over $4.0 \le \log T \le 8.0$ at 0.1 dex intervals. For smaller intervals, see the `ion_interp` routine.

Examples:

```
result=g_of_t(26,13,dens=9.)
result=g_of_t(26,13,wrange=[200,205],/choose,/abund)
result=g_of_t(26,13,/no_de)
```

One can also use this routine to derive the $T_{\rm mem}$ of the emission line, by way of the `ion_interp.pro` routine, e.g.,

```
result=g_of_t(26,13,dens=9.)
ion_interp,t,result,ti,g_ti,10
print,ti(where(g_ti eq max(g_ti)))
```

`result` is tabulated at 0.1 dex intervals in temperature. `ion_interp` interpolates `result` and in this case gives it at 0.01 dex intervals.

5 The secondary routines

These routines are called by the routines above.

5.1 emiss_calc.pro

Calculates the ion emissivity (Eq. 11) for all transitions within the CHIANTI model of the ion. The returned data is in the form of a structure. The default is to calculate emissivities for temperatures $T_{\rm max}$ and log $T_{\rm max} \pm 0.15$, and densities log $N_{\rm e} = 8.0, 8.5, 9.0, ..., 12.0$.

Example:

```
emiss=emiss_calc(26,13)
```

5.2 pop_solver.pro

A major overhaul of the populate.pro routine supplied in the original CHIANTI routine package which is significantly quicker, and which also allows excitation and de-excitation by a background radiation field and proton collisions. The latter are accounted for by writing Eq. 6 as

$$\alpha_{\rm ij} = {\rm C_{ij}} + {\rm A_{ij}} + {\rm C^p_{ij}} \tag{26}$$

where

$$\mathrm{A_{ij}} = \begin{cases} W A_{\rm ji} \frac{\omega_{\rm j}}{\omega_{\rm i}} \frac{1}{\exp(\Delta E/kT_0)-1} & \mathrm{i < j} \\ A_{\rm ij} \left[1 + W \frac{1}{\exp(\Delta E/kT_0)-1}\right] & \mathrm{i > j} \end{cases} \tag{27}$$

and T_0 is the black-body temperature of the background radiation field (6000 K for the Sun), and W is the radiative dilution factor given by

$$W = \frac{1}{2}\left[1 - \left(1 - \frac{1}{R^2}\right)^{1/2}\right]. \tag{28}$$

R is the distance from the Sun in solar radius units and so, at the solar surface, $R = 1$ and $W = 1/2$, while at infinity $W = 0$.

$\mathrm{C^p}$ in Eq. 26 is given by

$$\mathrm{C^p_{ij}} = \begin{cases} 0.83 N_\mathrm{e} q^\mathrm{p}_\mathrm{ij} & \mathrm{i < j} \\ 0.83 N_\mathrm{e} \frac{\omega_\mathrm{j}}{\omega_\mathrm{i}} \exp\left(\frac{\Delta E}{kT}\right) q^\mathrm{p}_\mathrm{ji} & \mathrm{i > j} \end{cases} \tag{29}$$

where the q^p are the proton rate coefficients, and the factor 0.83 is the ratio of protons to electrons (a constant for $T \gtrsim 10^4$ K).

All the routines described in this document make use of this routine rather than populate.pro.

5.3 emiss_select.pro

Allows the selection of lines/blends from the emiss structure created by `emiss_calc.pro`. This routine is useful if you want to access the emissivities of lines directly, e.g.,

```
emiss=emiss_calc(26,13)
em202=emiss_select(emiss,wra=[200,205],sel_ind=sel_ind)
```

In this example, calling emiss_select yields a widget that allows one to select a line/blend from the 200–205 Å range. The emissivities of this line blend will be contained in `em202`, while the `emiss` index/indices of this line/blend will be contained in `sel_ind`.

5.4 ion_interp.pro

When reading the ionisation equilibrium files, you will receive an array with absolute (as opposed to log) ion fractions tabulated at 0.1 dex intervals from $\log T = 4.0$ to 8.0. A common need is to interpolate this data and obtain the ion fraction tabulated at smaller intervals. As the ion fractions are generally sharply peaked, normal interpolation will lead to negative ion fractions at several temperatures, and so a more satisfactory method is to interpolate the log of the ion fraction. However, you need to take the log of only the non-zero values of the ion fraction.

The several lines of code required to perform the interpolation are straightforward but irritating (when typed on many occasions!), and so this routine performs the task.

Example: Use g_of_t to create a $G(T)$ function for one of the Fe XIII lines,

```
result=g_of_t(26,13,dens=9.)
ion_interp,t,result,ti,g_ti,5
```

The $G(T)$ function is now tabulated at 0.02 dex intervals. Note that if t is not specified, it is assumed to be a vector going from 4.0 to 8.0 in 0.1 dex intervals.

5.5 choose_ioneq.pro

Some problems were encountered at institutions where CHIANTI had not been updated to v1.01. Because v1.0 did not contain the Arnaud & Raymond (1992) ionisation balance data file, several of the above routines crashed. I've now delegated the task of selecting the ion balance data to this routine which checks if the Arnaud & Raymond file exists.

References

[1] Jordan, C. & Wilson, R. (1971). In: *Physics of the Solar Corona*, (ed. C. J. Macris), p. 199, Reidel, Dordrecht

[2] Jordan, C. & Brown, A. (1981). In: *Solar Phenomena in Stars and Stellar Systems*, (eds. R. M. Bonnet and A. K. Dupree), p. 199, Reidel, Dordrecht

[3] Pottasch, S. R. (1964). Space Sci. Rev. 3, 816

[4] Thomas, R. J. & Neupert, W. M. (1994). ApJS 91, 461

[5] Young, P. R., Landi, E., & Thomas, R. J. (1998). AA 329, 291