
CHIANTI

An Astrophysical Database for Emission Line Spectroscopy

CHIANTI TECHNICAL REPORT No. 4

Creating scaled upslon (SCUPS) files for CHIANTI

Version 1.2, 17 August 2017, Peter Young
Version 1.1, 4 August 2017, Peter Young
Version 1.0, 3 February 2017, Peter Young

1 Overview

Version 8 of CHIANTI was released during 2015 September, and the formats of the UPSDAT and SPLUPS files were changed. The new files are named UPS and SCUPS, and this report describes how the SCUPS file is created from the UPS file. The formats of the two files are described in CHIANTI Technical Report No. 13.

2 Notation

The effective collision strengths are denoted by Υ (or “upsilons”) and temperatures by T . The Burgess & Tully (1992) scaling method yields scaled temperatures, T_s , and scaled upsilons, Υ_s . If there are n Υ_s values, then these are denoted by $\Upsilon_{s,1}, \Upsilon_{s,2} \dots \Upsilon_{s,n}$.

3 Creating the SCUPS data

The critical change in how the SCUPS data are created is that scaled upsilons are no longer fit with a spline. Instead, the scaled upsilons (and temperatures) are simply written directly to the SCUPS file. When the CHIANTI software needs a Υ value at a particular temperature, then a spline is run through the values in the SCUPS file to yield the appropriate Υ_s , which is then de-scaled in the usual way (the routine DESCALE_SCUPS performs the de-scaling).

A linear extrapolation of the two lowest temperature points, $\Upsilon_{s,1}, \Upsilon_{s,2}$ is used to derive the scaled epsilon value at $T_s = 0$. This value is written to the SCUPS file.

If a transition does not have a high temperature limit point, then a linear extrapolation of the $\Upsilon_{s,n-1}$ and $\Upsilon_{s,n}$ values is performed to yield the Υ_s value at $T_s = 1$. If the transition does have a high temperature limit point, then this value is written to the SCUPS file.

In summary, if there are n Υ values for a particular transition in the original atomic data-set, then the SCUPS file will contain $n + 2$ T_s and Υ_s values for this transition.

4 Recommended procedure

The recommended procedure for creating the SCUPS file from the UPS file is to first run an automatic assessment routine called WRITE_SCUPS:

```
IDL> write_scups, ion_name
```

where `ion_name` is the name of the ion in standard CHIANTI format. This creates the file `ion_name.scups_auto` containing the scaled upsilons for all transitions except those for which some problem was found in the assessment.

The problem transitions can either be dealt with through keywords to WRITE_SCUPS (see Sect. 5), or the transitions can be manually assessed using an IDL GUI. The latter is called with

```
IDL> write_scups_gui, ion_name
```

This routine will automatically check the list of problem transitions and so the user only needs to fit these transitions. After running `WRITE_SCUPS_GUI`, the `SCUPS_AUTO` file is copied to a `SCUPS` file, and the manually-fitted transitions are appended. See Sect. 6 for more details.

5 Automatic creation of the SCUPS file (`WRITE_SCUPS`)

The call to `WRITE_SCUPS` is

```
IDL> write_scups, ion_name
```

which first checks to see if `ion_name.ups` exists, and then all transitions in the file are automatically assessed. Information from the assessment is printed to the IDL window, and an example is shown in Appendix B.

Three files are produced:

<code>ion_name.scups_auto</code>	The SCUPS file.
<code>ion_name_flags.txt</code>	The list of problem transitions, with the problem type identified (see Sect. 5.1 and 5.4)
<code>ion_name_log.txt</code>	A file giving additional information about the problem transitions.

`ion_name.scups_auto` and `ion_name_flags.txt` are used by `WRITE_SCUPS_GUI` (Sect. 6). The format for `ion_name.scups_auto` is described in CHIANTI Technical Report No. 13, and the format for `ion_name_flags.txt` is given in Sect. 5.4.

5.1 Problem transitions

If the automatic assessment finds a problem with a transition, then the transition is assigned a “problem type” (PT) index. The descriptions of these PTs are given below.

- 1 The energy for the transition is zero
- 2 At least one of the upslon values is negative
- 3 The lowest temperature point is anomalous
- 4 The highest temperature point is anomalous
- 5 At least one of the intermediate points is anomalous
- 6 The points do not tend towards the high temperature limit

PT 1 sometimes arises for transitions within a term. The atomic physicist may have only presented theoretical energies for the LS terms, thus the two fine structure levels will have been assigned the same energy. The standard solution is to introduce an energy separation usually taken from the experimental separation of the levels.

PT 2 is rare and the atomic physicist should be consulted. If the negative value is small, then simply setting it to zero may be a good solution.

PT 3 is checked by performing a linear extrapolation of the 2nd and 3rd points back to the 1st point. See Sect. 5.2 for a description of the extrapolation method.

PT 4 is checked by performing a linear extrapolation of the (n-2)th and (n-1)th points up to the nth point (note that the nth point is *not* the high temperature limit – this is checked separately for PT 6). See Sect. 5.2 for a description of the extrapolation method.

Problem type 5 is checked by considering all the points between the low and high temperature points. Considering each of these intermediate points in turn, a spline is fit to all the points except the point under consideration. The spline value at the point’s location is then compared with the actual value and if the spline value is different by a factor $(1 \pm \lambda)$ then the transition is flagged. For example, $\lambda = 0.3$ implies a $\pm 30\%$ difference.

Problem type 6 is when the scaled upsilons do not tend towards the high temperature limit point. This check uses the same extrapolation method as PTs 3 and 4, and is described in Sect. 5.2.

5.2 Extrapolation method

Problem types 3, 4 and 6 are identified by linearly extrapolating from two points to a third point, and then checking if the extrapolated point is discrepant with the actual data point. The method for creating an allowed parameter range is described here. Please refer to Figure 1 to understand the steps.

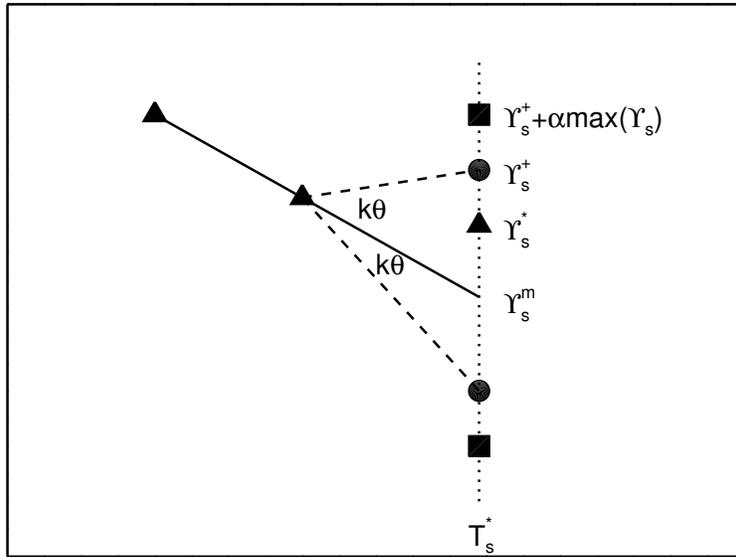


Figure 1: A schematic diagram to illustrate how the extrapolation checking procedure works.

The data point that we wish to check corresponds to a scaled temperature T_s^* and a scaled upslon, Υ_s^* . The two reference points yield a straight line with a gradient, m , which intersects T_s^* at Υ_s^m (see the solid line in Figure 1).

We now consider two additional straight lines (dashed lines in Figure 1) that pass through the reference point nearest to T_s^* . The gradients for these two points are defined as:

$$m_- = \tan(\arctan(m) - k\theta) \quad (1)$$

$$m_+ = \tan(\arctan(m) + k\theta) \quad (2)$$

where $\theta = \arctan(\max(\Upsilon_s))$, and k is a multiplicative factor (see Sect. 5.3). This definition mimics how the human eye would try to estimate an allowed range for the extrapolation, based on the Y -range of the visual plot (hence the use of $\max(\Upsilon_s)$).

These two straight lines, yield two Υ values at T_s^* that we label Υ_s^- and Υ_s^+ (circles in Figure 1); one smaller than Υ_s^* , and one larger.

We then further increase the allowed Υ range to $[\Upsilon_s^- - \alpha \max(\Upsilon_s), \Upsilon_s^+ + \alpha \max(\Upsilon_s)]$, to give the points shown as squares in Figure 1. If the actual data point, Υ_s^* , lies outside of the range defined by the squares, then a problem transition is flagged.

5.3 Extrapolation parameters

WRITE_SCUPS takes as an optional input the keyword PARAMS, which is a five element array that controls how strict the checks are for problem types 3 to 6. Please refer to Sects. 5.1 and 5.2 for more details about the extrapolation method and parameters k , α and λ .

Table 1 identifies the components of PARAMS. For normal operation, the parameter values indicated in column ‘Standard’ are used by WRITE_SCUPS. These values were adopted after trial-and-error with some data-sets to ensure that only genuinely anomalous transitions are flagged. The keyword /STRICT assigns a much more strict set of parameters that allow the user to check if any problem transitions are being missed by the standard parameters.

Table 1: The PARAMS optional input for WRITE_SCUPS.

	Parameter	Standard	/STRICT
PARAMS[0]	k for PT=3	2.5	1.0
PARAMS[1]	k for PT=4	2.5	1.0
PARAMS[2]	k for PT=6	1.5	1.0
PARAMS[3]	α for PT=3,4,6	0.1	0.0
PARAMS[4]	λ for PT=5	0.3	0.1

5.4 The flag file

If problem transitions are found, then information on these is stored in the `[ion_name]_flag.txt` output file. This has a simple 3i7 format that gives the lower and upper levels of the transition and the flag number (the problem type from Sect. 5.1). This file is read by WRITE_SPLUPS_GUI, allowing the routine to work out which routines need to be manually processed.

5.5 Plots

The routine WRITE_SCUPS optionally produces plots of the scaled upsilons that are sent to the postscript file `[ion_name]_plots.ps`.

By setting the keyword `plot_prob_type` to one of the PTs 3 to 6, then all of the problem transitions of the specified type will be plotted. (Note that PTs 1 and 2 do not require plots.)

The keyword `plot_trans_type` allows all transitions of the specified transition type to be plotted. For example `plot_trans_type=1` means all allowed transitions are plotted. Note that both good transitions and problem transitions will be plotted.

The postscript file is called `[ion_name]_plots.ps`. 2×4 plots are shown on each page. The scaled temperature is plotted from -0.05 to 1.05 , with dashed lines shown at $T_s = 0$ and 1 .

The data points are shown as '+'. If a high temperature limit point is shown and `PT=6`, then the limit is plotted with a square at $T_s = 1$ (otherwise it will be a +). The extrapolation method discussed in Sect. 5.2 results in an allowed range of Υ_s values and this range is indicated for `PT=3,4` and `6` transitions by two '*' symbols (see Figure 3).

An example of a `PT=5` problem transition is shown in Fig. 2 for a Si II transition.

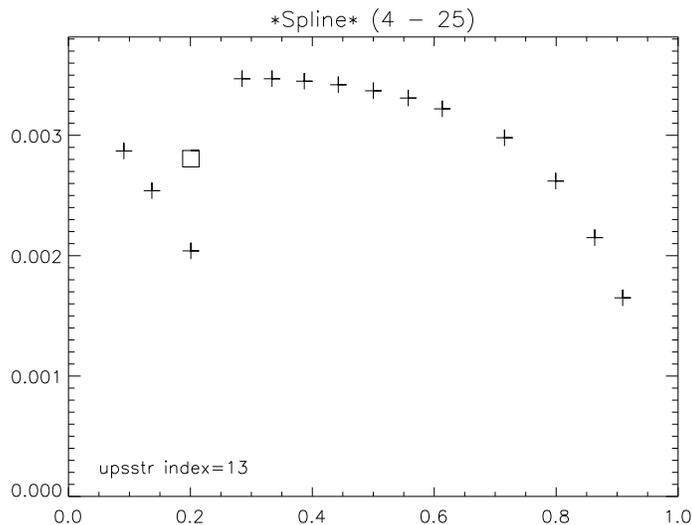


Figure 2: The `WRITE_SCUPS` output for transition 4-25 of Si II, which shows a type 5 problem. The crosses show the scaled upsilons, and the square indicates the scaled upsilong value derived by fitting a spline through all of the crosses except the third one ($T_s = 0.2$).

An example of a `PT=6` transition is shown in Fig. 3 for a transition of Mg v.

5.6 Automatic scaling of temperatures and upsilons

The original concept of the Burgess & Tully (1992) scaling method was to manually choose a scaling parameter, C , that distributed the upsilong values fairly uniformly across the $T_s=[0,1]$ interval.

In order to perform an automatic scaling within `WRITE_SCUPS`, we have the routine `BURLY_OPTIMIZE_C` that derives C . This routine begins by calling `BURLY_FIND_MID_C` which computes the C value that places the middle temperature point at T_s .

`BURLY_OPTIMIZE_C` proceeds to then vary C from this value in order to find the value that results in the smallest gradients between consecutive data points.

Note that, for the original 5 and 9-point spline methods used in `CHIANTI` versions 1 to 7, adjustment of the scaling parameter was critical to obtaining the best fit to the data points. Since fits are

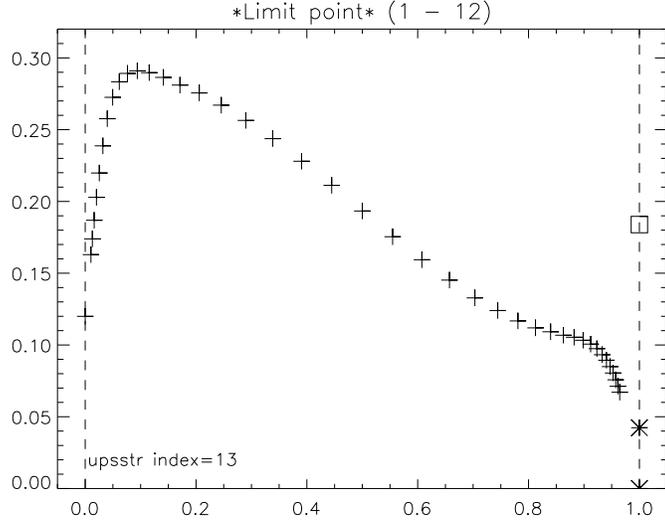


Figure 3: The WRITE_SCUPS output for transition 1–12 of Mg V, which shows a type 6 problem. The crosses show the scaled upsilons, and the square indicates the high temperature limit point. The two stars at $T_s = 1$ show the allowed range for the limit point based on extrapolating the final two scaled upsilon values.

no longer performed with the new method, then adjustment of the scaling parameter is no longer so important but the optimization procedure can be useful in avoiding some of the extrapolation problem cases (PT=3,4 and 6).

5.7 Transition type

The transition type is selected automatically by the routine BURLY_GET_TTYPE, based on the following logic. If gf value for the transition is zero, then the type is 2. If gf is non-zero and $\geq 10^{-3}$ then the type is 1; if $< 10^{-3}$ then the type is 4. At present there is no means to automatically identify a type 3 transition.

5.8 The /IGNORE_LIMIT keyword

The most common problem likely to be encountered is PT=6, when the data points do not tend to the high temperature limit (e.g., Figure 3). There are two solutions for this:

- Keep the high temperature limit and assume that there are errors in the calculation that result in the data points not tending to the limit. This is solved by using WRITE_SCUPS_GUI to remove some of the high temperature data points thus allowing a smooth transition to the limit.
- Assume the high temperature limit is wrong. This can be fixed by using the keyword /IGNORE_LIMIT that will tell WRITE_SCUPS to ignore the high temperature limits, *but only for the PT=6 transitions*.

With regard the latter it is possible that the reason the points do not tend to the limit is because the upsilons are not calculated to a high enough temperature thus the calculated upsilons may actually tend to the limit if the temperature range was larger. For example, in Figure 3, although the upsilons are tending in the opposite direction to the limit point, they may turn up again at higher temperatures.

6 Manual creation of the SCUPS file (WRITE_SCUPS_GUI)

For CHIANTI versions 1 to 7 the spline fits to the upsilons were performed manually using the routine BURLY_UPS (or variations thereof). Each transition would be loaded one-by-one; the data assessor would choose the transition type and scaling parameter, extrapolate the data to zero and one, and (if necessary) remove data points in order to get a good fit.

The user has the option to use the old BURLY_UPS routine and continue to fit 5 to 9 point splines to the data. (This requires a modification to BURLY_UPS, however, in order to read the new UPS file and write to the new SCUPS file.)

An additional (and recommended) option is to use the new routine WRITE_SCUPS_GUI, which provides a graphical means of assessing the data without a spline fit being performed.

WRITE_SCUPS_GUI automatically works out which transitions need to be fit, depending if the FLAG file (created by WRITE_SCUPS) exists. If yes, then only the transitions listed in this file will be loaded into WRITE_SCUPS_GUI. If the FLAG file does not exist (or the /NO_FLAG option is given), then every transition in the UPS file will be loaded.

The first transition is automatically loaded into WRITE_SCUPS_GUI and the fit displayed. The data assessor can then adjust the fit in three ways:

1. Adjust the number of temperature points that are displayed.
2. Select a different transition type.
3. Change the scaling parameter (C). See Sect. 6.1.

When happy with the fit, the data assessor should click on the 'WRITE and NEXT' button to move on to the next transition. If no acceptable fit is performed, then the 'SKIP' button should be pressed.

At any point the user can exit the GUI. Transitions that had been fit will be stored in the SCUPS file. If the GUI is re-started, then it will start at the point where it was previously exited.

6.1 Options for the scaling parameter

WRITE_SCUPS_GUI automatically works out the C-value, but options exist to modify this. Firstly, the text box containing the C-value is editable, and so the user can type in values.

Secondly, there are buttons below this text box that modify the method by which the C-value is computed. The 'gradient' method (described earlier) is the default. Sometimes this method can lead to points being bunched towards $T_s = 0$ or 1, so the 'maxmin' and 'minmax' methods allow an

optimization based on the separation of the points in T_s space. The ‘maxmin’ method maximizes the minimum separation between two points, while the ‘minmax’ method minimizes the maximum separation between two points. In addition there are two hybrid options that take the mean value between gradient and maxmin, and gradient and minmax.

A further option is to use the C-value from the transition that was previously studied by clicking on the ‘Use previous value’ button.

Finally, there is a slider above the C-value box that allows the value to be continuously varied over a range of 101 values. The neighboring buttons < and > allow the user to step through values across this range.

6.2 Backup files

When starting up, WRITE_SCUPS_GUI checks if the SCUPS file already exists and then makes a copy of the file to `ion_name.scups_backup1`. As each transition is fitted the SCUPS file is updated. Immediately prior to the new SCUPS file being written, WRITE_SCUPS_GUI copies the current file to `ion_name.scups_backup2`. Therefore the user should not lose progress if the routine crashes at some point.

7 Sanity check

The routine COMPARE_UPS_SCUPS is available to compare how accurately the scaled upsilons reproduce the original upsilons. It is called as

```
IDL> compare_ups_scups, ion_name, missing_trans=missing_trans, prob_trans=prob_trans
```

The routine will print a message to indicate if there are transitions missing from the SCUPS file that are present in the UPS file, and the identities are stored in the structure MISSING_TRANS. A message is also printed to indicate if there are transitions for which the descaled upsilons are significantly different from the original upsilons. The identities of these transitions are stored in the PROB_TRANS structure. *If you have used WRITE_SCUPS_GUI to remove temperature points from a transition, then these transitions will be flagged!*

The keyword /AUTO is available so that the SCUPS_AUTO file (produced by WRITE_SCUPS) will be checked instead of the SCUPS file.

From a check of one ion it seems the accuracy of the scaled upsilons is about $\leq 0.5\%$, which stems from rounding errors in the printed scaled upilon values.

References

- Burgess, A., & Tully, J. A. 1992, A&A, 254, 436
Tayal, S. S., & Sossah, A. M. 2015, A&A, 574, A87

A Document history

Ver. 1.2, 17-Aug-2017. Updated Sect. 6.1.

Ver. 1.1, 4-Aug-2017. Added Sect. 6.1.

B Example output from WRITE_SCUPS

The text below shows sample output from the routine WRITE_SCUPS for an assessment of Mg V data from Tayal & Sossah (2015).

```
IDL> write_scups,'mg_5'
```

Transition type summary:

- type 1: 474
- type 2: 2682
- type 4: 499

The following problems were found. These transitions were NOT output to the scups file. PT refers to problem type. See Technical Report No. 4 for more details.

- PT 4 (high temp): 3
- PT 6 (allowed limit): 104

Please check the file mg_5_log.txt for more details.

To make plots of the problem transitions, use the PLOT_PROB_TYPE keyword.

The following files have been created:

- mg_5.scups_auto
- mg_5_flags.txt
- mg_5_log.txt