CHIANTI

An Astrophysical Database for Emission Line Spectroscopy

CHIANTI TECHNICAL REPORT No. 16

CHIANTI lookup tables for fast calculations

Version 1.5, 23 October 2024, Peter Young

1 Overview

The standard CHIANTI software calculates quantities such as emissivities and contribution functions in real-time directly from the atomic data. For large atomic models and a wide range of density and/or temperature parameters the calculation for a single ion may take several minutes. An alternative is to write out lookup tables that can be read and used multiple times without the need for new calculations. This is particularly useful for modeling codes used in astrophysics and solar physics that predict temperature and density in the plasma as a function of spatial position, typically with millions of pixels. Lookup tables are then the approprate way to derive the radiative emissions from these pixels in an efficient manner. Another application is when contribution functions are used in deriving differential emission curves for stellar atmosphere, where many iterations are performed over ranges of density and temperature (see also CHIANTI Technical Report No. 22). This document describes software within CHIANTI that enables lookup tables to be printed and then read back into IDL.

2 Methodology

The method chosen here is to write out, not the emissivities of each emission line from the ion, but the level populations. This is a much smaller data-set, and emissivities can be generated easily from the level populations without detailed calculation.

In order to improve the accuracy of the level populations derived from the lookup tables, the populations themselves are not written but instead scaled quantities derived from the populations. The formula for metastable levels is:

$$n_i' = 10^{5+\alpha_i/T} n_i \tag{1}$$

and for non-metastable levels it is:

$$n_i' = 10^{20+\alpha_i/T} \frac{n_i}{N_\mathrm{e}} \tag{2}$$

and the quantity printed to the tables is $\log n_i'$. Appendix B explains how metastable levels are identified.

These formulae are chosen because the populations of metastable levels (including the ground state) are approximately constant with density, while for non-metastables the population generally scales with N_e. The temperature dependence of the populations generally scales with $\exp(-E/kT)$, where E is the energy separation between the level and the principal excitation level. Usually the prinicipal excitation energy can be taken relative to the ground level, but for levels excited by recombination this is not the case. For this reason E is computed as follows. If the populations are computed over the temperature range T_0 to T_1, and the density range $N_{\mathrm{e},0}$ to $N_{\mathrm{e},1}$ then we consider the level population averaged over the density range, $\bar{n}(T)$, with $\bar{n}'$ given by Eqs. 1 or 2 with n replaced by $\bar{n}$. We then set $\bar{n}'(T_0) = \bar{n}'(T_1)$ and, following some algebra, we have

$$\alpha = \frac{T_0 T_1}{T_1 - T_0} (\log\, \bar{n}(T_1) - \log\, \bar{n}(T_0)) \tag{3}$$

The values of α are written to the lookup table.

The populations are computed from the lookup tables with `ch_lookup_table_interp`, which derives the interpolated values of the stored parameters and then converts them back to level populations.

3 Set-up: $CHIANTI_LOOKUP environment variable

[*Solarsoft users do not have to do this as $CHIANTI_LOOKUP is automatically set to point to the Solarsoft directory containing the pre-computed tables.*]

It is recommended that a user create the environment variable `$CHIANTI_LOOKUP` to point to where the lookup tables will be stored. If it is defined, then the lookup tables will automatically be written there (see Sect. 4) and the routine `ch_lookup_table_interp` will automatically look in this location when deriving the level populations (Sect. 5). Other routines such as `ch_lookup_gofnt` and `ch_lookup_emiss` (Sect. 7) also use the environment variable.

If you use `$CHIANTI_LOOKUP`, then it is recommended that you use a broad range of density parameters that covers all of the densities you anticipate encountering in your analyses.

When `$CHIANTI_LOOKUP` is defined, then you can write lookup tables for all of the CHIANTI ions into this directory using `ch_lookup_all_ions.pro`:

```
IDL> ch_lookup_all_ions
```

This calculation will take several hours and will generate files occupying approximately 550 Mbytes of disk space (for CHIANTI version 9 and with the default density range).

A set of pre-computed tables for the density range log $N_{\rm e} = 7.0$ to 13.0 is available at:

`http://chiantidatabase.org/chianti_download.html`

4 Writing the lookup tables

To write a lookup table, the IDL procedure is, for example,

```
IDL> ch_write_pop_lookup_table, 'c_3'
```

which writes a single table for C III to `$CHIANTI_LOOKUP` (use optional input `DIR_LOOKUP` to specify a different location). A default set of temperatures and densities are assumed, the output filename is `pop_lookup_c_3.txt`, and the populations of all levels are written.

Optional inputs exist to specify the temperature and density ranges, and to restrict the number of levels that are printed.

The temperatures are derived from the default CHIANTI ionization balance file: 0.05 dex intervals are used, and those temperatures for which the ion fraction is $\geq 10^{-8}$ of the maximum ion fraction are selected.

The default density range is log $N_{\rm e} = 8$ to 12, at 0.2 dex intervals.

5 Deriving level populations from the lookup tables

The recommended way of reading the lookup table is, for example,

```
IDL> output=ch_lookup_table_interp('c_3', dens, temp)
```

where `dens` and `temp` are the density and temperature ranges for which you want values, and the routine automatically looks in `$CHIANTI_LOOKUP` for the lookup table. The output is a structure with the following tags:

```
   POP             FLOAT     Array[401, 31, 337]
   LDENS           FLOAT     Array[401]
   LTEMP           FLOAT     Array[31]
   LEVELS          INT       Array[337]
   TEMP            FLOAT     Array[31]
   DENS            FLOAT     Array[401]
   CHIANTI_VERSION STRING    '8.0.6'
```

where `pop` is a 3D array (density $\times$ temperature $\times$ levels) containing the level populations. The array `levels` contains the CHIANTI level indices.

5.1 Reading the lookup table

The lookup table is read as follows:

```
IDL> output=ch_read_pop_lookup_table(filename)
```

which returns a structure with tags:

```
   POP             DOUBLE    Array[401, 31, 337]
   DENS            FLOAT     Array[401]
   TEMP            FLOAT     Array[31]
   META            BYTE      Array[337]
   LEVELS          INT       Array[337]
   LDENS           FLOAT     Array[401]
   LTEMP           FLOAT     Array[31]
   CHIANTI_VERSION STRING    '8.0.6'
```

These are directly the quantities stored in the lookup table, so the `pop` are the scaled level populations. Users are generally recommended to derive level populations with `ch_lookup_table_interp`.

6 Test of interpolation accuracy

When deriving populations from the lookup tables, `ch_lookup_table_interp` performs bilinear interpolation with the IDL routine `bilinear` in the log density—log temperature plane. This was preferred to cubic interpolation (using the routine `interpolate`) as the spline can sometimes give unnatural periodic behavior.

Tests on the accuracy of the interpolation procedure were performed by creating the default lookup tables for C III and Fe XVIII. Additional lookup tables were derived for both ions by specifying

a density array with steps of 0.01 dex. The default lookup tables were then interpolated onto this scale with `ch_lookup_table_interp` and populations compared with the fine scale data. In both cases the populations derived from the default lookup tables were within 0.65% of the correct populations.

Appendix C gives some comparisons of populations derived from the lookup tables with those from the regular CHIANTI routines.

7 Using lookup tables to reproduce the behavior of the regular CHIANTI routines

7.1 Emissivities (emiss_calc)

The routine `emiss_calc` is used to compute the emissivities for all lines of an ion. The emissivity for a transition $j \rightarrow i$ is defined as:

$$\varepsilon_{ij} = E_{ij} A_{ji} n_j \tag{4}$$

where E_{ij} is the transition energy, A_{ji} is the radiative decay rate, and n_j the level population.

To derive the emissivities from the lookup tables, the call is, e.g.,

```
IDL> em=ch_lookup_emiss('fe_13')
```

and the routine looks for the lookup table in the $CHIANTI_LOOKUP directory. The format of the output structure is identical to that from `emiss_calc`.

Note: a check for Fe XIII against `emiss_calc` with default temperatures and densities showed agreement to better than 1.5% for all transitions except those involving level 427, which were discrepant by 1.8%.

7.2 Contribution functions (gofnt)

The contribution function for an atomic transition $j \rightarrow i$ is defined as

$$G(T, N_{\rm e}) = \frac{\epsilon}{4\pi} \frac{hc}{\lambda} \frac{n_j(T, N_{\rm e})}{N_{\rm e}} F(T) A_{ji} \tag{5}$$

where ϵ is the element abundance (relative to hydrogen), hc/λ is the energy for the transition, T is the temperature, $F(T)$ is the ionization fraction of the emitting ion (a function of temperature only in CHIANTI), and A_{ji} is the radiative decay rate. In the regular CHIANTI IDL software this function is computed with the routine `gofnt.pro`—see the CHIANTI User Guide for more details. For complex ions such as the coronal iron ions `gofnt.pro` can be slow due to the need to calculate the n_j values for a range of temperatures. The use of lookup tables greatly speeds up the calculation, and the calculation is performed with the routine `ch_lookup_gofnt`. An example is:

```
IDL> g=ch_lookup_gofnt('o_6',wmin=1030,wmax=1035,log_dens=9.0)
```

where a widget will pop up allowing you to choose a line in the specified wavelength range. Alternatively the transition can be specified directly with

```
IDL> g=ch_lookup_gofnt('o_6',lower=1,upper=3,log_dens=9.0)
```

which will select the 1–3 transition at 1031.9 Å.

By default the routine uses the temperature range from the ioneq file over which the ionization fraction is non-zero. A different range can be specified with the `log_temp` keyword. Either a constant density or a constant pressure can be specified with the `log_dens` and `log_press` keywords.

There is also the option to input `log_dens` or `log_press` as arrays that have the same length as `log_temp`. In this case the contribution function is calculated for each temperature/density (or tempeature/pressure) pair. This can be useful if you have a plasma model with temperature and density/pressure as a function of time. The contribution function as a function of time can then be calculated with a single call to the routine.

For CHIANTI 11 (released in 2024), `ch_lookup_gofnt` now uses the advanced model ionization model by default. Thus the routine calls `ch_calc_ioneq` to compute the ion fraction using the specified density or pressure. To use the standard, zero-density ionization balance (the `chianti.ioneq` file), you must set `advanced_model=0` in the routine inputs.

Calculating the advanced model ion balance model file usually takes a few seconds. If you plan to use `ch_lookup_gofnt` multiple times, then it is recommended that you use `ch_calc_ioneq` to calculate the ion balance file, and then give this as an input to `ch_lookup_gofnt` through the `ioneq_file` input. In this case you must set `advanced_model=0` otherwise the routine will attempt to calculate a new ion balance.

The output is a structure with the following tags:

```
LTEMP           FLOAT     Array[23]
GOFNT           DOUBLE    Array[23]
LDENS           FLOAT           9.00000
LPRESS          FLOAT          -1.00000
ABUND_FILE      STRING    '/Users/pryoung/chianti_repository/dbase/trunk/
                               abundance/sun_coronal_1992_feldman_ext.abund'
IONEQ_FILE      STRING    '/Users/pryoung/chianti_repository/dbase/trunk/ioneq/
                               chianti.ioneq'
CHIANTI_VERSION STRING    '8.0.7'
TIME_STAMP      STRING    'Wed Apr 10 15:58:02 2019'
```

with `LTEMP` containing the log temperature array, and `GOFNT` containing the contribution function.

As an exercise the reader should compare the output with `gofnt`. For example:

```
IDL> g=ch_lookup_gofnt('fe_13',lower=1,upper=20,log_dens=9.0,/noabund)
IDL> gofnt,'fe_13',202,202.1,tt,gg,desc,lower=1,upper=20,dens=1e9,/noabund,
          logt0=g.ltemp,value=value)
IDL> p=plot(g.ltemp,g.gofnt,/xsty)
IDL> q=plot(/overplot,g.ltemp,value,symbol='+')
```

7.3 Density and temperature diagnostics (dens_plotter, temp_plotter)

The routines `dens_plotter` and `temp_plotter` are GUI-based routines for exploring density and temperature diagnostics from individual ions. For both routines the keyword /lookup has been implemented to enable the lookup tables to be used when calculating the ratios. This speeds up the routines significantly. Note that the density and temperature ranges are restricted to the ranges within the lookup tables, however.

```
IDL> temp_plotter, 'o_6', /lookup
IDL> dens_plotter, 'fe_13', /lookup
```

7.4 Synthetic spectra (ch_synthetic, ch_ss)

Synthetic spectra are created with the IDL routine `ch_synthetic` and the GUI-based routine `ch_ss`. For the former, the keyword /lookup has been added, which switches to using the lookup tables. Note that the output structure has a new tag called "lookup" which takes the value 0 or 1 to indicate if the keyword has been set. For ch_ss, the routine checks if the user has the `$CHIANTI\_LOOKUP` keyword defined. If yes, then a new widget is added to the GUI to allow the lookup tables to be used.

Appendix D compares the synthetic spectrum results when the lookup option is given.

8 Count rates from models

For a 3D simulation with a pixel of volume V we typically would like to know the photon count rate entering unit solid angle, and this is given by

$$P = \epsilon\, G'(T, N_{\rm e}) N_{\rm H} N_{\rm e} V \tag{6}$$

where G' is the modified contribution function given by `ch_lookup_gofnt` with the /photons and /noabund keywords set (it is often useful to deal with abundances separately in case you want to investigate, e.g., photospheric vs. coronal abundances).

Note that the hydrogen number density, $N_{\rm H}$, can be replaced by $(N_{\rm H}/N_{\rm e})N_{\rm e}$, where the hydrogen-to-electron ratio is obtained with the CHIANTI routine `proton_dens.pro`:

```
IDL> nh_ne=proton_dens(ltemp,/hydrogen,abund_file=abund_file)
```

The abundance file should be the same as that used for computing P.

9 The CHIANTI advanced models

CHIANTI 11 (Dufresne et al., 2024) was released in 2024 and features advanced models for ionization and recombination that have a significant effect on the ionization balance for transition region ions. The ion balance files can be created that are a function of density or pressure by using the routine `ch_calc_ioneq`.

The ion balance file enters into the level balance equations for an ion in two ways. One is through proton rate calculations, whereby the number density of protons has a dependence on the ion fractions. Secondly, some ions in CHIANTI include level-resolved ionization and recombination rates in their models, hence these processes depend on the ion fractions of the neighboring elements.

In both cases the effects of the new ion balance files on the level balance calculations are likely to be small or negligible. Hence it should be safe to use the population lookup tables that are calculated with the standard, zero-density ion balance file (`chianti.ioneq`).

If there are concerns that an advanced model ion balance file may impact the level populations, then a check can be performed with the `pop_solver` routine. For example, the code below computes two sets of level populations for O VI. The first uses the default zero-density ion balance file, and the second uses an ion balance file calculated for a density of 10^{12} cm^{-3}. By comparing the population arrays, `pop1` and `pop2`, one can check if the advanced model populations are significantly different (by $> 1\%$, say).

```
IDL> input1=ch_setup_ion('o_6')
IDL> pop_solver,input1,5e5,1e12,pop1
IDL> logt=findgen(81)/20.+4.
IDL> d=ch_calc_ioneq(10.^logt,dens=1e12,ele='O',outname='new.ioneq')
IDL> input2=ch_setup_ion('o_6',ioneq_file='new.ioneq')
IDL> pop_solver,input2,5e5,1e12,pop2
```

References

Dufresne, R. P., Del Zanna, G., Young, P. R., et al. 2024, ApJ, 974, 71, doi: http://doi.org/10.3847/1538-4357/ad676510.3847/1538-4357/ad6765

A Document history

Version 1.5, 23-Oct-2024. Some updates to Sect. 9.
Version 1.4, 26-Sep-2024. Sect. 9 has been added, and Sect. 7.2 has been updated.
Version 1.3, 27-Jan-2023. The document has been moved to Overleaf; eps format figures have been changed to png; Appendices C and D have been updated.
Version 1.2, 14-Jan-2021. Updated Sect. 3.
Version 1.1, 1-Jul-2020. Added Sect. 7.4 and Appendix D.
Version 1.0, 24-Mar-2020. First version of document

B Metastable levels

The software uses the CHIANTI routine `metastable_levels` to identify which levels are metastable. The criterion is very simple: if the maximum radiative decay rate (plus autoionization rate) from the level is $< 10^5$ s^{-1} then the level is considered to be metastable. Note that autoionization rates only exist for levels above the ionization limit.

The cutoff rate is set to identify metastable levels at typical coronal densities (10^8–10^{10} cm^{-3}). For very low densities (say, $\leq 10^4$ cm^{-3}) these levels may not be actually be metastable.

A better procedure would be to identify metastables based on the density range and the critical density of the levels (see the routine `ch_critical_density`). In reality the distinction between metastable and non-metastable does not have a significant effect on the accuracy of the table interpolation.

C Accuracy of output populations

Figure 1a shows the ratio of O VI populations derived with `ch_lookup_table_interp` to those derived with `ch_pops`. This example is chosen because of the complexity of the O VI model, which has many levels populated by recombination. The lookup tables are defined at temperature intervals of 0.05 dex so examples are shown at one of the node points (6.000) and precisely midway between two node points (6.025). The density of log $N_\mathrm{e} = 9.1$ is also chosen to be midway between two node points.

For log $T = 6.000$ the agreement is good to within 0.11%, whereas for log $T = 6.025$ the maximum difference is 1.02% and there is clearly a systematic discrepancy of about 0.6% for levels 100 and higher. These arise because of the bilinear interpolation that is applied. Most of the levels in O VI model are populated by recombination and the resulting populations show very similar temperature variation. Since bilinear interpolation always leads to small errors for data that are curved, the errors will be similar for the recombination lines. An accuracy of 1% is considered acceptable and we note that the important UV and EUV lines of O VI come from levels 1–8 for which the accuracy is much higher.

Figure 1b shows the simpler case of O III. The temperature of maximum ionization (T_max) is used for the comparison. The maximum difference at log $T = 4.900$ is 0.49% and at log $T = 4.925$ it is 0.38%. The density was set to log $N_\mathrm{e} = 9.1$.

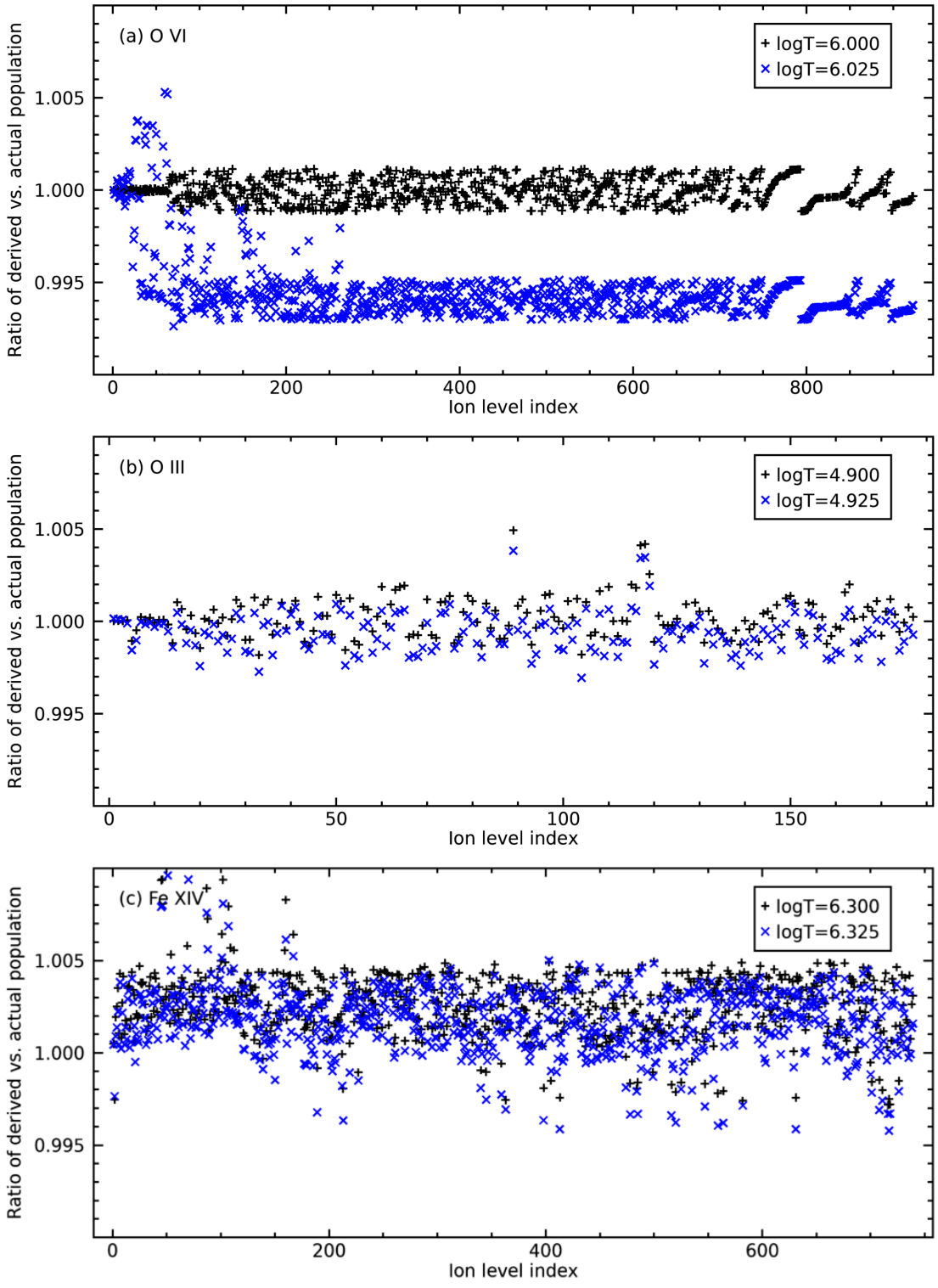

Figure 1: The ratio of lookup table populations to actual populations for three ions. The density is $10^{9.1}$ cm^{-3} in each case. Temperatures are indicated in the plots.

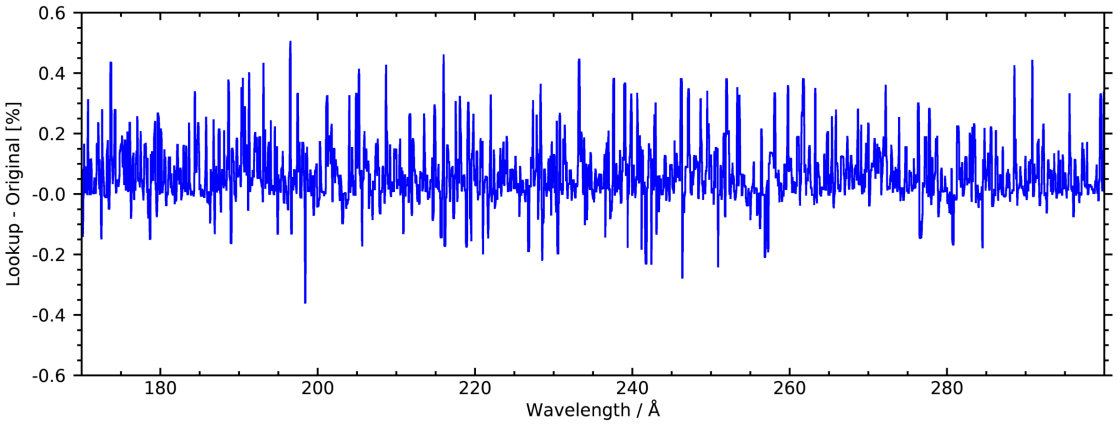

Figure 2: Synthetic spectra were computed for the wavelength range 170–300 Å. This plot shows the percentage difference between the lookup spectrum and the original spectrum.

Like O VI the coronal iron ions have 100's of levels in their models, but no recombination data. Figure 1c shows a comparison of level populations for Fe XIV. The maximum difference at $\log T = 6.3000$ is 1.09% and at $\log T = 6.325$ is 0.96%. The density was set to $\log N_e = 9.1$.

In summary these results show that the lookup populations are mostly accurate to 1% or better, which should be acceptable for most applications.

D Synthetic spectra calculations

Synthetic spectra were created with the `ch_synthetic` and `make_chianti_spec` routines assuming an electron pressure of 10^{15} K cm^{-3}, the CHIANTI active region DEM and the default photospheric abundances. The bin size was set to 22.3 mÅ and the instrumental FWHM to 62 mÅ (these are the parameters for the Hinode/EIS instrument). With the original software the calculation took 229 seconds on the author's computer, while with the lookup software it took 32 seconds (these numbers have been updated for CHIANTI 10.1). Note that the latter time is mostly taken up with reading data files.

Figure 2 shows the percentage difference between the two spectra. The maximum absolute difference is 0.54% and the median absolute difference is 0.057%. These results demonstrate that the lookup spectra should be of sufficient accuracy for most applications.