The plot shows $4\pi \nu J_\nu$ [erg cm^{-2} s^{-1}] versus Wavelength [Angstroms], labeled 10^8 K solar abundances.

Hazy

a brief introduction to CLOUDY *C13.04*

2. Results, computational environment, and test suite

Cloudy & Associates

Cover image: The spectrum emitted by a cloud with solar abundances, a temperature of 10^7 K, a density of $10^{10}\,\mathrm{cm}^{-3}$, and a column density of $10^{15}\,\mathrm{cm}^{-2}$. This shows recent improvements in the X-ray spectrum and was done with the test case coll_t7.in located in tsuite/auto. The command **set continuum resolution 0.1** was added to increase the continuum resolution by a factor of ten.

CONTENTS

LIST OF FIGURES

LIST OF TABLES

Chapter 1

OUTPUT

1.1 Overview

This section defines the output produced by CLOUDY. Each section begins with a sample of the output described, and then goes on to describe the meaning of the printout in greater detail. The output actually shown is from the Orion H II Region / PDR / molecular cloud test case (`orion_hii_pdr_pp.in`).

1.2 Header Information

Several lines of output echo the input commands and outline some properties of the initial continuum.

```
                                  Cloudy 06.01.02

****************************************06Jan02****************************************
*                                                                                   *
* title the Orion HII Region / PDR / Molecular cloud with an open geometry          *
* c                                                                                 *
* c commands controlling continuum =========                                       *
* c the incident continuum is two parts                                             *
* c kurucz continuum and flux of photons striking cloud                             *
* c this is the photosphere of the OVI star, its temperature and phi(H)             *
* table star kurucz 39600K                                                          *
* phi(H) 13                                                                         *
* c this adds the observed hot brems                                                *
* c its temperature (as log of T) and the flux of                                   *
* c photons striking the cloud                                                      *
* brems 6                                                                           *
* phi(h) 10                                                                         *
* c                                                                                 *
* c cosmic rays are important for pdr chemistry                                     *
* cosmic rays, background                                                           *
* c                                                                                 *
* c commands controlling geometry  =========                                       *
* c this turns off the stop temperature option                                      *
* c so the sim will not stop due to temperature                                     *
* stop temperature off                                                              *
* c this sets the thickness of the HII region & PDR                                 *
* stop thickness 0.5 linear parsec                                                  *
* c this will result in a milli gauss B-field in molecular region                   *
* magnetic field -5 gauss                                                           *
* c assume constant pressure                                                        *
* constant pressure                                                                 *
* set nend 2000                                                                     *
* c                                                                                 *
* c other commands for details     =========                                       *
* failures 3                                                                        *
* c mimic existence of unmodeled molecular gas                                      *
* double                                                                            *
* c iterate since lines optically thick                                             *
* iterate                                                                           *
* c set microturbulence in equipartition with B field                              *
```

```
* turbulence equipartition                                           *
* c set the line width so lines appear on the save continuum         *
* set SaveLwidth 10 km/s                                             *
* c                                                                 *
* c commands for density & abundances =========                     *
* c this is the log of the initial H density, cm-3                   *
* hden 4                                                            *
* c this will speed up the calculation a bit                        *
* init file="ism.ini"                                               *
* c this uses HII region abundances, but no grains                  *
* abundances hii region no grains                                   *
* c this uses orion grains                                          *
* grains orion                                                     *
* >>>> mie_read_opc reading file -- graphite_orion_10.opc      <<<< *
* >>>> mie_read_opc reading file -- silicate_orion_10.opc      <<<< *
* c turn on PAHs, with an abundance that depends on H0 fraction,     *
* c as suggested by long-slit observations of Orion bar,            *
* c with an abundance 3x larger than default built into the code     *
* grains pah function 3                                            *
* >>>> mie_read_opc reading file -- pah1_bt94_10.opc          <<<< *
* c                                                                 *
* c commands controlling output    =========                        *
* c print lots of faint CO lines                                    *
* print line faint -4                                               *
* c normalize to Ha                                                 *
* normalize to "H  1" 6563                                          *
* c                                                                 *
* c orion hii pdr pp.in                                             *
* c class hii pdr                                                   *
*                                                                  *
*********************************************************************
```

This begins with the version number of CLOUDY, the date that the version was released, in the form yy.mm.dd. The following line gives this date in another form.

All of the input command lines, with the exception of those starting with a #, %, or *, are echoed before the calculation begins, and are saved to be reprinted after the calculation is completed.

```
3198CellPeak1.00E+00   Lo 9.99e-09=912.21cm   Hi-Con:1.20E+02 Ryd   E(hi):7.35E+06Ryd   E(hi):    100.01 MeV
I(nu>1ryd):    2.4792   Average nu:1.382E+00   I( X-ray):  -1.6194   I(BalC):    2.8255   Phi(BalmrC): 13.7081
phi(1.0-1.8):12.9508    phi(1.8-4.0): 12.033   phi(4.0-20):  9.388   phi(20--):  7.634   Ion pht flx:1.001E+13
I(gam ray):   0.0000    phi(gam r):   0.0000   I(Infred):   1.5064   Alf(ox):   0.0000   Total inten:   3.0012
U(1.0----):3.339E-02    U(4.0----):8.293E-06   T(En-Den):4.586E+01   T(Comp):3.532E+04   nuJnu(912A):5.422E+02
Occ(FarIR):2.911E+08    Occ(H n=6):1.326E-11   Occ(1Ryd):2.470E-14   Occ(4R):5.400E-20   Occ (Nu-hi):1.201E-32
Tbr(FarIR):4.835E+05    Tbr(H n=6):5.816E-08   Tbr(1Ryd):3.905E-09   Tbr(4R):3.413E-14   Tbr (Nu-hi):2.269E-25
```

This block of information describes the continuum that strikes the illuminated face of the cloud. The full block of information is shown above, and in the following discussion each line is given again just before it is described.

```
3198CellPeak1.00E+00   Lo 9.99e-09=912.21cm   Hi-Con:1.20E+02 Ryd   E(hi):7.35E+06Ryd   E(hi):    100.01 MeV
```

This gives the number of numerical frequency cells in the continuum followed by the energy (in Ryd) of the peak of hydrogen-ionizing continuum. This is the point with the largest flux density per unit energy interval (J_ν). Next is the energy of the low-energy limit of the continuum, in both Ryd and cm. The last two numbers are the energies of the high-energy limit of the continuum in Ryd and MeV.

```
I(nu>1ryd):   2.4792   Average nu:1.382E+00   I( X-ray):  -1.6194   I(BalC):   2.8255    Phi(BalmrC): 13.7081
```

This line gives the intensity or luminosity of the continuum source. Luminosities are printed if the inner radius of the cloud is specified. The units will be energy radiated by the central object into 4π sr [erg s^{-1}]. If an inner radius is not set then the code will compute the intensity case and give the emission per unit area of cloud surface. This is loosely called the intensity but is more formally $4\pi J$ where J is the proper mean intensity [erg cm^{-2} s^{-1} sr^{-1} for an emission line; AGN3 Appendix 1].

The line gives the log of the energy (erg s^{-1} cm^{-2} or erg s^{-1}, depending on whether it is the intensity or luminosity case) in the hydrogen-ionizing continuum (1 Ryd $\leq h\nu <$ 100 MeV), and

the average energy of the hydrogen-ionizing continuum, in Ryd, weighted by photon number;

$$\langle h\nu \rangle = \frac{\int_{1\,Ryd}^{\infty} 4\pi J_\nu\, d\nu}{\int_{1\,Ryd}^{\infty} 4\pi J_\nu/h\nu\, d\nu} \quad [\text{Ryd}]. \tag{1.1}$$

The log of the energy in the X-ray continuum ($20.6\,\text{Ryd} \le h\nu \le 7676\,\text{Ryd}$) is followed by the log of the energy (erg s^{-1} cm^{-2} or erg s^{-1}) and the number of photons (cm^{-2} s^{-1} or s^{-1}) in the Balmer continuum (0.25 Ryd to 1.0 Ryd).

```
phi(1.0-1.8):12.9508   phi(1.8-4.0): 12.033   phi(4.0-20):  9.388   phi(20--):  7.634      Ion pht flx:1.001E+13
```

The third line gives the log of the number of photons (cm^{-2} s^{-1} or s^{-1}) in four frequency bins ($1.0\,\text{Ryd} \le h\nu < 1.807\,\text{Ryd}$, $1.807\,\text{Ryd} \le h\nu < 4.0\,\text{Ryd}$, $4.0\,\text{Ryd} \le h\nu < 20.6\,\text{Ryd}$, and $20.6\,\text{Ryd} \le h\nu < 7676\,\text{Ryd}$). The last number "Ion pht flx" is the flux of hydrogen ionizing photons;

$$\Phi\left(\text{H}^0\right) = \frac{Q\left(\text{H}^0\right)}{4\pi r^2} \quad [\text{cm}^{-2}\text{s}^{-1}]. \tag{1.2}$$

In this equation $Q(\text{H}^0)$ is the total number of hydrogen-ionizing photons emitted by the central object (s^{-1}), and r is the separation between the center of the central object and the illuminated face of the cloud. Unlike the majority of the quantities printed in the header, $\Phi(\text{H}^0)$ (per unit area) is always printed, never $Q(\text{H}^0)$ (into 4π sr).

```
I(gam ray):   0.0000   phi(gam r):   0.0000   I(Infred):   1.5064   Alf(ox):   0.0000     Total inten:   3.0012
```

The fourth line of the header gives some information about the low and high energy portions of the incident continuum. The first number is the log of the luminosity or intensity in the gamma-ray ($\sim$100 keV to $\sim$100 MeV) continuum. The second number is the log of the number of photons over this energy range. The third number is the log of the luminosity in the continuum between 0.25 Ryd and the lowest energy considered, presently an energy of 1.001×10^{-8} Ryd. All of these entries are either per unit area, or radiated into 4π sr, depending on whether the intensity or luminosity case was specified.

The entry "Alf(ox)" is the spectral index α_{ox}, defined as in Zamorani et al. (1981), except for the difference in sign convention. This is the spectral index which would describe the continuum between 2 keV (147 Ryd) and 2500Å (0.3645 Ryd) if the continuum could be described as a single power-law, that is,

$$\frac{f_\nu\,(2\,\text{keV})}{f_\nu\left(2500\,\text{Å}\right)} = \left(\frac{\nu_{2\,\text{keV}}}{\nu_{2500\,\text{Å}}}\right)^{\alpha} = 403.3^{\alpha}. \tag{1.3}$$

The definition of α_{ox} used here is slightly different from that of Zamorani et al. since implicit negative signs are *never* used by CLOUDY. Typical AGN have $\alpha_{ox} \sim -1.4$. If no X-rays are present then $\alpha_{ox} = 0$. The last number on the line is the log of the total energy in the continuum between 1.001×10^{-8} Ryd and 100 MeV.

```
log L/Lsun:   3.9743   Abs bol mg:  -5.1858   Abs V mag:   2.4170   Bol cor:  -7.6028     nuFnu(Bbet):  34.5867
```

This line is optional, depending on whether the luminosity or intensity case is specified. (It was not printed in this model since we are working in the intensity case but a sample is shown). This is printed in the luminosity case. First comes the log of the total luminosity in the continuum in solar units. The absolute bolometric magnitude, absolute V magnitude, and the bolometric correction, are then given, followed by the log of the continuum specific luminosity $[\nu F_\nu(\text{H}\beta)]$ at the wavelength of Hβ [erg s^{-1}].

```
U(1.0----):3.339E-02    U(4.0----):8.293E-06    T(En-Den):4.586E+01    T(Comp):3.532E+04    nuJnu(912A):5.422E+02
```

This line begins with two ionization parameters. The first is the dimensionless ratio of ionizing photon to hydrogen densities, defined as

$$U \equiv \frac{\Phi\left(H^0\right)}{n_H c}, \tag{1.4}$$

where n_H is the total hydrogen density. The second number is defined in a similar way, but the numerator is the number of photons with energies greater than 4 Ryd (i.e., helium-ionizing). The third number is the equivalent black-body temperature corresponding to the energy density u at the illuminated face of the cloud, from the incident continuum and Stefan's radiation density constant a;

$$T_u \equiv \left(L/4\pi r^2 ac\right)^{1/4} \quad [\text{K}]. \tag{1.5}$$

T(Comp) is the Compton temperature of the incident radiation field.[1] The last number is $4\pi\nu J_\nu(912Å)$, the flux at 912Å (erg cm^{-2} s^{-1}), where J_ν is the mean intensity of the incident continuum (Mihalas, 1978).

```
Occ(FarIR):2.911E+08    Occ(H n=6):1.326E-11    Occ(1Ryd):2.470E-14    Occ(4R):5.400E-20    Occ (Nu-hi):1.201E-32
Tbr(FarIR):4.835E+05    Tbr(H n=6):5.816E-08    Tbr(1Ryd):3.905E-09    Tbr(4R):3.413E-14    Tbr (Nu-hi):2.269E-25
```

These lines give dimensionless photon occupation numbers $\eta(\nu)$, for the incident continuum at several energies. The occupation number is defined as

$$\eta_\nu \equiv J_\nu(\nu)\left(\frac{2h\nu^3}{c^2}\right)^{-1}, \tag{1.6}$$

and the incident continuum brightness temperature $T_b(\nu)$, [K] is defined as

$$T_b(\nu) \equiv J_\nu(\nu)\left(\frac{2k\nu^2}{c^2}\right)^{-1} \quad [\text{K}]. \tag{1.7}$$

These energies correspond to the lowest frequency considered (presently 1.001×10^{-8} Ryd); the ionization potential of the $n = 6$ level of hydrogen (1/36 Ryd); one Rydberg; four Rydbergs, and the high-energy limit of the incident continuum. The energy where the last number is evaluated depends on the continuum shape. The energy is given by the fifth number on the first line of the continuum output.

1.3   Chemical composition

```
                              Gas Phase Chemical Composition
   H :  0.0000  He: -1.0223  C : -3.5229  N : -4.1549  O : -3.3979  Ne: -4.2218  Mg: -5.5229  Si: -5.3979  S : -5.0000
                              Cl: -7.0000  Ar: -5.5229  Fe: -5.5229

                              Grain Chemical Composition
              C : -3.6259  O : -3.9526  Mg: -4.5547  Si: -4.5547  Fe: -4.5547

                              Number of grains per hydrogen
                        Carbonaceous: -14.166   silicate: -14.103
```

---

[1]For a blackbody radiation field $T_{Compton}$ is roughly 4% lower than the blackbody color temperature T_{color} when the energy density temperature T_u is $> T_{color}$. Only when $T_u \equiv T_{color}$ does induced Compton heating cause $T_{Compton} \equiv T_{color}$. If $T_u > T_{color}$ then $T_{Compton} > T_{color}$ because of induced Compton heating. All of the relevant physics is included in the Compton temperature printed here.

The chemical composition of the cloud comes next. The three blocks of numbers give the gas-phase abundances of the elements, the abundances contained in grains, and the number of each type of grains per unit hydrogen. The numbers are the logs of the number densities of the elements, relative to the gas-phase hydrogen abundance of unity (so, 0 on the log scale). Only the active elements are included (those turned off with the **elements off** command are not printed). If grains are not present then the second two blocks are not printed.

1.4 Comments before or during the calculation

The code may print comments as the calculation proceeds. These are printed with the start of the comment in capital letters to make them easy to find with a script. The comments fall into three categories:

DISASTER something has unexpectedly caused the calculation to stop. The results are bogus and should not be trusted. The code should immediately return to the calling program.

PROBLEM something did not go as expected but the calculation is continuing. This might be a convergence failure at a point in the cloud. This is an indication that the code was having difficulties and a few of these can occur in a normal set of calculations. The predictions can be used if only a few occur and if the comments at the end of the calculation (see page 9 below) do not identify other problems.

NOTE this gives advice about the calculation. These can be ignored if you prefer.

1.5 Zone Results

Next comes a summary of the conditions in the first and last zone. This print out is done in routine *PrtZone* which should be consulted if there are any questions. The following is the output produced for one zone. Details follow.

```
#### 1 Te:9.361E+03 Hden:1.000E+04 Ne:1.101E+04 R:1.000E+30 R-R0:5.223E+11 dR:1.045E+12 NTR:  3 Htot:3.461E-16 T912: 1.48e-05###
Hydrogen      1.47e-04 1.00e+00 H+o/Hden 1.00e+00 4.12e-12 H-     H2 8.43e-17 4.10e-13 H2+ HeH+ 1.13e-12 Ho+ ColD 1.54e+12 1.04e+16
Helium        6.57e-04 9.45e-01 5.47e-02 He I2SP3 3.52e-06 Comp H,C 1.75e-22 4.64e-23 Fill Fac 1.00e+00 Gam1/tot 1.00e+00
He singlet n  6.54e-04 2.35e-11 6.59e-18 2.09e-18 3.18e-18 2.52e-18 He tripl 3.52e-06 9.32e-16 7.41e-18 4.77e-17 7.56e-18
Pressure      NgasTgas 2.06e+08 P(total) 2.84e-08 P( gas ) 2.84e-08 P(Radtn) 1.79e-11 Rad accl 3.51e-05 ForceMul 3.33e+03
              Texc(La) 4.21e+03 T(contn) 4.59e+01 T(diffs) 2.03e+00 nT (c+d) 1.16e+07 Prad/Gas 6.30e-04 Pmag/Gas 2.80e-04
gra-orion01*  DustTemp 2.12e+02 Pot Volt 5.32e+00 Chrg (e) 1.21e+02 drf cm/s 4.83e+03 Heating: 4.26e-18 Frac tot 1.23e-02
gra-orion02*  DustTemp 2.02e+02 Pot Volt 5.08e+00 Chrg (e) 1.43e+02 drf cm/s 5.26e+03 Heating: 3.69e-18 Frac tot 1.07e-02
gra-orion03*  DustTemp 1.91e+02 Pot Volt 4.86e+00 Chrg (e) 1.70e+02 drf cm/s 5.60e+03 Heating: 3.21e-18 Frac tot 9.26e-03
gra-orion04*  DustTemp 1.81e+02 Pot Volt 4.66e+00 Chrg (e) 2.01e+02 drf cm/s 5.86e+03 Heating: 2.79e-18 Frac tot 8.06e-03
gra-orion05   DustTemp 1.70e+02 Pot Volt 4.47e+00 Chrg (e) 2.39e+02 drf cm/s 6.07e+03 Heating: 2.43e-18 Frac tot 7.03e-03
gra-orion06   DustTemp 1.60e+02 Pot Volt 4.30e+00 Chrg (e) 2.85e+02 drf cm/s 6.22e+03 Heating: 2.13e-18 Frac tot 6.16e-03
gra-orion07   DustTemp 1.50e+02 Pot Volt 4.15e+00 Chrg (e) 3.40e+02 drf cm/s 6.33e+03 Heating: 1.87e-18 Frac tot 5.41e-03
gra-orion08   DustTemp 1.40e+02 Pot Volt 4.02e+00 Chrg (e) 4.07e+02 drf cm/s 6.42e+03 Heating: 1.65e-18 Frac tot 4.77e-03
gra-orion09   DustTemp 1.31e+02 Pot Volt 3.91e+00 Chrg (e) 4.89e+02 drf cm/s 6.48e+03 Heating: 1.46e-18 Frac tot 4.21e-03
gra-orion10   DustTemp 1.22e+02 Pot Volt 3.80e+00 Chrg (e) 5.89e+02 drf cm/s 6.54e+03 Heating: 1.29e-18 Frac tot 3.73e-03
sil-orion01*  DustTemp 1.55e+02 Pot Volt 2.85e+00 Chrg (e) 6.44e+01 drf cm/s 1.45e+04 Heating: 2.44e-18 Frac tot 7.06e-03
sil-orion02*  DustTemp 1.49e+02 Pot Volt 2.70e+00 Chrg (e) 7.58e+01 drf cm/s 1.62e+04 Heating: 2.09e-18 Frac tot 6.03e-03
sil-orion03*  DustTemp 1.43e+02 Pot Volt 2.57e+00 Chrg (e) 8.92e+01 drf cm/s 1.77e+04 Heating: 1.78e-18 Frac tot 5.15e-03
sil-orion04   DustTemp 1.37e+02 Pot Volt 2.44e+00 Chrg (e) 1.05e+02 drf cm/s 1.91e+04 Heating: 1.53e-18 Frac tot 4.42e-03
sil-orion05   DustTemp 1.31e+02 Pot Volt 2.33e+00 Chrg (e) 1.24e+02 drf cm/s 2.02e+04 Heating: 1.32e-18 Frac tot 3.81e-03
sil-orion06   DustTemp 1.26e+02 Pot Volt 2.24e+00 Chrg (e) 1.47e+02 drf cm/s 2.11e+04 Heating: 1.14e-18 Frac tot 3.30e-03
sil-orion07   DustTemp 1.20e+02 Pot Volt 2.15e+00 Chrg (e) 1.76e+02 drf cm/s 2.18e+04 Heating: 9.93e-19 Frac tot 2.87e-03
sil-orion08   DustTemp 1.15e+02 Pot Volt 2.08e+00 Chrg (e) 2.10e+02 drf cm/s 2.23e+04 Heating: 8.67e-19 Frac tot 2.50e-03
sil-orion09   DustTemp 1.10e+02 Pot Volt 2.01e+00 Chrg (e) 2.52e+02 drf cm/s 2.26e+04 Heating: 7.60e-19 Frac tot 2.20e-03
sil-orion10   DustTemp 1.06e+02 Pot Volt 1.96e+00 Chrg (e) 3.03e+02 drf cm/s 2.28e+04 Heating: 6.69e-19 Frac tot 1.93e-03
pah-bt9401 *  DustTemp 3.41e+02 Pot Volt 6.60e+00 Chrg (e) 1.06e+00 drf cm/s 1.97e+02 Heating: 9.29e-22 Frac tot 2.68e-06
pah-bt9402 *  DustTemp 3.44e+02 Pot Volt 6.64e+00 Chrg (e) 1.28e+00 drf cm/s 2.13e+02 Heating: 1.01e-21 Frac tot 2.92e-06
pah-bt9403 *  DustTemp 3.47e+02 Pot Volt 6.61e+00 Chrg (e) 1.49e+00 drf cm/s 2.37e+02 Heating: 1.07e-21 Frac tot 3.08e-06
pah-bt9404 *  DustTemp 3.50e+02 Pot Volt 6.54e+00 Chrg (e) 1.70e+00 drf cm/s 2.67e+02 Heating: 1.09e-21 Frac tot 3.16e-06
pah-bt9405 *  DustTemp 3.53e+02 Pot Volt 6.44e+00 Chrg (e) 1.92e+00 drf cm/s 3.02e+02 Heating: 1.09e-21 Frac tot 3.16e-06
```

```
pah-bt9406 *   DustTemp 3.55e+02 Pot Volt 6.45e+00 Chrg (e) 2.22e+00 drf cm/s 3.29e+02 Heating: 1.13e-21 Frac tot 3.26e-06
pah-bt9407 *   DustTemp 3.57e+02 Pot Volt 6.47e+00 Chrg (e) 2.54e+00 drf cm/s 3.60e+02 Heating: 1.17e-21 Frac tot 3.38e-06
pah-bt9408 *   DustTemp 3.59e+02 Pot Volt 6.43e+00 Chrg (e) 2.87e+00 drf cm/s 4.00e+02 Heating: 1.18e-21 Frac tot 3.40e-06
pah-bt9409 *   DustTemp 3.61e+02 Pot Volt 6.46e+00 Chrg (e) 3.27e+00 drf cm/s 4.35e+02 Heating: 1.20e-21 Frac tot 3.46e-06
pah-bt9410 *   DustTemp 3.63e+02 Pot Volt 6.48e+00 Chrg (e) 3.71e+00 drf cm/s 4.74e+02 Heating: 1.22e-21 Frac tot 3.52e-06
Carbon         6.76e-07 1.91e-02 9.70e-01 1.10e-02 1.77e-04 2.36e-06 0.00e+00 H2O+/O   0.00e+00 OH+/Otot 0.00e+00 Hex(tot) 0.00e+00
Nitrogen       1.60e-06 1.72e-02 9.69e-01 1.41e-02 4.77e-05 1.34e-07 9.81e-14 0.00e+00 O2/Ototl 0.00e+00 O2+/Otot 0.00e+00
Oxygen         9.23e-06 1.17e-01 8.59e-01 2.47e-02 6.97e-05 9.60e-08 4.83e-11 0.00e+00 0.00e+00
Neon           5.77e-05 3.93e-01 5.95e-01 1.17e-02 7.25e-05 5.08e-08 1.99e-12 1.47e-16 0.00e+00 0.00e+00
Magnesium      6.41e-06 7.31e-03 9.65e-01 2.74e-02 1.76e-04 5.41e-07 2.86e-10 3.72e-14 9.96e-19 0.00e+00 0.00e+00 0.00e+00 0.00e+00
Silicon      0 1.13e-07 8.39e-03 8.16e-01 1.56e-01 1.86e-02 4.68e-05 3.08e-08 5.13e-12 1.63e-16 0.00e+00 0.00e+00 0.00e+00 0.00e+00
Sulphur      0 7.25e-08 7.46e-03 9.03e-01 8.81e-02 1.60e-03 1.67e-04 5.91e-06 9.11e-10 3.49e-14 3.20e-19 0.00e+00 0.00e+00 0.00e+00
Chlorine     0 2.51e-07 1.32e-02 9.57e-01 2.92e-02 7.57e-04 5.50e-05 7.07e-07 9.54e-09 2.94e-13 2.17e-18 0.00e+00 0.00e+00 0.00e+00
Argon        0 9.50e-08 2.83e-03 9.69e-01 2.73e-02 4.35e-04 2.65e-05 3.24e-07 1.53e-09 1.16e-11 8.21e-17 0.00e+00 0.00e+00 0.00e+00
Iron         0 1.78e-08 2.88e-04 1.08e-01 8.56e-01 3.26e-02 2.33e-03 2.33e-05 2.30e-07 1.28e-10 1.94e-14 9.07e-19 0.00e+00 0.00e+00
```

The results of calculations for the first and last zones are always printed. Results for intermediate zones can be printed if desired (see the **print every** command). The following is a line-by-line description of the output produced for each printed zone.

```
#### 1  Te:9.361E+03 Hden:1.000E+04 Ne:1.101E+04 R:1.000E+30 R-R0:5.223E+11 dR:1.045E+12 NTR:  3 Htot:3.461E-16 T912: 1.48e-05###
```

The line begins with a series of #### characters to make it easy to locate with an editor. The zone number is the first number and it is followed by the electron temperature of the zone ("Te"). A lower case "u" will appear before the "Te" if the temperature solution is possibly thermally unstable. This occurs when the derivative of the net cooling with respect to temperature is negative. This is discussed further in the section on thermal stability problems starting on page below. The total hydrogen ("Hden") and electron ("Ne") densities (cm^{-3}) follow. The next number ("R") is the distance from the center of the central object to the center of the zone. The depth, the distance between the illuminated face of the cloud and the center of the zone, ("R-R0", or "r-r^o"), and the thickness of the zone ("dR", or δr), (all are in cm), follow. The inner edge of the zone is $(r - r_o) - \delta r/2$ from the illuminated face of the cloud. The line ends with a number indicating how many ionization iterations were needed for this zone to converge (NTR), followed by the total heating[2] ("Htot"; photoelectric and otherwise, erg cm^{-3} s^{-1}), and the optical depth between the *illuminated* face of the cloud and the *outer* edge of the zone at the Lyman limit (T912; the number is the *total absorption* optical depth at 912Å, and not the hydrogen Lyman-limit optical depth).

```
WIND; V:-7.000e+00km/s G:-0.00e+00 Accel: 8.62e-06 Fr(cont): 1.000 Fr(line): 0.000 Fr(dP): 0.000
```

A line describing the velocity and acceleration of the zone is printed if the cloud is a wind. The numbers are the wind velocity at the outer edge of the current zone (km s^{-1}), inward gravitational acceleration (cm s^{-2}), total outward radiative acceleration (cm s^{-2}), and the fraction of this acceleration caused by the incident continuum, line driving, and the gradient of the radiation pressure.

```
P(Lines):(Mg 2 2796A 0.32) (Mg 2  2803A 0.20) (H  1  6563A 0.13) (H 1 1.875m 0.10) (Fe 2  1786A 0.06) (H  1  4861A 0.05)
```

A line describing the source of the radiation pressure is generated if the ratio of line radiation to gas pressure, P_{rad}/P_{gas}, is greater than 5%. The line begins with the label "P(Lines)" and continues with the fraction of the total radiation pressure produced by that emission line, the spectroscopic designation of the line, and its wavelength. Up to twenty lines can be printed, although in most cases only Lα and a few others will dominate.

```
Hydrogen      1.47e-04 1.00e+00 H+o/Hden 1.00e+00 4.12e-12 H-    H2 8.43e-17 4.10e-13 H2+ HeH+ 1.13e-12 Ho+ ColD 1.54e+12 1.04e+16
```

---

[2]CLOUDY defines heating as the energy input by the freed photoelectron, or $h\nu - $IP, where IP is the ionization potential of the atom or ion, and $h\nu$ is the energy of the photon. See AGN3 for more details.

The line begins with the ratios $n(\mathrm{H}^0)/n(H_{tot})$ and $n(\mathrm{H}^+)/n(H_{tot})$ where H_{tot} is the total density in H all forms (including molecular). If **print h-like departure coefficients** has been specified then departure coefficients are also printed on the following line. Neutral hydrogen H^0 is defined to be the total population of atomic hydrogen in all explicitly computed bound levels. Next comes "H+o/Hden", the ratio $[n(\mathrm{H}^0) + n(\mathrm{H}^+)]/n(H_{tot})$.

The following five numbers give densities of the negative hydrogen ion and several molecules (H^-, H_2, H_2^+, and HeH^+) relative to the total hydrogen density. Note that, with this definition of the hydrogen density, a fully molecular gas will have $n(\mathrm{H}2)/n(\mathrm{H}) = 0.5$. These molecular abundances are also expressed as departure coefficients if the **print departure coefficients** command occurs. The last number is the H^0 and H^+ column densities (cm^{-2}).

```
H  1 1S-12    1.39e-01 3.43e-04 1.02e-03 1.02e-03 1.24e-03 1.62e-03 2.12e-03 2.73e-03 3.43e-03 4.23e-03 5.12e-03 6.12e-03 7.20e-03
H  1 rest     8.39e-03 9.66e-03 1.10e-02 1.25e-02 1.41e-02 1.57e-02 1.75e-02 1.93e-02 2.13e-02 2.33e-02 2.54e-02 2.77e-02 3.00e-02
              3.24e-02 3.49e-02 3.75e-02 4.02e-02 4.30e-02 4.59e-02 4.89e-02 5.20e-02 5.51e-02 5.84e-02 6.18e-02 6.52e-02 6.88e-02
              7.24e-02 7.62e-02 8.00e-02 8.39e-02 8.79e-02 9.21e-02 9.63e-02 1.01e-01 1.05e-01 1.09e-01 1.14e-01 1.19e-01
```

This information is only printed if the **print H-like populations** command occurs. The numbers give the populations of the H^0 levels relative to the ionized hydrogen density. All of these populations usually are relative to the ionized hydrogen density, but can also be printed as LTE departure coefficients if the **print departure coefficients** command is given.

```
Helium        6.57e-04 9.45e-01 5.47e-02 He I2SP3 3.52e-06 Comp H,C 1.75e-22 4.64e-23 Fill Fac 1.00e+00 Gam1/tot 1.00e+00
```

The first three numbers are the total populations of the three ionization stages of helium relative to the total helium abundance. The population of atomic helium is the sum of the total population in the triplets and singlets, including the population of all explicitly computed levels of each. These populations can also be expressed as departure coefficients if this option is set with the **print departure coefficients** command. The population of He 2^3 S, relative to the total helium abundance, follows. The Compton heating and cooling rates (both $\mathrm{erg\ cm}^{-3}\ \mathrm{s}^{-1}$) are next, followed by the gas filling factor. The last number is the fraction of the total hydrogen ionizations that are caused by photoionization from the ground state.

```
He singlet n  6.54e-04 2.35e-11 6.59e-18 2.09e-18 3.18e-18 2.52e-18 He tripl 3.52e-06 9.32e-16 7.41e-18 4.77e-17 7.56e-18
```

The first numbers are the level populations of the *l*-levels within $n = 1$ to 3 of the He^0 singlets. The next group consists of He^0 triplet populations of 2*S*, the three 2^3P_j, levels and the 3*S*, 3*P*, and 3*D* levels. All populations are relative to the total helium abundance. Departure coefficients are also printed if requested.

```
Pressure      NgasTgas 2.06e+08 P(total) 2.84e-08 P( gas ) 2.84e-08 P(Radtn) 1.79e-11 Rad accl 3.51e-05 ForceMul 3.33e+03
              Texc(La) 4.21e+03 T(contn) 4.59e+01 T(diffs) 2.03e+00 nT (c+d) 1.16e+07 Prad/Gas 6.30e-04 Pmag/Gas 2.80e-04
```

Some information concerning the pressure is printed. The gas equation of state includes thermal gas pressure, the radiation pressure due to trapped line emission, magnetic and turbulent pressure, and the radiation pressure due to absorption of the incident continuum. The first number is the gas pressure $n_{gas}T_{gas}$ (with units cm^{-3} K), followed by the total pressure (dynes cm^{-2}), and is followed by the gas pressure ($n_{gas}kT_{gas}$) in dynes cm^{-2}. The radiation pressure follows. The second to last number is the radiative acceleration (cm s^{-2}) at the inner edge of this zone. The radiative acceleration is computed with all continuous scattering and absorption opacities included. The last number is a force multiplier, defined as in Tarter and McKee (1973), and is the ratio of total opacity to electron scattering opacity.

The second line gives more information. The line starts with "Texc(La)", the excitation temperature T_{exc} of Lα, defined as

$$\frac{n(2p)/g(2p)}{n(1s)/g(1s)} = \exp\left[-h\nu/kT_{exc}(L\alpha)\right]. \tag{1.8}$$

This is followed by the temperature corresponding to the energy density of the attenuated incident continuum ("T(contn)") and the diffuse continua ("T(diffs)"). This includes all trapped lines and diffuse continuous emission. The entry "nT (c+d)" is the energy density of the sum of these two continua expressed as an equivalent pressure nT [cm^{-3} K]. The line ends with the ratios of the radiation to gas pressure "Prad/Gas" and the ratio of magnetic to gas pressure "Pmag/Gas".

```
gra-orion01*  DustTemp 2.12e+02 Pot Volt 5.32e+00 Chrg (e) 1.21e+02 drf cm/s 4.83e+03 Heating: 4.26e-18 Frac tot 1.23e-02
gra-orion02*  DustTemp 2.02e+02 Pot Volt 5.08e+00 Chrg (e) 1.43e+02 drf cm/s 5.26e+03 Heating: 3.69e-18 Frac tot 1.07e-02
gra-orion03*  DustTemp 1.91e+02 Pot Volt 4.86e+00 Chrg (e) 1.70e+02 drf cm/s 5.60e+03 Heating: 3.21e-18 Frac tot 9.26e-03
gra-orion04*  DustTemp 1.81e+02 Pot Volt 4.66e+00 Chrg (e) 2.01e+02 drf cm/s 5.86e+03 Heating: 2.79e-18 Frac tot 8.06e-03
gra-orion05   DustTemp 1.70e+02 Pot Volt 4.47e+00 Chrg (e) 2.39e+02 drf cm/s 6.07e+03 Heating: 2.43e-18 Frac tot 7.03e-03
gra-orion06   DustTemp 1.60e+02 Pot Volt 4.30e+00 Chrg (e) 2.85e+02 drf cm/s 6.22e+03 Heating: 2.13e-18 Frac tot 6.16e-03
gra-orion07   DustTemp 1.50e+02 Pot Volt 4.15e+00 Chrg (e) 3.40e+02 drf cm/s 6.33e+03 Heating: 1.87e-18 Frac tot 5.41e-03
gra-orion08   DustTemp 1.40e+02 Pot Volt 4.02e+00 Chrg (e) 4.07e+02 drf cm/s 6.42e+03 Heating: 1.65e-18 Frac tot 4.77e-03
gra-orion09   DustTemp 1.31e+02 Pot Volt 3.91e+00 Chrg (e) 4.89e+02 drf cm/s 6.48e+03 Heating: 1.46e-18 Frac tot 4.21e-03
gra-orion10   DustTemp 1.22e+02 Pot Volt 3.80e+00 Chrg (e) 5.89e+02 drf cm/s 6.54e+03 Heating: 1.29e-18 Frac tot 3.73e-03
sil-orion01*  DustTemp 1.55e+02 Pot Volt 2.85e+00 Chrg (e) 6.44e+01 drf cm/s 1.45e+04 Heating: 2.44e-18 Frac tot 7.06e-03
sil-orion02*  DustTemp 1.49e+02 Pot Volt 2.70e+00 Chrg (e) 7.58e+01 drf cm/s 1.62e+04 Heating: 2.09e-18 Frac tot 6.03e-03
sil-orion03*  DustTemp 1.43e+02 Pot Volt 2.57e+00 Chrg (e) 8.92e+01 drf cm/s 1.77e+04 Heating: 1.78e-18 Frac tot 5.15e-03
sil-orion04   DustTemp 1.37e+02 Pot Volt 2.44e+00 Chrg (e) 1.05e+02 drf cm/s 1.91e+04 Heating: 1.53e-18 Frac tot 4.42e-03
sil-orion05   DustTemp 1.31e+02 Pot Volt 2.33e+00 Chrg (e) 1.24e+02 drf cm/s 2.02e+04 Heating: 1.32e-18 Frac tot 3.81e-03
sil-orion06   DustTemp 1.26e+02 Pot Volt 2.24e+00 Chrg (e) 1.47e+02 drf cm/s 2.11e+04 Heating: 1.14e-18 Frac tot 3.30e-03
sil-orion07   DustTemp 1.20e+02 Pot Volt 2.15e+00 Chrg (e) 1.76e+02 drf cm/s 2.18e+04 Heating: 9.93e-19 Frac tot 2.87e-03
sil-orion08   DustTemp 1.15e+02 Pot Volt 2.08e+00 Chrg (e) 2.10e+02 drf cm/s 2.23e+04 Heating: 8.67e-19 Frac tot 2.50e-03
sil-orion09   DustTemp 1.10e+02 Pot Volt 2.01e+00 Chrg (e) 2.52e+02 drf cm/s 2.26e+04 Heating: 7.60e-19 Frac tot 2.20e-03
sil-orion10   DustTemp 1.06e+02 Pot Volt 1.96e+00 Chrg (e) 3.03e+02 drf cm/s 2.28e+04 Heating: 6.69e-19 Frac tot 1.93e-03
pah-bt9401 *  DustTemp 3.41e+02 Pot Volt 6.60e+00 Chrg (e) 1.06e+00 drf cm/s 1.97e+02 Heating: 9.29e-22 Frac tot 2.68e-06
pah-bt9402 *  DustTemp 3.44e+02 Pot Volt 6.64e+00 Chrg (e) 1.28e+00 drf cm/s 2.13e+02 Heating: 1.01e-21 Frac tot 2.92e-06
pah-bt9403 *  DustTemp 3.47e+02 Pot Volt 6.61e+00 Chrg (e) 1.49e+00 drf cm/s 2.37e+02 Heating: 1.07e-21 Frac tot 3.08e-06
pah-bt9404 *  DustTemp 3.50e+02 Pot Volt 6.54e+00 Chrg (e) 1.70e+00 drf cm/s 2.67e+02 Heating: 1.09e-21 Frac tot 3.16e-06
pah-bt9405 *  DustTemp 3.53e+02 Pot Volt 6.44e+00 Chrg (e) 1.92e+00 drf cm/s 3.02e+02 Heating: 1.09e-21 Frac tot 3.16e-06
pah-bt9406 *  DustTemp 3.55e+02 Pot Volt 6.45e+00 Chrg (e) 2.22e+00 drf cm/s 3.29e+02 Heating: 1.13e-21 Frac tot 3.26e-06
pah-bt9407 *  DustTemp 3.57e+02 Pot Volt 6.47e+00 Chrg (e) 2.54e+00 drf cm/s 3.60e+02 Heating: 1.17e-21 Frac tot 3.38e-06
pah-bt9408 *  DustTemp 3.59e+02 Pot Volt 6.43e+00 Chrg (e) 2.87e+00 drf cm/s 4.00e+02 Heating: 1.18e-21 Frac tot 3.40e-06
pah-bt9409 *  DustTemp 3.61e+02 Pot Volt 6.46e+00 Chrg (e) 3.27e+00 drf cm/s 4.35e+02 Heating: 1.20e-21 Frac tot 3.46e-06
pah-bt9410 *  DustTemp 3.63e+02 Pot Volt 6.48e+00 Chrg (e) 3.71e+00 drf cm/s 4.74e+02 Heating: 1.22e-21 Frac tot 3.52e-06
```

Some properties of the grain populations are printed if they are present. Each line gives the results of calculations for a specific type and size of grain. Graphite and silicate are normally included when grains are present. Each line begins with the name of the grain and an asterisk appears if quantum heating was important for the species. Quantum heating is only computed if it is significant due to its computational expense. The remainder of the line gives the equilibrium temperature of the grain, the potential in volts, the charge, the drift velocity, the gas heating (erg cm^{-3} s^{-1}) due to grain electron photoemission, and the dimensionless fraction of the total gas heating due to grain electron photoemission. For quantum-heated grains the temperature is the average weighted by T^4.

```
Molecules     CH/Ctot: 3.65e-04 CH+/Ctot 5.82e-13 CO/Ctot: 5.52e-01 CO+/Ctot 1.95e-14 H2O/Otot 3.27e-09 OH/Ototl 4.18e-11
```

A line giving relative abundances of some molecules is printed if the molecular fraction is significant. All molecular abundances are relative to either the total carbon or total oxygen abundance (this is indicated in the label for each). In order, the molecules are CH, CH$^+$, CO, CO$^+$, H$_2$O, and OH.

```
Lithium       8.94e-02 9.11e-01 9.53e-10 0.00e+00 Berylliu 9.99e-01 6.40e-04 6.38e-05 7.06e-10 0.00e+00 sec ion: 7.66e-15

Abundances of each stage of ionization of lithium and beryllium relative
to the total gas-phase abundance of the element are followed by the
suprathermal secondary ionization rate [s$^{-1}$].
{\setverbatimfontsize{\tiny}
\begin{verbatim}
Carbon        6.76e-07 1.91e-02 9.70e-01 1.10e-02 1.77e-04 2.36e-09 0.00e+00 H2O+/O   0.00e+00 OH+/Otot 0.00e+00 Hex(tot) 0.00e+00
```

The abundances of the seven stages of ionization of carbon relative to the total gas-phase carbon abundance begin the line. The abundance of H_2O^+ and OH^+ relative to the total gas-phase oxygen abundance are given. These are followed by "Hex(tot)", the extra heat (erg cm^{-3} s^{-1}) due to fast neutrons, dissipation of turbulence, or added with the **hextra** command.

```
Nitrogen      1.60e-06 1.72e-02 9.69e-01 1.41e-02 4.77e-05 1.34e-07 9.81e-14 0.00e+00 O2/Ototl 0.00e+00 O2+/Otot 0.00e+00
```

The relative populations of the eight ionization stages of nitrogen are printed first. The abundance of O_2 and O_2^+, relative to the total oxygen abundance, follows.

```
Silicon    0  1.13e-07 8.39e-03 8.16e-01 1.56e-01 1.86e-02 4.68e-05 3.08e-08 5.13e-12 1.63e-16 0.00e+00 0.00e+00 0.00e+00
Sulphur    0  7.25e-08 7.46e-03 9.03e-01 8.81e-02 1.60e-03 1.67e-04 5.91e-06 9.11e-10 3.49e-14 3.20e-19 0.00e+00 0.00e+00 0.00e+00
Chlorine   0  2.51e-07 1.32e-02 9.57e-01 2.92e-02 7.57e-04 5.50e-05 7.07e-07 9.54e-09 2.94e-13 2.17e-18 0.00e+00 0.00e+00 0.00e+00
Argon      1  9.50e-08 2.83e-03 9.69e-01 2.73e-02 4.35e-04 2.65e-05 3.24e-07 1.53e-09 1.16e-11 8.21e-17 0.00e+00 0.00e+00 0.00e+00
Iron       3  1.78e-08 2.88e-04 1.08e-01 8.56e-01 3.26e-02 2.33e-03 2.33e-05 2.30e-07 1.28e-10 1.94e-14 9.07e-19 0.00e+00 0.00e+00
```

There are too many ionization stages to print across the line for elements more massive than neon. Although all stages with non-trivial abundances are computed, only the highest twelve stages of ionization are printed. The first number is an integer indicating how many stages are "off the page to the left". If the number is 2, then the first printed stage of ionization is twice ionized, i.e., Fe^{+2}.

1.6 Comments after the calculation

```
    the Orion HII Region / PDR / Molecular cloud with an open geometry
    Calculation stopped because outer radius reached. Iteration 1 of 2
```

A series of messages appear after the printout of the last zone. The first will say why the calculation stopped. In a valid calculation the model will stop because one of the specified stopping criteria specified was met. If no other criteria are specified then the calculation usually stops when the default lowest temperature of 4000 K is reached. If the code stops because of an unintended reason (i.e., internal errors, or reaching the default limit to the number of zones) then a warning is printed saying that the calculation may have halted prematurely.

Only one stopping criterion message will be printed. The possible messages, and their interpretations, are:

. . . because of radiation pressure By default a cloud will have constant density. CLOUDY will keep the total pressure, particle and radiation, constant if constant pressure is specified with the **constant pressure** command. The radiation pressure is small at the boundaries of the cloud, so the cloud will be unstable if the ratio of radiation to total pressure exceeds 0.5. The calculation stops, and this message is generated, if $P_{rad}/P_{tot} > 0.5$ occurs after the first iteration.

. . . because lowest EDEN reached The calculation can be forced to stop when the electron density (**eden**) falls below the value set by the **stop eden** command. This can be used to stop the calculation at an ionization front. The default lowest electron density is negative, so this stopping criterion occurs only when the command is entered.

. . . because low electron fraction The calculation can be forced to stop when the ratio of electron to hydrogen densities falls below a certain value, as set by the **stop efrac** command. This can be used to stop the calculation at an ionization front when the hydrogen density there is not known (for instance, in a constant pressure model). The default lowest electron density is negative, so this stopping criterion applies only when the command is entered.

. . . because large H_2**/H fraction** The calculation can be forced to stop when the ratio of densities of molecular hydrogen to total hydrogen rises above the value set by the **stop mfrac** command. The molecular fraction is defined as $2n(H_2)/n(H_{tot})$. This can be used to stop the calculation at some depth into a PDR. The default highest molecular density is $\gg 1$, so this stopping criterion occurs only when the **stop mfrac** command is entered.

. . . because wind veloc too small The code can perform a wind calculation which includes the outward force due to radiation pressure and the inward force of gravity. This message is printed if the gas decelerates to a stop.

. . . because code returned BUSTED The calculation stopped because something bad happened. Please post the input script and version number on the discussion board.

. . . because DRAD small - set DRMIN The Strömgren radius of the H^+ zone is estimated at the start of the calculation and the smallest allowed zone thickness is set to a very small fraction of this. The calculation will stop if the zone thickness falls below this smallest thickness. This can occur because of any of several logical errors within CLOUDY (adaptive logic is used to continuously adjust the zone thickness), although it can rarely occur for physical reasons as well. The smallest thickness can be reset to any number with the set **drmin** command but it should not be necessary to do this. Please post the input script and version number on the discussion board.

. . . because DR small rel to thick The depth into the cloud is stored as the double precision variable *depth* and the zone thickness is stored as the double precision variable *drad*. If the zone size becomes too small relative to the depth (*drad/depth* $< 10^{-14}$) then the depth variable will underflow such that *depth + drad = depth*. The calculation will stop if this problem prevents the density from being properly evaluated.

. . . because optical depth reached The largest allowed continuous absorption optical depth can be set with the **stop optical depth** command. The command specifies both the absorption optical depth, and the energy at which it is to be evaluated. Scattering opacities are not included since their effects are very geometry dependent. If the calculation stops because the largest continuum optical depth is reached, then this line is printed. This line is also printed if the **stop effective column density** command is used to stop the calculation, since this command is actually a form of the **stop optical depth** command.

. . . because outer radius reached The default outer radius is unphysically large, but can be changed with the **radius** or **stop thickness** commands. If the calculation stops because the outer radius set by one of these commands is reached, then this line is printed.

because column dens reached a limit to the largest allowed neutral, ionized, and total hydrogen column densities is set with the **stop column density**, **stop neutral column density**, or **stop ionized column density** commands. This message will be printed if one of these criteria stops the calculation.

. . . because lowest Te reached The default value of the lowest temperature allowed is 4000 K. This is reasonable when only emission from warm ionized gas is of interest. The limit can

be changed with the **stop temperature** command. This message is printed if the calculation stops because the lowest temperature is reached.

...because highest Te reached The default value of the highest temperature allowed is 1.001×10^{10} K. The limit can be changed with the **stop temperature exceeds** command. This message is printed if the calculation stops because the highest allowed temperature is exceeded.

...because freeze out fraction Nick Abel incorporated the condensation of molecules onto grain surfaces. Currently CO, H_2O, and OH condensation are treated. The chemistry network will become unstable when oxygen is highly depleted from the gas phase. By default the code stops when 99% of the oxygen abundance has condensed out of the gas phase.

...because NZONE reached By default the code will stop after computing 1400 zones. This can be reset with the **stop zone** command. This message is printed if the calculation stops because the limiting number of zones is reached. A warning is printed at the end of the calculation since this was probably not intended.

The default limit to the number of zones can be increased, while retaining the check that the default limit is not hit, by using the **set nend** command.

...because volume too large for this cpu This indicates that the effective volume of the last zone was too large to be represented as a single precision floating point number. This happens when R/R_{in} becomes really large. Usually this indicates that none of the intended stopping criteria were hit, so you should check those. If you really intend for the simulation to integrate this far, you should increase the inner radius.

...because line ratio reached It is possible to set a limit to the largest value of an emission-line intensity ratio with the **stop line** command. This message is printed if the calculation stops because the largest value of the ratio is reached.

...because internal error - DRAD An internal logical error caused this message to be printed. Please report the problem, including the command lines and the version number of CLOUDY, on the discussion board at the code's web site, .

...because initial conditions out of bounds The temperature of the first zone was not within the temperature bounds of the code. This is probably due to the incident continuum not being set properly.

...because zero electron density The electron density fell to zero because there was no source of ionization. This is unphysical and usually occurs because the cloud boundary conditions were not set properly. Consider adding at least galactic background cosmic rays with the **cosmic ray background** command and perhaps the galactic or extragalactic background.

...because reason not specified This is an internal error. Please post the input and version number on the code's discussion board.

1.7   Geometry

```
The geometry is plane-parallel.
```

The code will next say whether the geometry is plane parallel ($\Delta r/r_\mathrm{o} < 0.1$), a thick shell ($\Delta r/r_\mathrm{o} < 3$), or spherical ($\Delta r/r_\mathrm{o} \geq 3$), where r_o is the inner radius and Δr is the thickness of the cloud.

1.8   Warnings, Cautions, Surprises, and Notes

```
 C-Cloud thicker than smallest Jeans length=3.51e+16cm; stability problems? (smallest Jeans mass=2.58e-01Mo)
  !Magnetic field & cosmic rays both present.  Their interations are not treated.
  !Some input lines contained underscores, these were changed to spaces.
  !Suprathermal collisional ionization of H reached 83.84% of the local H ionization rate.
  !H2 vib deexec heating reached 6.68% of the local heating.
  !Charge transfer ionization of H reached 95.8% of the local H ionization rate.
  !The largest continuum brightness temperature was 4.835e+05K at 1.052e-08 Ryd.
  !Both constant pressure and turbulence makes no physical sense???
  !AGE: Cloud age was not set.  Longest timescale was 8.43e+15 s = 2.67e+08 years.
  !The excitation temp of Lya exceeded the electron temp, largest value was  4.60e+03K (gas temp there was  1.01e+03K, zone 310)
  !Line absorption heating reached 13.45% of the local heating - largest by level 1 line Si 2 34.8m
  !Some infrared fine structure lines are optically thick:  largest tau was  2.05e+03
  !Local grain-gas photoelectric heating rate reached  98.8% of the total.
  !The local H2 photodissociation heating rate reached 12.8% of the total heating.
  !The CMB was not included.  This is added with the CMB command.
  !The fraction of H  in H2    reached 100.0% at some point in the cloud.
  !The fraction of C  in CO    reached 100.0% at some point in the cloud.
  !The fraction of N  in N2    reached 99.7% at some point in the cloud.
  !The fraction of S  in CS    reached 100.0% at some point in the cloud.
  !The fraction of S  in SO    reached 56.9% at some point in the cloud.
  !The fraction of S  in OCS   reached 36.3% at some point in the cloud.
  !The fraction of Cl in HCl   reached 24.5% at some point in the cloud.
  !The fraction of O  in H2Ogrn reached 58.3% at some point in the cloud.
  !A signficant amount of molecules condensed onto grain surfaces.
  !These are the molecular species with "grn" above.
  !The optical depth in the H I 21 cm line is 6.84e-01
  !The optical depth in the 12CO J=1-0 line is 2.79e+05
  !The radiation pressure jumped by 123% at zone 334, from 5.28e-11 to 3.67e-11 to 3.73e-11
  !The H2 density varied by 12.6% between two zones
   Continuum fluorescent production of H-beta was significant.
   Te-ne bounds of Case B lookup table exceeded, H I Case B line intensities set to zero.
   Te-ne bounds of Case B lookup table exceeded, He II Case B line intensities set to zero.
   Destruction of He 2TriS reached 3.4% of the total He0 dest rate at zone 236, 3.4% of that was photoionization.
   Critical density for l-mixing of He I not reached.  More resolved levels are needed for accurate He I line ratios.
   The largest continuum occupation number was 2.911e+08 at 1.052e-08 Ryd.
   The continuum occupation number fell below 1 at 1.944e+04 microns.
   The continuum brightness temperature fell below 10000K at 1.433e+06 microns.
   Ratio of computed diffuse emission to case B reached 2.30168 for iso 1 element 2
   Global grain photoelectric heating of gas was  9.9% of the total.
   Local grain-gas cooling of gas rate reached  90.4% of the total.
   The local H2 cooling rate reached 7.2% of the local cooling.
   Local CO rotation cooling reached 83.0% of the local cooling.
   The Balmer continuum optical depth was  3.68e+03.
   The Balmer continuum stimulated emission correction to optical depths reached  7.64e-02.
   The Paschen continuum optical depth was 1.69e+03.
   The continuum optical depth at the lowest energy considered (1.052e-08 Ryd) was  3.213e+03.
   The optical depth to Rayleigh scattering at 1300A is 4.36e-02
   3 body rec coef outside range 1
   The fraction of Cl in CCl+   reached 0.103% at some point in the cloud.
   The fraction of N  in HNC    reached 0.312% at some point in the cloud.
   The fraction of C  in C2H    reached 2.60% at some point in the cloud.
   The fraction of C  in COgrn  reached 0.379% at some point in the cloud.
```

The next messages fall into four categories: warnings beginning with "W-"; cautions beginning with "C-"; surprising results beginning with an explanation mark ("!"), and notes.

CLOUDY does many internal sanity checks to confirm that its range of validity was not exceeded (Ferland, 2001). Warnings are issued to indicate that the program has not treated an important process correctly. For instance, warnings occur if the temperature was high enough for the electrons to be relativistic, if the global heating - cooling balance is off by more than 20%, or if the code stopped for an unintended reason. We would like to hear about warnings - the web site has a discussion board to place comments. Cautions are less severe, and indicate that CLOUDY is on thin ice. Examples are when the optical depths in excited states of hydrogen change during the

last iteration. Surprises indicate that, while the physical process has been treated correctly, the result is surprising. An example is when induced Compton heating is more that 5 percent of the total Compton heating. Notes indicate interesting features about the model, such as maser effects in lines or continua, or if the fine structure lines are optically thick. The messages are usually self-explanatory.

1.9 Optional Plot

```
l-                         Log(nu fnu) os log(nu) "NEW" PARIS MEETING PLANETARY NEBULA
l       o
l-           o         o       o ooo   o                    ..............
l        oooo  o    oo  ooo oooo  ooo    ............          .........
l-   oo       ooooooooooooooooooooooooooo....                      .....
loooooooooooooooooo       ........                             ...
oo           .........                            ...
l             ........                              ...
l-   ........                         o             ..
.......                                oo          ooooooo ..
l-                                     oo          ooo    oooo.
l                          o          oo         oo       ooo.
l-                                   ooo          oo         oo.
l                                     o            o         oo.
l-                                   oo           oo          oo.
l                   oo   o   o       oo           oo           o.
l-              ooooooooo       o                 o             o.
l          oo     o    ooooo            oo                     oo.
l-             ooooooo          o                              oo
l               oo                   oo                        oo
l-                                    o                        oo
l                                     o                        o.
l-                                    o                        oo
l                                     o                        oo
l-                                   ooo                       o.
l                                     o                         o
l-                        oo   o                               oo
l                         o    o                                o
l-                                  o                          oo
l                        oo   o                                 o
l-                       o    o                                oo
l                             o                                 o
l-                        o    o                               oo
l                             o                                 o
l-                            o                                oo
l                            oo                                 o
l-                            o                                o.
l                                                              o
l-                            o                                o
l                            o                                oo
l-                           o                                 o
l                           o                                 o.
l-                       o  o                                  o
l                           o                                 o
l-                          o                                 o.
l                                                             o
l-                       oo                                   o
l                       oo                                   oo
l-                                                           o
l                                                           o
l-                                                          o.
l                                                           o
l-                                                          o
l                                                          oo
l                                                           o
l-              l                         l                 o
-----------------------------------------------------------------------------------------o-
    0.1                     1                          10
```

If any of the optional plots are requested with a **plot - -** command then they will appear next. This option is seldom used today since it is much easier to create data files with save commands and then use other software to make plots.

1.10 Final Printout

```
******************************> Cloudy 06.01.02 <******************************
```

```
                    * title the Orion HII Region / PDR / Molecular cloud with an open geometry    *
                    * c                                                                            *
                    * c commands controlling continuum =========                                  *
           -
           -
           -
           -
                    * assert line luminosity "12CO" 235.4m  -2.80 error 0.15                       *
                    * assert line luminosity "12CO" 215.7m  -2.84 error 0.15                       *
                    * assert nzone < 1400                                                          *
                    * assert itrzn < 24                                                            *
                    * c orion hii pdr pp.in                                                        *
                    * c class hii pdr                                                              *
                    * c =====================================                                      *
                    ********************************> Log(U): -1.48 <********************************
                    >>>>>>>>>> Cautions are present.
```

The final printout begins by reprinting the input commands. The box surrounding it gives both the version number of CLOUDY (at the top) and the log of the ionization parameter (the ratio of ionizing photon to hydrogen densities) at the bottom.

```
        Emission Line Spectrum.  Constant  Pressure Model.    Open geometry.  Iteration 2 of 2.
                                  Intensity (erg/s/cm^2)
```

This line summarizes some properties of the model and output. The first part indicates whether the energy in the emission lines is given as the luminosity case (the energy radiated by a spherical shell covering Ω sr [erg s^{-1}] where $\Omega/4\pi$ is the covering factor) or the intensity case (emission produced by a unit area of gas [erg s^{-1} cm^{-2}])). Which of the two choices is printed is determined by whether the luminosity of the continuum was specified as the luminosity radiated by the central object into 4π sr or the intensity $(4\pi J)$ of the incident continuum (erg cm^{-2} s^{-1}) at the illuminated face of the cloud. If the cloud is spherical and the intensity case is computed then the emergent emission-line spectrum will be per unit area in units of the inner radius ro (that is, the total line luminosity radiated by a shell covering 4π sr will be the listed intensity $4\pi J \times 4\pi r_\text{o}^2$). The second part of this line indicates the density structure (i.e., wind, constant density, constant pressure, constant gas pressure, power-law density distribution, etc). The next section tells whether the geometry was open or closed (these are defined in Part I of this document). The last part indicates the iteration number.

```
                                          Intrinsic line intensities
general properties.........    H  1 1.500m  -2.042   0.0010      +Col 1.278m  -1.666   0.0024      O  2 833.8A  -1.615   0.0027
TOTL  4861A   0.473   0.3311    H  1 7.458m  -1.094   0.0090      Ca B 1.870m  -1.760   0.0019      O 2r  4651A  -2.135   0.0008
TOTL  1216A   0.740   0.6119    H  1 4.653m  -1.299   0.0056      +Col 1.870m  -1.757   0.0019      O 2r  4341A  -2.413   0.0004
Inci     0    3.001 111.7791    H  1 3.740m  -1.487   0.0036      Ca B 1.279m  -2.153   0.0008      TOTL  4341A  -2.413   0.0004
TotH     0    1.727   5.9400    H  1 3.296m  -1.650   0.0025      +Col 1.279m  -2.149   0.0008      TOTL  1665A  -1.363   0.0048
-
-
-
H  1 1.513m  -2.287   0.0006    Ca B 1.869m  -1.283   0.0058      O 2r  2471A  -2.363   0.0005      hyperfine structure.........
H  1 1.508m  -1.964   0.0012    +Col 1.869m  -1.275   0.0059      O 2r  7323A  -2.536   0.0003      inner shell................
H  1 1.504m  -1.999   0.0011    Ca B 1.278m  -1.676   0.0024      O 2r  7332A  -2.657   0.0002
                                          Emergent line intensities
TOTL  4861A   0.132   1.0000    12CO 517.8m  -3.749   0.0001      12CO 431.5m  -3.474   0.0002      12CO 369.8m  -3.268   0.0004
13CO 353.6m  -3.709   0.0001    13CO 309.4m  -3.605   0.0002      13CO 275.0m  -3.547   0.0002      13CO 247.5m  -3.544   0.0002
13CO 225.0m  -3.641   0.0002    H  1  1216A   0.547   2.6028          H  1  1026A  -1.178   0.0490      H  1 972.5A  -1.350   0.0330
-
-
-
C  3 398.4A  -3.773   0.0001    N  1 980.7A  -3.567   0.0002      N  2 916.3A  -2.513   0.0023      N  2 646.4A  -2.307   0.0036
N  3 348.7A  -3.763   0.0001    O  1 7950A  -2.869   0.0010       O  2 4188A  -3.756   0.0001      O  2 386.3A  -1.774   0.0124
O  2 385.7A  -1.955   0.0082    O  3 300.5A  -3.653   0.0002
```

A series of predicted quantities follow. These are mainly emission-line intensities although the output also includes other predicted quantities.

Some continua and various indications of contributors to lines and continua are included. The Chapter of this document describing observed quantities (starting on page 25) tells how to convert these into some observed quantities. Not all are printed by default. The discussions on the **print** commands in Part I tell how to get more or fewer predictions. This list of emission lines can also

be sorted by wavelength or intensity, and can be printed as a single column so that they can be entered into a spreadsheet.

The organization and meaning of the different of lines in the printout is discussed in the Chapter *The Emission Lines* starting on page .

A list of emission lines with negative relative intensities may follow the main block of lines. These are lines which heat rather than cool the gas (heating is negative cooling). This is not a problem but occurs if the line's collisional de-excitation rate exceeds its collisional excitation rate. This usually occurs when the line is radiatively excited but collisionally de-excited.

1.10.1 Intrinsic line intensities and luminosities

There are two blocks of predicted intensities. The block "Intrinsic line intensities" is the intrinsic emission from the cloud. The intrinsic emission includes all processes that affect the line formation and transfer. This includes collisional processes, fluorescence, line destruction by background opacities such as dust or the Lyman continuum of hydrogen, and recombination. The intrinsic intensities do not include the effects of absorbers or scatters that do not lie within the line-formation region.

1.10.2 Emergent line intensities and luminosities

The block "Emergent line intensities" is the emission observed from outside the cloud. In an open geometry the inward part of the line includes the effects of extinction between the line-forming region and the illuminated face. There is an additional contribution due to reflection off the gas in the outward direction. The outward part of the line includes the effects of extinction between the shielded face and the point where the line forms. In a closed geometry the emergent intensity is the emission escaping to the outer edge of the slab. All of this is very geometry dependent. For an open geometry, in particular, the observed emission will depend strongly upon the viewing angle.

If the intrinsic and emergent values are very different then it is important to understand the geometry and what is actually observed. This is an indication of the uncertainty if the predictions have a strong geometry dependence.

1.10.3 Some physical properties of the cloud

```
the Orion HII Region / PDR / Molecular cloud with an open geometry
Cooling: HFBc    0 :0.110 HFFc    0 :0.077 TOTL  3727A:0.074 O  3  5007A:0.210 O  3  4959A:0.070 S  3  9532A:0.074
Heating: BFH1    0 :0.817 BFHe    0 :0.074 GrGH    0 :0.099
```

Cooling: This line indicates the fraction of the total cooling (defined here as the energy of the freed photoelectron, see AGN3 Chapter 3) carried by the indicated emission lines. The line label is followed by the ratio of the energy in the line to the total cooling. This is an important indication of the fundamental power-losses governing conditions in the model. The labels used are the same as those in the line array.

Heating: This line indicates the fraction of the total heating produced by various processes. The format is the same as the line giving the cooling.

```
IONIZE PARMET:  U(1-) -1.4764  U(4-): -5.0813  U(sp): -4.46  Q(ion):  -4.574  L(ion)-13.505   Q(low): 15.158    L(low)  1.121
```

The line begins with the log of the H "U(1-)" and He^+ "U(4_)" ionization parameters. The third number "U(sp)" is the log of a spherical ionization parameter often used in spherical geometries, such as H II regions or planetary nebulae. It is defined as

$$U_{sph} = \frac{Q\left(H^0\right)}{4\pi R_s^2 n_H c} \tag{1.9}$$

where R_s is the Strömgren radius, defined as the point where the hydrogen neutral fraction falls to $H^0/H = 0.5$. If no ionization front is present then U_{sph} is evaluated at the outer edge of the cloud. The next two numbers are the log of the number of hydrogen-ionizing photons ($h\nu \geq 1$ Ryd) exiting the nebula "Q(ion)" and the log of the energy in this continuum "L(ion)". The next two numbers are the equivalent quantities for non-ionizing ($h\nu < 1$ Ryd) radiation. These are either per unit area or by a shell covering 4π sr. These have been corrected for the r^{-2} dilution if per unit area, and so are directly comparable with the numbers given at the start of the calculation.

```
ENERGY BUDGET: Heat:  1.725  Coolg:  1.725  Error:  0.0%  Rec Lin:  1.462  F-F  H -5.612   P(rad/tot)mx:1.10E-01
```

This line gives an indication of the energy budget of the nebula. The first number "Heat" is the log of the total heating (in ergs s^{-1}, but again either into 4π sr or cm^{-2}). The second number "Coolg" is the log of the total cooling, in the same units. Cooling is the total energy in collisionally excited lines and part of the recombination energy, but *does not* include recombination lines (AGN3 Chapter 3). The percentage error in the heating-cooling - cooling match "Error" follows. The next numbers give "Rec Lin", the log of the total luminosity in recombination lines, "F-F H", the log of the amount of energy deposited by free-free heating, and "P(rad/tot)mx", the largest value of the ratio of radiation to gas pressures that occurred.

```
   Col(Heff):     1.031E+25  snd travl time  5.06E+13 sec  Te-low: 1.69E+01  Te-hi: 1.08E+04 G0TH85:3.54E+05   G0DB96:4.54E+05
```

The effective column density "Col(Heff)", as defined in the section in Part 1 on the **stop effective column density** command, is printed. This is followed by "snd travl time", the sound travel time across the nebula in seconds. Constant pressure is only valid if the cloud is static for times considerably longer than this. The last two numbers are the lowest "Te-low" and highest "Te-hi" electron kinetic temperatures found in the computed structure. The last numbers "G0TH85" and "GHBD96" give the intensity of the ultraviolet radiation field relative to the background Habing value, as defined by Tielens and Hollenbach (1985) and Bertoldi and Draine (1996).

```
   Emiss Measure   n(e)n(p) dl     2.205E+25  n(e)n(He+)dl      1.962E+24  En(e)n(He++) dl    1.402E+21
```

This gives several line-of-sight emission measures. The definition of the line of sight emission measure of a species X is (AGN3 section 5.4)

$$E\left(X\right) = \int n\left(e\right) n\left(X\right) f\left(r\right) dr \quad [cm^{-5}] \tag{1.10}$$

where $f\left(r\right)$ is the filling factor. This is given for H^+, He^+, and He^{2+}.

```
He/Ha:9.61E-02  =  1.01*true  Lthin:1.81E+01  itr/zn: 9.78  H2 itr/zn: 0.00  File Opacity: F MassTot 1.12e32  (gm)
```

This line includes some quantities deduced from the predicted emission-line spectrum. The first ("He/Ha") number is the apparent helium abundance He/H, measured from the emission-line intensities using techniques described in AGN3 (Chapter 5);

$$\left(\frac{\text{He}}{\text{H}}\right)_{apparent} = \frac{0.739 \times I(5876) + 0.078 \times I(4686)}{I(\text{H}\beta)}. \tag{1.11}$$

The intensities of all lines are the total predicted intensities and include contributions from collisional excitation and radiative transfer effects. The second number (i.e., "1.07*true") is the ratio of this deduced abundance to the true abundance. This provides a simple way to check whether ionization correction factors, or other effects, would upset the measurement of the helium abundance of the model nebula. This is followed by the longest wavelength in centimeters "Lthin" at which the nebula is optically thin. Generally the largest FIR opacity source is bremsstrahlung and the number will be 10^{30} if the nebula is optically thin across the full continuum. The number "itr/zn" is the average number of iterations needed to converge each zone while "H2 itr/zn" is the number of iterations per zone required to converge the large H2 model if it is included. "MassTot" gives the total mass of the computed structure in grams if the inner radius was specified. If the inner radius was not specified, and the cloud is plane parallel, then the mass per unit area [gm cm^{-2}] is reported.

```
Temps(21 cm)   T(21cm/Ly a)  8.40E+02      T(<nH/Tkin>)  1.04E+03      T(<nH/Tspin>)  1.29E+03      TB21cm     5.32E+02
               N(H0/Tspin)   5.22E+18      N(OH/Tkin)    2.01E+13
```

This line gives various quantities related to the H I 21 cm line. "T(21cm/Ly a)" gives the temperature deduced from the ratio of the 21 cm to Lα line optical depths (AGN3 Section 5.5). The opacity within the 21 cm line is proportional to $n\left(\text{H}^0\right)\chi/kT$ where χ is the excitation energy of the line. "T(<nH/Tkin>)" gives the harmonic mean temperature

$$\langle T \rangle = \frac{\int T n\left(\text{H}^0\right)\chi/kT\ dr}{\int n\left(\text{H}^0\right)\chi/kT\ dr} \tag{1.12}$$

where T is the electron or gas kinetic temperature. "T(nH/Tspin)" is the temperature derived from the n/T_{spin} ratio using the 21 cm spin temperature. T_{spin} is calculated from the ratio of populations of the ground fine structure levels, which is computed including the effects of Lα scattering. The spin and kinetic temperatures are often assumed to be equal although they are not in practice. The number "TB21cm" is an estimate of the brightness temperature of the 21 cm line as viewed from the illuminated face of the cloud. It is the spin temperature at a depth where the 21 cm line becomes optically thick at line center.

On the next line the next two numbers are the H^0 and OH column densities divided by the 21 cm spin temperature in the case of H^0 and by the kinetic temperature in the case of OH. These ratios are proportional to the optical depth of a line at radio frequencies.

```
<a>:0.00E+00  erdeFe0.0E+00  Tcompt1.90E+06  Tthr1.19E+13  <Tden>: 2.38E+01  <dens>:2.00E-17 <Mol>:2.31E+00
```

The mean radiative acceleration "<a>" [cm s^{-2}] is printed if the geometry is a wind model and zero otherwise. This is followed by some time scales. "erdeFe" is the time scale, in seconds, to photoerode Fe (Boyd and Ferland, 1987). This is 0s if the γ-ray flux is zero. The next gives the Compton equilibrium timescale "Tcompt" and the thermal cooling timescale "Tthr" [s]. The density (gm cm^{-3}) weighted mean temperature "<Tden>", radius-weighted mean density "<dens>" (gm cm^{-3}), and mean molecular weight "<Mol>", follow.

```
   Mean Jeans  l(cm)4.19E+16  M(sun)4.36E-01  smallest:     len(cm):3.53E+16  M(sun):2.61E-01 Alf(ox-tran):     0.0000
```

This gives the mean Jeans length "l(cm)" (cm) and Jeans mass "M(sun)" (in solar units). This is followed by the smallest Jeans length "smallest len(cm)" and the smallest Jeans mass "M(sun)" which occurred in the calculation. The last quantity "Alf(ox-tran)" is the spectral index α_{ox}, defined as in the header, but for the transmitted continuum (attenuated incident continuum plus emitted continuum produced by the cloud).

```
Rion:1.001E+17  Dist:6.17E+21  Diam:6.689E+00
```

This gives the radius of the hydrogen ionization front (in cm), or the outer radius in case the hydrogen ionization front is never reached. This is followed by the distance (in cm), or zero in case the **distance** command was not used. Finally the angular diameter of the ionized region is given (in arcsec), or zero if the distance was not set. The angular diameter is defined as `2*Rion/Dist`.

```
Hatom level 26  NHtopoff:  22  HatomType: add  HInducImp  F  He tot level: 63  He2 level:   26 ExecTime 2233.72
```

This line gives the number of levels of the model hydrogen atom, the "topoff" level, above which the remainder of the recombination coefficient is added, the type of top off used for this calculation, and the number of levels used for the atomic helium. The last number on the line is the execution wall-clock time in seconds.

```
ConvrgError(%)  <eden>  0.075  MaxEden  0.543  <H-C>   0.20  Max(H-C)   0.50  <Press>   0.042 MaxPres  2.535
 Continuity(%)  chng Te   3.9  elec den   5.8  n(H2)  12.8  n(CO)        8.5
```

The first line gives some estimates of the errors that occurred in several quantities that the code converges. A pair of numbers gives the mean and largest percentage errors for the electron density, the heating-cooling balance, and the pressure. The second line gives the percentage changes that occurred from one zone to the next for the temperature, the electron density, and the H_2 and CO densities.

1.10.4   Average temperatures and densities

```
                                              Averaged Quantities
           Te       Te(Ne)    Te(NeNp)  Te(NeHe+)Te(NeHe2+) Te(NeO+)  Te(NeO2+)  Te(H2)     N(H)      Ne(O2+)    Ne(Np)
Radius:  6.86e+02  9.19e+03  9.27e+03  9.18e+03  9.09e+03  9.65e+03  8.86e+03  2.28e+01  8.30e+06  1.43e+04  1.50e+04
```

This begins with several temperature and density averages, over either radius or volume. The volume averages are only printed if the **sphere** command is entered. The quantity which is printed is indicated at the top of each column. The quantity being averaged is the first part of the label, and the weighting used is indicated by the quantity in parenthesis. For instance "Te(NeO2+)" is the electron temperature averaged with respect to the product of the electron and O^{2+} densities.

```
Peimbert T(OIIIr)9.08E+03 T(Bac)0.00E+00 T(Hth)0.00E+00 t2(Hstrc)  6.01e-03 T(O3-BAC)0.00E+00 t2(O3-BC) 0.00e+00 t2(O3str) 1.81e-03
 Be careful: grains exist.  This spectrum was not corrected for reddening before analysis.
```

This series of quantities deal with temperature fluctuations (t^2, Peimbert, 1967; AGN3 section 5.11) . The code analyzes the predicted emission line and continuum spectrum using the same steps that Manuel outlined in this paper. The code does not attempt to correct the predicted emission-line intensities for collisional suppression or reddening, so this line is only printed if the density is below the density set with the **set tsqden** command—the default is 10^7 cm^{-3}. This code does not attempt to deredden the spectrum: a caution is printed if grains are present.

The nature of temperature fluctuations is, in my option, the biggest open question in nebular astrophysics. Theory (CLOUDY too) predicts that they should be very small because of the steep dependence of the cooling function on the temperature, while some observations indicate a very large value of t^2 (see Liu et al., 1995, Kingdon and Ferland, 1995, and Ferland, 2003). If something is missing from our current understanding of the energy source of photoionized nebulae then the entire nebular abundance scale (for both the Milky Way and the extragalactic nebulae) is in error by as much as 0.5 dex.

Two fundamentally different t^2s enter here—the "structural" t^2 and the "observational" t^2. The structural value comes from the computed ionization and thermal structure of the nebula while the observational value comes from an analysis of the predicted emission-line spectrum following the methods outlined in Peimbert's 1967 paper.

The structural t^2 for the $\mathrm{H^+}$ ion is defined as

$$t^2\left(\mathrm{H^+}\right) = \left\langle \left[\frac{T\left(r\right) - \langle T \rangle}{\langle T \rangle} \right]^2 \right\rangle = \frac{\int \left[T\left(r\right) - \langle T \rangle \right]^2 n_e n\left(\mathrm{H^+}\right) f\left(r\right) dV}{\langle T \rangle^2 \int n_e n\left(\mathrm{H^+}\right) f\left(r\right) dV} \tag{1.13}$$

where $< T >$ is the density-volume weighted mean temperature

$$\langle T \rangle = \frac{\int T\left(r\right) n_e n\left(\mathrm{H^+}\right) f\left(r\right) dV}{\int n_e n\left(\mathrm{H^+}\right) f\left(r\right) dV}. \tag{1.14}$$

This quantity is given in the averaged quantities block as the column "Te(NeNp)."

The observational t^2—related quantities are the following: "T(OIIIr)" is the electron temperature indicated by the predicted [OIII] 5007/4363 ratio in the low-density limit. This number is meaningless for densities near or above the critical density of the [O III] lines. "T(Bac)" is the hydrogen temperature resulting from the predicted Balmer jump and Hβ. "T(Hth)" is the same but for optically thin Balmer continuum and case B Hβ emission. "t2(Hstrc)" is the structural H II t^2. The entries "T(O3-BAC)" and t2(O3-BC)" are the mean temperature and t^2 resulting from the standard analysis of the [O III] and H I spectra (Peimbert, 1967). Finally "t2(O3str)" is the structural t^2 over the $\mathrm{O^{2+}}$ zone. Only the structural t^2s are meaningful for high densities. This section was developed in association with Jim Kingdon, and Kingdon and Ferland (1995) provide more details.

1.10.5 Average grain properties

```
Average Grain Properties (over radius):
        gra-orion01* gra-orion02* gra-orion03* gra-orion04* gra-orion05* gra-orion06* gra-orion07  gra-orion08  gra-orion09  gra-orion10
   nd:      0            1            2            3            4            5            6            7            8            9
<Tgr>: 3.659e+01    3.592e+01    3.522e+01    3.450e+01    3.377e+01    3.306e+01    3.236e+01    3.170e+01    3.106e+01    3.045e+01
<Vel>: 4.177e+02    4.780e+02    5.483e+02    6.279e+02    7.223e+02    8.327e+02    9.572e+02    1.095e+03    1.241e+03    1.396e+03
<Pot>: 3.072e-01    2.905e-01    2.736e-01    2.575e-01    2.423e-01    2.281e-01    2.151e-01    2.036e-01    1.932e-01    1.842e-01
<D/G>: 1.202e-04    1.337e-04    1.486e-04    1.653e-04    1.837e-04    2.043e-04    2.271e-04    2.525e-04    2.808e-04    3.122e-04

        sil-orion01* sil-orion02* sil-orion03* sil-orion04* sil-orion05  sil-orion06  sil-orion07  sil-orion08  sil-orion09  sil-orion10
   nd:     10           11           12           13           14           15           16           17           18           19
<Tgr>: 3.303e+01    3.261e+01    3.216e+01    3.173e+01    3.130e+01    3.090e+01    3.051e+01    3.015e+01    2.981e+01    2.950e+01
<Vel>: 1.093e+03    1.237e+03    1.385e+03    1.532e+03    1.676e+03    1.817e+03    1.953e+03    2.086e+03    2.215e+03    2.342e+03
<Pot>: 1.693e-01    1.598e-01    1.509e-01    1.427e-01    1.352e-01    1.283e-01    1.220e-01    1.163e-01    1.109e-01    1.062e-01
<D/G>: 2.031e-04    2.259e-04    2.511e-04    2.792e-04    3.104e-04    3.451e-04    3.837e-04    4.267e-04    4.744e-04    5.274e-04

        pah-bt9401 * pah-bt9402 * pah-bt9403 * pah-bt9404 * pah-bt9405 * pah-bt9406 * pah-bt9407 * pah-bt9408 * pah-bt9409 * pah-bt9410 *
   nd:     20           21           22           23           24           25           26           27           28           29
<Tgr>: 4.620e+01    4.634e+01    4.650e+01    4.662e+01    4.675e+01    4.683e+01    4.691e+01    4.697e+01    4.702e+01    4.706e+01
<Vel>: 9.565e+00    1.100e+01    1.255e+01    1.426e+01    1.617e+01    1.819e+01    2.029e+01    2.261e+01    2.516e+01    2.779e+01
<Pot>: 2.697e+00    2.424e+00    2.181e+00    1.964e+00    1.768e+00    1.594e+00    1.442e+00    1.307e+00    1.188e+00    1.084e+00
<D/G>: 9.119e-08    9.556e-08    1.002e-07    1.050e-07    1.100e-07    1.153e-07    1.208e-07    1.266e-07    1.327e-07    1.391e-07
Dust to gas ratio (by mass): 5.457e-03, A(V)/N(H)(pnt):5.543e-22, (ext):4.024e-22, R:3.647e+00 AV(ext):5.152e+03 (pnt):7.097e+03
```

The next lines give some information concerning grains if these were included in the calculation. These lines give the mean temperature, drift velocity, and potential, for all of the grain populations included in the calculation. An asterisk will appear to the right of the name of any species with quantum heating included. In this case the mean temperature is weighted by T^4.

The last line gives some information related to the grain abundance and optical properties. The first number is the dust to gas ratio by mass. The next two are the total visual extinction per unit hydrogen column density for a point and extended source. These are different because of the different effects of forward scattering (AGN3 section 7.6). Next comes the ratio of total to selective extinction. The line ends with A_V for both an extended and point source.

1.10.6   Optical depths

```
Contin Optical Depths: COMP: 1.07e-03     H-: 1.90e-04     R(1300): 4.53e-02  H2+: 1.12e-06  Bra:1.55E+02
                       Pa: 7.89e+02     Ba: 3.98e+03      Hb: 5.31e+03    La: 1.07e+04    1r:1.151E+05  1.8:3.79E+07 4.:4.382E+06
 Line Optical Depths: 10830:  8.12e+02  3889: 3.49e+01   5876: 1.91e-04 7065: 2.62e-05  2.06m: 1.29e-02  21c: 5.29e-01
```

The first two lines give the continuum optical depths at various energies. These are the total optical depths, including the correction for stimulated emission, and will be negative if maser action occurs. All opacity sources are included. The labels, and their interpretation, are as follows. COMP is the Thomson scattering optical depth. "H-" is the optical depth at the wavelength where the negative hydrogen ion has its greatest maximum cross section. "R(1300)" is the optical depth due to Rayleigh scattering by H^0 at 1300Å. "H2$^+$" is the optical depth at the dissociation threshold of the molecular hydrogen ion. "Bra" is the optical depth at the wavelength of the Brackett α transition (5-4).

The next line gives total continuous optical depths at the energies of various hydrogen and helium ionization edges and mean line optical depths[3]. These are evaluated at the energies of the Paschen α, Balmer α and β, and Lα lines, and the ionization edges of hydrogen, atomic helium, and the helium ion.

The third line gives optical depths of some He I lines. These are computed with a full model of the He^0 atom (Porter et al., 2005).

```
Old, new H  1 continuum optical depths:
      1 1.13e+05      2 3.68e+03      3 1.69e+03      4 8.24e+02      5 4.03e+02      6 2.12e+02      7 1.24e+02      8 8.47e+01
      9 7.60e+01     10 3.02e+02     11 2.24e+02     12 1.05e+02     13 9.80e+01     14 1.31e+02     15 1.16e+02     16 9.21e+01
     17 7.61e+01     18 6.54e+01     19 5.68e+01     20 4.88e+01     21 4.19e+01     22 3.55e+01     23 3.00e+01     24 2.55e+01
     25 2.17e+01
      1 1.13e+05      2 3.61e+03      3 1.66e+03      4 8.08e+02      5 3.95e+02      6 2.08e+02      7 1.22e+02      8 8.31e+01
      9 7.46e+01     10 2.96e+02     11 2.20e+02     12 1.03e+02     13 9.61e+01     14 1.28e+02     15 1.14e+02     16 9.03e+01
     17 7.46e+01     18 6.41e+01     19 5.57e+01     20 4.78e+01     21 4.11e+01     22 3.48e+01     23 2.95e+01     24 2.50e+01
     25 2.12e+01

Old, new H  1 line optical depths:
  2- 1 2.59e+09  3- 2 1.58e-01  4- 3 9.14e-08  5- 4 5.27e-08  6- 5-1.66e-08  7- 6-1.39e-07  8- 7-5.70e-07
  9- 8-1.24e-06 10- 9-1.79e-06 11-10-3.40e-06 12-11-4.47e-06 13-12-4.36e-06 14-13-3.91e-06 15-14 9.70e-06 16-15-6.04e-05
 17-16-1.08e-04 18-17-1.88e-04 19-18-3.44e-04 20-19-6.90e-04 21-20 1.37e-02 22-21 2.07e-02 23-22 6.52e-02 24-23 8.91e-02
 25-24 1.18e-01
  2- 1 1.16e+09  3- 2 7.08e-02  4- 3 6.02e-08  5- 4 2.67e-08  6- 5-2.15e-08  7- 6-1.51e-07  8- 7-5.76e-07
  9- 8-1.23e-06 10- 9-1.78e-06 11-10-3.35e-06 12-11-4.41e-06 13-12-4.30e-06 14-13-3.81e-06 15-14 4.83e-06 16-15-5.97e-05
 17-16-1.07e-04 18-17-1.85e-04 19-18-3.39e-04 20-19-6.74e-04 21-20 6.72e-03 22-21 1.01e-02 23-22 3.20e-02 24-23 4.38e-02
 25-24 5.79e-02

Old, new He 2 continuum optical depths:
      1 4.47e+06      2 1.12e+05      3 4.60e+03      4 3.68e+03      5 2.54e+03      6 1.69e+03      7 1.17e+03      8 8.24e+02
      9 5.77e+02     10 4.03e+02     11 2.88e+02     12 2.12e+02     13 1.61e+02     14 1.24e+02     15 1.01e+02     16 8.47e+01
     17 7.13e+01     18 7.60e+01     19 1.43e+02     20 3.02e+02     21 3.11e+02     22 2.24e+02     23 1.54e+02     24 1.05e+02
     25 8.75e+01
      1 4.38e+06      2 1.13e+05      3 4.52e+03      4 3.61e+03      5 2.50e+03      6 1.66e+03      7 1.15e+03      8 8.08e+02
      9 5.66e+02     10 3.95e+02     11 2.82e+02     12 2.08e+02     13 1.57e+02     14 1.22e+02     15 9.93e+01     16 8.31e+01
     17 7.00e+01     18 7.46e+01     19 1.40e+02     20 2.96e+02     21 3.05e+02     22 2.20e+02     23 1.51e+02     24 1.03e+02
```

---

[3]Line center optical depths were reported through version C10. Mean line optical depths are now given, although the code works with line center optical depths internally.

```
   25 8.58e+01

Old, new He 2 line optical depths:
 2- 1 7.60e+06  3- 2 6.42e-07  4- 3 2.62e-13  5- 4-1.24e-12  6- 5-5.54e-12  7- 6-1.45e-11  8- 7-2.98e-11
 9- 8-5.02e-11 10- 9-7.50e-11 11-10-9.49e-11 12-11-9.32e-11 13-12-3.54e-11 14-13 2.08e-10 15-14 1.07e-09 16-15-1.24e-09
17-16-2.30e-09 18-17-3.80e-09 19-18-6.67e-09 20-19-1.30e-08 21-20 2.38e-07 22-21 3.59e-07 23-22 1.20e-06 24-23 1.63e-06
25-24 2.15e-06
 2- 1 4.03e+06  3- 2 3.21e-07  4- 3 1.31e-13  5- 4-1.23e-12  6- 5-5.53e-12  7- 6-1.44e-11  8- 7-2.97e-11
 9- 8-5.01e-11 10- 9-7.48e-11 11-10-9.47e-11 12-11-9.29e-11 13-12-3.54e-11 14-13 1.04e-10 15-14 5.32e-10 16-15-1.23e-09
17-16-2.30e-09 18-17-3.80e-09 19-18-6.66e-09 20-19-1.30e-08 21-20 1.18e-07 22-21 1.78e-07 23-22 5.96e-07 24-23 8.09e-07
25-24 1.07e-06


Old He Is optical depths:  1 3.86e+07   2 4.04e+03   3 3.94e+03   4 3.80e+03   5 3.80e+03   6 3.80e+03   7 3.67e+03   8 2.16e+03
New He Is optical depths:  1 3.79e+07   2 3.96e+03   3 3.86e+03   4 3.72e+03   5 3.72e+03   6 3.72e+03   7 3.60e+03   8 2.12e+03
        Old He Is Lines: 2-1 9.19e+11 3-2 6.78e-09
        New He Is Lines: 2-1 4.51e+11 3-2 3.44e-09
```

Hydrogen and helium optical depths in continua and $\alpha(n \rightarrow n-1)$ transitions follow. The first block of lines are the optical depths assumed at the start of the present iteration and the second block gives the newly computed total optical depths. Negative optical depths indicate maser action. For each of the pairs the first block is the optical depth at thresholds of levels of hydrogen. The optical depths in the $\alpha(n \rightarrow n-1)$ transitions of hydrogen or helium follow.

```
 Line Optical Depths: 10830: 3.52e+01  3889: 1.51e+00   5876: 2.11e-08 7065: 2.88e-09  2.06m: 2.86e-05  21c: 5.35e-05
  H  1 1215A 5.15e+05 H  1 1025A 8.26e+04 H  1  972A 2.87e+04 H  1  949A 1.35e+04 H  1  937A 7.45e+03 H  1  930A 4.56e+03
  H  1  926A 3.00e+03 H  1  923A 2.08e+03 H  1  920A 1.51e+03 H  1  919A 1.12e+03 H  1  918A 8.62e+02 H  1  917A 6.75e+02
  H  1  916A 5.39e+02 H  1  915A 4.37e+02 H  1  915A 3.59e+02 H  1  914A 2.99e+02 H  1  914A 2.52e+02 H  1  914A 2.14e+02
  H  1  914A 1.83e+02 H  1  913A 1.58e+02 H  1  913A 1.37e+02 H  1  913A 1.20e+02 H  1  913A 1.06e+02 H  1  913A 9.34e+01
  --------
```

Line optical depths are not normally printed, but will be if the **print line optical depths** command is entered.

1.10.7 Column densities

```
                              Log10 Column density (cm^-2)
   Htot : 25.107   HII  : 21.167   HI   : 21.830   H-   : 12.687   H2g  : 24.806   H2*  : 16.421   H2+  : 11.203   HeH+ : 11.485
   H3+  : 13.820
   CH   : 17.799   CH+  : 12.136   OH   : 14.764   OH+  : 11.763   O2   : 17.909   CO   : 21.489   CO+  : 10.423   H2O  : 17.459
   H2O+ : 11.049   O2+  : 11.735   H3O+ : 14.429   CH2+ : 11.869   CH2  : 17.534   HCO+ : 15.394   CH3+ : 13.437   SiH2+: 11.835
   SiH  : 15.523   HOSi+: 14.069   SiO  : 17.953   SiO+ :  9.222   CH3  : 20.296   CH4  : 20.207   CH4+ : 11.132   CH5+ : 13.562
   N2   : 20.644   N2+  : 12.546   NO   : 15.954   NO+  : 12.873   S2   :  4.242   S2+  :  0.565   OCN  : 12.648   OCN+ :  9.954
   NH   : 14.634   NH+  :  9.266   NH2  : 16.135   NH2+ :  9.125   NH3  : 16.966   NH3+ : 12.225   NH4+ : 13.243   CN   : 18.325
   CN+  :  7.814   HCN  : 18.107   HCN+ :  8.175   HNO  :  8.826   HNO+ : 10.277   HS   : 14.673   HS+  : 13.458   CS   : 19.929
   CS+  : 10.371   NO2  : 10.136   NO2+ :  4.356   NS   : 13.418   NS+  : 12.254   SO   : 19.303   SO+  : 14.554   SiN  : 13.857
   SiN+ : 10.516   N2O  : 10.072   HCS+ : 15.996   OCS  : 19.077   OCS+ : 14.312   C2   : 20.023   C2+  : 11.133   CCl  : 14.010
   ClO  :  2.369   HCl+ : 11.968   HCl  : 17.061   H2Cl+: 12.002   CCl+ : 12.258   H2CCl+:10.100   ClO+ : -1.084   HNC  : 18.151
   HCNH+: 14.265   C2H  : 19.590   C2H+ : 10.670   C2H2 : 16.355   C2H2+: 15.324   C3H  :  4.558   C3H+ : -0.848   C2H3+: 11.612
   C3   :  3.013   C3+  : -6.584   COgrn : 17.793   H2Ogrn: 21.202   OHgrn:  8.957


              1     2     3     4     5     6     7     8     9     10    11    12    13    14    15    16    17

Hydrogen  21.830 21.167 25.107 (H2)              Log10 Column density (cm^-2)
Helium    24.085 20.117 17.105
Carbon    17.950 18.459 17.563 14.609 13.216  3.870
Nitrogen  19.044 16.241 16.933 14.065 11.478  5.692 -1.143
Oxygen    20.608 17.460 17.476 14.327 11.487  6.302  2.306
Neon      20.885 16.857 16.224 13.324 10.890  5.202  0.097
Magnesium 19.584 16.541 15.619 12.650 10.152  4.929  0.953
Silicon   19.701 16.932 15.711 14.515 13.010 10.127  3.809 -0.668
Sulphur   19.036 17.151 16.136 14.583 12.601 11.685 10.014  2.678 -2.437
Chlorine  18.066 14.800 14.129 11.924 10.392  9.305  4.190  1.698 -3.512
Argon     19.584 14.672 15.606 13.401 11.627 10.513  5.326  2.378 -0.441
Iron      19.584 16.655 15.236 15.417 12.498 11.847  9.292  5.178  1.303 -3.214
Exc state    He1* 14.756   CII* 18.248   C11* 17.606   C12* 17.569   C13* 17.067   O11* 20.607   O12* 18.042   O13* 17.426
             Si2* 16.187   C30* 12.885   C31*  8.857   C32* 13.149
```

This lists the column densities (cm^{-2}) of some atoms, ions, and molecules. The first number "Htot" is the total column density of hydrogen in all forms (including atoms, ions, and molecules). The following two numbers are the column densities in H$^+$ and H^0. The last four numbers are column densities in four ions and molecules (H$^-$, H$_{2g}$, H$_2^*$, H$_2^+$, H$_3^+$, and HeH$^+$). The remaining

lines give column densities in various molecules. Molecules with names ending in "grn" are solids that have condensed onto grain surfaces.

The next block gives column densities in atoms and ions of the heavy elements. For hydrogen the last number is the H_2 column density. Column densities within certain excited states of the heavy elements, listed in Table 8.1 on page 76, are also printed. The label gives the element, ionization stage, and level within the ground term. These upper levels are photoexcited by the so-called "level 2 lines". Their pumping will not be included and excited state column densities will not be predicted if the level 2 lines are disabled with the **no level 2 lines** command.

1.10.8   Mean ionization and temperature

```
              1      2      3      4      5      6      7      8      9     10     11     12     13     14     15     16

Hydrogen   -3.278 -3.940 -0.000 (H2)                   Log10 Mean Ionisation (over radius)
Helium     -0.000 -3.968 -6.980
Carbon     -3.635 -3.126 -4.021 -6.975 -8.368-17.715
Nitrogen   -1.908 -4.712 -4.019 -6.887 -9.474-15.261-22.096
Oxygen     -1.101 -4.249 -4.234 -7.382-10.222-15.407-19.403
Neon       -0.000 -4.028 -4.662 -7.561 -9.996-15.684-20.788
Magnesium  -0.000 -3.044 -3.966 -6.934 -9.432-14.656-18.631
Silicon    -0.008 -2.778 -3.999 -5.194 -6.700 -9.582-15.900-20.377
Sulphur    -1.071 -2.956 -3.971 -5.524 -7.507 -8.423-10.093-17.429-22.545
Chlorine   -0.041 -3.307 -3.978 -6.184 -7.716 -8.802-13.918-16.410-21.619
Argon      -0.000 -4.912 -3.979 -6.183 -7.957 -9.071-14.259-17.206-20.025
Iron       -0.001 -2.929 -4.348 -4.167 -7.086 -7.737-10.292-14.406-18.282-22.799

              1      2      3      4      5      6      7      8      9     10     11     12     13     14     15     16

Hydrogen   -1.292 -0.090 -0.864 (H2)          Log10 Mean Ionisation (over radius*electron density)
Helium     -0.622 -0.119 -3.265
Carbon     -2.105 -0.549 -0.176 -3.188 -4.552-14.000
Nitrogen   -0.770 -0.843 -0.174 -3.101 -5.664-11.546-18.381
Oxygen     -0.813 -0.378 -0.405 -3.646 -6.501-11.692-15.688
Neon       -0.730 -0.171 -0.855 -3.828 -6.272-11.969-17.073
Magnesium  -1.269 -0.741 -0.117 -3.184 -5.702-10.941-14.916
Silicon    -1.764 -0.643 -0.148 -1.382 -2.932 -5.840-12.185-16.662
Sulphur    -2.018 -0.696 -0.122 -1.719 -3.712 -4.627 -6.332-13.714-18.829
Chlorine   -0.856 -0.966 -0.130 -2.390 -3.910 -4.994-10.203-12.695-17.904
Argon      -0.732 -1.160 -0.130 -2.396 -4.146 -5.259-10.544-13.491-16.310
Iron       -1.472 -0.778 -0.478 -0.331 -3.339 -3.971 -6.562-10.691-14.567-19.083
```

The next blocks of output give the log of the mean ionization, averaged over volume, area (if the model is spherical), and over radius. The volume-averaged ionization fraction for ion i of element a is given by

$$\left\langle \frac{n_a^i}{n_a} \right\rangle_{vol} = \frac{\int n_a^i f(r)\, r^2\, dr}{\int n_a f(r)\, r^2\, dr}. \tag{1.15}$$

the area average by

$$\left\langle \frac{n_a^i}{n_a} \right\rangle_{area} = \frac{\int n_a^i f(r)\, r\, dr}{\int n_a f(r)\, r\, dr}. \tag{1.16}$$

and the radius average by

$$\left\langle \frac{n_a^i}{n_a} \right\rangle_{rad} = \frac{\int n_a^i f(r)\, dr}{\int n_a f(r)\, dr}. \tag{1.17}$$

Where n_a is the total gas-phase density and n_a^i is the density in ionization stage i. Similar blocks of information will give the mean ionization weighted by electron density and volume, area, or radius.

```
              1      2      3      4      5      6      7      8      9     10     11     12

Hydrogen    3.127  3.965  1.357 (H2)                   Log10 Mean Temperature (over radius)
Helium      1.372  3.961  3.937
Carbon      1.485  3.097  3.957  3.945  3.948  3.972
Nitrogen    1.949  4.004  3.956  3.945  3.951  3.972  3.972
```

```
Oxygen      1.586  3.983  3.946  3.956  3.963  3.972  3.972
Neon        1.370  3.968  3.944  3.956  3.961  3.972  3.972
Magnesium   1.357  2.953  3.963  3.951  3.957  3.972  3.972
Silicon     1.354  2.714  3.963  3.945  3.946  3.950  3.972  3.972
Sulphur     1.463  2.867  3.964  3.943  3.946  3.946  3.944  3.972  3.972
Chlorine    1.348  3.198  3.962  3.944  3.947  3.947  3.972  3.972  3.972
Argon       1.370  3.922  3.960  3.945  3.948  3.948  3.972  3.972  3.972
Iron        1.357  2.822  3.984  3.950  3.953  3.947  3.957  3.972  3.972  3.972

              1      2      3      4      5      6      7      8      9     10     11     12

Hydrogen    3.223  3.967  2.109 (H2)   Log10 Mean Temperature (over radius*electron density)
Helium      3.430  3.963  3.959
Carbon      2.234  3.742  3.958  3.945  3.949  3.972
Nitrogen    2.664  4.006  3.958  3.945  3.952  3.972  3.972
Oxygen      2.826  3.985  3.947  3.956  3.963  3.972  3.972
Neon        2.675  3.971  3.944  3.956  3.961  3.972  3.972
Magnesium   2.041  3.498  3.965  3.951  3.957  3.972  3.972
Silicon     1.496  3.474  3.965  3.945  3.946  3.950  3.972  3.972
Sulphur     1.721  3.369  3.966  3.944  3.946  3.946  3.943  3.972  3.972
Chlorine    2.140  3.842  3.964  3.944  3.947  3.948  3.972  3.972  3.972
Argon       2.631  4.016  3.962  3.945  3.949  3.949  3.972  3.972  3.972
Iron        1.803  3.140  3.986  3.952  3.953  3.947  3.956  3.972  3.972  3.972
```

The next blocks give the mean temperature weighted by volume, area, and radius. These are followed by the mean temperature weighted by volume, area, or radius and electron density. In all cases the volume and area averages will only be shown in spherical models.

1.10.9   Convergence statistics

```
--------------Convergence statistics--------------
       1.24 mean iterations/state convergence
       2.67 mean cx acceleration loops/iteration
       1.02 mean iso convergence loops/ion solve
          1 mean steps/chemistry solve
          2 mean step length searches/chemistry step
--------------------------------------------------
```

1.10.10   Final report

```
Cloudy ends: 196 zones,   1 iteration,  1 caution.  ExecTime(s) 86.97
[Stop in maincl, Cloudy exited OK]
```

The code ends by listing the number of zones and iterations that were performed and the number of warnings and cautions that occurred. Next comes the elapsed wall-clock time [s]. The last line will say "Cloudy exited OK" if the calculation is successful.

Chapter 2

OBSERVED QUANTITIES

2.1 Overview

This section describes how to convert the quantities actually used or predicted by CLOUDY into commonly observed ones.

2.2 Intensities of various continua

2.2.1 Incident radiation field

The incident radiation field is the light striking the cloud. The main printout printout gives the intensity of the incident radiation field with the label "Inci". The total continuum [units erg s^{-1} or erg cm^{-2} s^{-1}] integrated over all energies is given with this label and a wavelength of 0. The incident radiation field is also evaluated at two wavelengths, 4860 Å and 1215 Å, as λF_λ or νF_ν, [units erg s^{-1} or erg cm^{-2} s^{-1}].

The **save continuum** command will produce a file that contains the entire incident radiation field.

These entries will not be included if the **aperture** command is in effect.

2.2.2 The radiation field at specific wavelengths

The intensity of the diffuse radiation field that is emitted or reflected by the cloud is not normally included in the main output. The **print continuum** command will include this radiation field in the emission-line printout at a series of energies. These have units λF_λ or νF_ν. See the discussion of the **print continuum** and **set nFnu** commands in Part 1 for further details. You can easily add more continuum points using the **set nFnu add** command, also described in Part 1.

These entries will not be included if the **aperture** command is in effect.

2.2.3 The radiation field integrated over a range of wavelengths

The emitted radiation field integrated over a series of wavelength bands is included in the main printout. The file `continuum_bands.ini` in the data directory specifies a series of wavelength

bands. Table 9.1 on page 94 lists the default bands. The label and wavelength that are printed in the emission-line output are given in the first two columns of that table. The code will integrate over these bands to find the total radiated energy and report this in the main printout. The `continuum_bands.ini` file can be edited to change the number of bands or their detailed properties.

This gives the *total* emission that occurs over the band and includes the incident radiation field and both line and continuum diffuse emission. Under many circumstances the radiation field within a band may be dominated by the incident field. The code can also report the emissivity in any of the emission quantities given in the main printout. For these continuum bands the emissivity will only be the diffuse emission from the cloud and will not include the incident field.

These entries will not be included if the **aperture** command is in effect.

2.3   Emission-Line Equivalent Widths

The equivalent width of an emission or absorption line is the number of Angstroms of the continuum that is equivalent to the energy in the line. It is defined as

$$W_\lambda = \int \frac{F_\lambda^c - F_\lambda^t}{F_\lambda^c} d\lambda \approx -\lambda \frac{F_{line}}{\lambda F_\lambda^c} \quad [\text{units of } \lambda] \tag{2.1}$$

where the fluxes are in the incident continuum (F_λ^c) and the line (F_{line}). By this convention the equivalent width of an emission line is negative.

The code's output can be used to predict a line's equivalent width. The previous section describes several of the radiation fields that are predicted. The code prints the intensity or luminosity of all lines and continua and the intensity of each relative to a normalization line.

The ratio of a line to continuum intensity or luminosity will be the dimensionless ratio $F_{line}/\lambda F_\lambda^c$, part of the last term in equation 2.1. The line equivalent width is this ratio multiplied by the wavelength where the continuum is evaluated. For instance, you could trick the code into printing the relative intensities of the lines as an equivalent width relative to the incident radiation field at 1215 Å by including the command

```
normalize to ''Inci'' 1215 scale factor = 1215
```

This has two effects. It gives the intensities relative to the incident continuum at 1215 Å and multiplies this by the continuum wavelength in Angstroms, producing the rightmost ratio in equation 1.

A covering factor will complicate this slightly. (Covering factors are defined in the section *Definitions* in Part I of this document and in Section 5.9 of AGN3.) In the luminosity case partial coverage of the source is taken into account with the **covering factor** command and the luminosities are correct for this coverage. The ratio of line to continuum given in equation 2.1 will represent what is observed. In the intensity case the line intensity is given per unit area of cloud no matter what covering factor is specified. In this second case the ratio in equation 2.1 must be scaled by the covering factor.

2.4 Emission-Line Asymmetries

The inward fraction of the total emission of each line is always predicted by the code. It is not reported by default. Many lines are significantly inwardly beamed and this can lead to emission-line asymmetries if the envelope is expanding. The inward part of the lines will be printed if the **print line inward** command is entered. Note that the effects of this line beaming are very geometry dependent.

2.5 Surface Brightness

CLOUDY will normally predict a line's intensity as $4\pi J$, the intensity radiated into 4π sr by a unit area of cloud, with units erg s^{-1} cm^{-2}. Observations of resolved sources often measure the surface brightness, with units erg s^{-1} cm^{-2} arcsec^{-2} or s^{-1} cm^{-2} sr^{-1}. Be careful! Some workers may report surface brightness with units erg s^{-1} cm^{-2} arcsec^{-2} sr^{-1}. Remove the sr^{-1} before continuing by multiplying by 4π.

To obtain the predicted surface brightness we must divide the intensity $4\pi J$ by the number of square seconds of arc in 4π sr or the number of sr in 4π sr. One radian is $360/2\pi = 57.29578$ degrees, so 1 sr is $(180/\pi)^2 = 3282.806$ degree^2. There are $(60 \times 60)^2$ square seconds in a square degree, so there are 5.3464×10^{11} square arc seconds in 4π sr. The surface brightness (per square second of arc) is the intensity $4\pi J$ multiplied by the inverse of this, or 1.8704×10^{-12} arcsec^{-2}. The surface brightness (per sr) is the intensity $4\pi J$ divided by 4π.

Note that this is only correct for a line that is emitted isotropically, because the code predicts $4\pi J$ while an observer measures I along a particular direction. (The code does predict the fraction of a line that is emitted from the illuminated face of the cloud.) This discussion is only formally correct if $I = J$.

There is a **print line surface brightness** command, described in Section 15.21.17 *Print line surface brightness [arcsec]* of Hazy 1 which will change the intensity into surface brightness units. By default the final units will then be erg s^{-1} cm^{-2} sr^{-1}, but the command has an **arcsec** keyword to specify the surface brightness in erg s^{-1} cm^{-2} arcsec^{-2}.

2.6 Flux to luminosity

The luminosity is the intensity of a line multiplied by the total area of the shell. For full coverage this is $4\pi r^2$ where r is the radius of the shell. If the shell only partially covers the continuum source then this should be multiplied by the covering factor.

2.7 Flux at the Earth

If the distance to the object is specified with the **distance** command, and the luminosity case is predicted then the flux observed at the Earth will be predicted if the **print flux at Earth** command also appears.

2.8   Relative hydrogen line intensities

Hydrogen line intensities can be predicted with great precision when Case B applies. Ferguson and Ferland (1997) describe CLOUDY's original hydrogen atom and Ferland et al. (2009) describe its current implementation. It gives good results for all reported lines. The number of reported lines can be increased to go higher in the H I model atom by using the **atom H-like levels** command. This gives better results at the expense of more compute time.

   The accuracy of CLOUDY's H I line emissivities is limited by the size of the model hydrogen atom that can be computed on the fly. The definitive calculation for hydrogen recombination is that of Hummer and Storey (1987) and Storey and Hummer (1995), who used a 1000 level atom with all l-levels explicitly considered (that works out to something like a million levels!). The code interpolates on their tables and includes their Case A and Case B predictions within the main printout. The test cases limit_case*.in in the test suite compare the code's predictions with their Case B values.

   The Hummer and Storey (1987) calculation is for Case B conditions, which assume that many processes are unimportant (see AGN3). Neglected processes include collisional excitation from the ground or first excited states, induced processes where the incident continuum causes the atom to fluoresce, and line transfer in all non-Lyman lines. Case B is often an excellent assumption for galactic nebulae such as planetary nebulae or H II regions. Case B is not valid for gas densities greater than 10^6 cm^{-3} or when X-rays are present. When any of these processes are important the predictions of CLOUDY's model atom are more realistic than Case B predictions.

2.9   Helium line intensities

The code includes a model of the He^0 atom that is applied all along the helium iso-electronic sequence. The model can have an arbitrarily large number of levels (Bauman et al., 2005; Porter et al., 2005; Porter and Ferland, 2007). The predictions become more exact as the number of levels is increased.

2.10   Line Intensities in a dusty cloud

Two sets of line intensities are printed in the main output. The first block of lines, with the title "Intrinsic Intensities", includes all the physics of line formation, including destruction by background opacities such as grains or photoelectric absorption, but does not include the effects of absorption and scattering of the line from regions away from that were the line forms. This would be the spectrum you would observe after correction for reddening or line of sight extinction.

   The second block of emission-line intensities, with the title "Emergent Intensities", includes the effects of absorption and scattering from regions outside those where the lines form. The distinction is important for any cloud that has a large dust optical depth of A$_V$, for instance, an H$^+$ region on the surface of a molecular cloud.

   There are two simple limits for dust extinction. If the simulation only includes the ionized region of a photoionized cloud then extinction of optical or IR lines will be very small. This is because the total optical depth across an H$^+$ layer is $\tau \leq 1$ at 912 Å (AGN3, Chapter 2). Grain
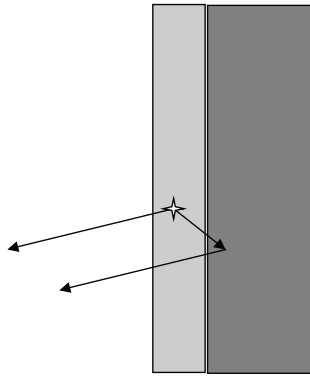
Figure 2.1: The geometry assumed in an open dusty geometry. The lightly shaded area is the H^+ region, and is a layer on the surface of an infinitely optically thick molecular cloud, the dark area on the right. Light can be emitted from the illuminated face of the cloud. A fraction of the light emitted towards the molecular cloud is reflected back towards the illuminated face.

extinction is much smaller in the visual than in the FUV around 912 Å, so the visual extinction will also be small. Extinction is even smaller in the IR due to the much smaller grain opacity there.

If the optical emission line spectrum is reddened it is almost certainly by layers of dust that lie outside the H^+ layer.

This geometry is appropriate for the Orion Nebula, a blister H II region on the surface of Orion Molecular Cloud 1 (OMC1). An idealized geometry is shown in Figure 2.1. The code computes the fraction of the line emission that is directed towards the illuminated and shielded faces. The outward line emission is emitted towards the PDR that may have a large optical depth due to grains. The albedo of the gas-grain mixture is computed and the fraction reflected is passed back towards the illuminated face. The total intensities are roughly half what would be expected were the cloud emitting from both sides. Something like 10% of the light striking the molecular cloud will be reflected back to the observer, and so slightly more than 50% of isotropically emitted lines will emerge from the illuminated face. The code uses the albedo of the gas at the wavelength of the line to predict this reflected portion and the full optical depths computed on previous iterations.

2.11 Continuum pumping contribution to line intensities

Continuum pumping or fluorescence is included for all lines. The contribution is only explicitly reported if the **print line pump** command is entered. Whether or not this contribution actually adds to the observed line emission depends on the geometry. Continuum pumping increases the line emission if no related continuum absorption occurs. This will be the case if the continuum source is either not observed or not covered by absorbing gas. If absorbing gas covers an observed continuum source then the situation is like the P Cygni problem, and pumping may not increase the net intensity of the line at all (the absorption component will have the same equivalent width as the associated emission). The printed line intensity includes this contribution unless the **no induced processes** command is entered. (The **no induced processes** command has many other

effects and so should only be used as a test.)

The output produced by the **save continuum** commands does not include the pumped part of the line contribution. This is correct if the continuum source is included in the beam, but is not if only the gas is observed.

2.12   Column densities

The column densities of all constituents are printed at the end of the calculation. Column densities within many excited states are also printed. The excited states are identified with a '*'. The table that accompanies the description of the *cdColm* command (see Table 8.1 on page 76 above) identifies the various labels.

2.13   A synthetic spectrum

A table of emission-line intensities is part of the normal output. Sometimes a synthetic spectrum, rather than a table, is desired. Very coarse spectra can be generated with the **save continuum** or **save spectrum** commands, but a detailed synthetic spectrum is not the main purpose of this output.

It is best to save the emission-line spectrum and then post-process this data using your own software. Then blends of lines can be synthesized at any spectral resolution desired. The spectrum can be save two ways. The main block of emission-line intensities in the final printout can be printed as a single column, which can be sorted by intensity or wavelength (by using options on the **print lines** command in Section 1 of this document). The **save spectrum** command includes a set of all lines with non-zero intensities. Write a small program or script to read these tables and create a final synthesized spectrum.

2.14   Line profiles

The observed line profile can be predicted by integrating the emissivity of the line over the computed structure while taking the local velocity structure into account. The emissivity is obtained with the **save lines emissivity** command, described in Part 1 of this document. This gives the net emission, with units erg cm^{-3} s^{-1}, emitted by a unit volume of gas and emergent from the cloud surface. The total emission is the integral of the emissivity. An integral over radius will give the line intensity $4\pi J$ while an integral over volume will give the luminosity.

The observed profile will depend on the velocity field at each point in the integration. For static models this will be the Voigt function at the local temperature and microturbulence. For a dynamical model it will include bulk motion of the gas.

Chapter 3

LIMITS, ASSUMPTIONS, AND RELIABILITY

3.1 Overview

This section outlines some of the assumptions and limits that define the range of validity of CLOUDY. The code is designed to be autonomous and self-aware, and to check that these limits are not exceeded during a calculation. This self-checking is a central feature of the code since it is designed to be used to compute large grids with thousands of models, where the examination of individual results would not be possible. CLOUDY should print a warning after the last zone results if any aspects of the calculation are in doubt.

3.2 Time steady

CLOUDY does not assume that the gas is in equilibrium. In most cases it does, however, assume that atomic processes occur on timescales that are much faster than other changes in the system, so that atomic rates have had time to become time-steady. It is possible to follow time-dependent conditions or an advective flow.

In practice most calculations assume that the cloud is old enough for atomic processes to have become time steady. The **age** command (described in Part I of this document) should be used to specify the age of the cloud. The code will then confirm that the time-steady assumption is valid by comparing the system's age with a host of rates and timescales, and will generate a warning if the environment is not time-steady.

Various time scales characterize the approach to equilibrium of an ionized gas (see Spitzer, 1962, and Ferland, 1979 for a specific application). Generally, for an ionized gas with nebular temperatures ($T \sim 10^4$ K), the longest is the H^+ recombination time scale,

$$T_{rec} = \frac{1}{\alpha_A(T_e)\, n_e} = 7.6 t_4^{0.8}\, n_4^{-1} \text{ years} = 0.66 t_4^{0.8}\, n_9^{-1} \text{ hours} \tag{3.1}$$

where t_4 is the temperature in units of 10^4 K, n_9 is the electron density in units of 10^9 cm^{-3}, and case A recombination is assumed (AGN3).

The time scales are far more ponderous in molecular regions. Generally among the longer of the time scales is the time to form H^-, an important pacesetter for H_2 formation in grain-free environments. This time scale is roughly given by

$$T_{molecule} = \frac{1}{\alpha_{rad}(T_e)\, n_e} = 0.3\, t_3^{-0.8}\, n_9^{-1} \text{ years} \tag{3.2}$$

where t_3 is the temperature in units of 10^3 K.

If the age of the cloud is not long enough for atomic processes to become time steady then a time-dependent calculation should be done. If the gas flow is fast enough for advection to add significant terms to the statistical equilibrium equations then a wind calculation should be performed. Commands for these simulations are described in Chapter 13, *DYNAMICAL & TIME-DEPENDENT CALCULATIONS*, in Hazy1.

3.3   Atomic and molecular database

This section outlines some of the atomic and molecular physics issues that affect the reliability of numerical simulations of nebulae. These uncertainties were extensively discussed in the Lexington Plasma 2000 meeting (ASP Conf series 247, *Spectroscopic challenges of photoionized plasmas,* Gary Ferland & Daniel Savin, editors) and they underscore the importance of atomic and molecular physics for the interpretation of astrophysical spectroscopy.

3.3.1   Collisional processes

By its nature, the electron temperature of a photoionized gas is low compared with the ionization temperature of the mixture of atoms and ions, as defined by the Saha equation (if the two were comparable, the gas would be collisionally ionized). Because of this, the rate coefficients describing collisional effects, such as the production of cooling emission lines, are often dominated by the cross sections near threshold. This is where laboratory experiments are difficult and *ab initio* quantum theory must often be used. As a result, the collision strengths undergo constant revision, towards better and more reliable values.

To cite one extreme example, the collision strength for transitions within the ^3P ground term of Ne^{+4} underwent three revisions between 1984 and 1991, each by a factor of 10, because of theoretical uncertainties in positions of autoionizing states with unknown energies (Lennon and Burke, 1991). The intensities of all emission lines can be affected by major changes in the atomic data for only one line for some conditions. This is because (in this case) the infrared fine structure lines of Ne^{+4} can be important coolants in low-density high-ionization gasses such as planetary nebulae, and changing their cooling rate alters the thermal structure of the entire nebula. Such changes often give even models of time-steady objects such as planetary nebulae certain time-dependent characteristics.

At present, there are fairly reliable calculations of collision strengths and transition probabilities for the majority of the strong optical and ultraviolet lines in moderate ionization nebulae. A series of papers by Oliva and collaborators (see Oliva et al., 1996 and van Hoof et al., 2000) outline observational evidence concerning accuracies in collision strengths of moderate ionization far infrared lines.

Some strong cooling lines of high ionization species do not have accurate collision rates. As an example, few of the "level2" lines included in the code have real collision strengths. Various forms of the "g-bar" approximation are used for those lines that do not have accurate collision rates.

3.3.2   Photoionization cross sections

The photoionization cross-section database has undergone a dramatic improvement with the completion of the Opacity Project (Seaton, 1987) and its fitting with analytic approximations (Verner et al., 1996). These are the photoionization cross sections used by CLOUDY and they should be as accurate as 10%. All inner shell multi-electron processes are included (Kaastra and Mewe, 1993) using distorted wave cross sections (referenced in Verner et al.). This part of the data base is in fairly good shape, although greater accuracy is always desired.

"Fano profiles", due to autoionization resonances, appear as large changes in photoionization cross section that occur over a narrow range of energy, are averaged over, as described by Verner et al. Their positions are not accurately known, and they could make a difference if sharp spectral features occur in the stellar continuum at the position of a resonance. The photoionization rate could be changed dramatically in this case. Experimental data would be needed to upgrade the photoionization data base to include Fano profiles.

3.3.3   Recombination rate coefficients

Recombination from closed shell species is accurately known (Verner and Ferland, 1996) since these are dominated by radiative recombination. Reliable dielectronic recombination coefficients do not now exist for most other stages of ionization. Currently there is no theory that can reproduce the best experiments (Savin et al., 1999). For these, CLOUDY uses the guestimates described in Part I of this document. This is clearly the greatest single gap in the atomic data base today. Savin (2000) shows an example where this uncertainty has a direct impact on cosmological studies.

3.3.4   Charge transfer

The rate coefficients for charge transfer are another uncertainty in the atomic and molecular database. This process is sometimes the dominant neutralization mechanism for singly or doubly ionized heavy elements. At present many charge exchange rate coefficients are the result of Landau-Zenner calculations using semi-empirical potential curves (Kingdon and Ferland, 1996, Kingdon and Ferland, 1999). These are thought to be no more accurate than a factor of three. Even the best quantal calculations are not thought to have an accuracy much better than 50 percent. Unpublished tests suggest that these uncertainties affect some line intensities at the $\sim$20% level, and a few by more than this.

3.4   Continuous opacity

All significant continuous opacity sources are treated for the energy range considered by the code, 1.001×10^{-8} Ryd to 7.354×10^6 Ryd. These opacity sources include inverse bremsstrahlung,

grains (when present), H^- absorption, electron scattering, the damping wings of strong resonance lines (i.e., Rayleigh scattering), pair production, photoelectric absorption by the ground and excited states of all ions of the lightest 30 elements, and photoabsorption by molecules. This treatment should be adequate as long as the optical depths to electron scattering are not large. CLOUDY is not now designed to simulate Compton-thick regimes. (A warning will be issued after the last zone calculation if the nebula is very optically thick to electron scattering.)

3.5   Temperature range

CLOUDY assumes that the electrons are non-relativistic, which limits it to temperatures below roughly 10^9 K. CLOUDY goes to the Compton temperature of the radiation field to great accuracy in the limit of very high levels of ionization for blackbody radiation fields with temperatures between 2.8 K and 1.001×10^{10} K. There is no formal lower temperature limit to its validity. Note that very cold gas is rarely in steady state, however, because of the very slow collision rates. Similarly, the collision timescales in very hot gas $(T \gg 10^8$ K$)$ are not rapid enough to ensure that the electrons and ions have the same temperature, or that heating - cooling balance has become time steady (Johnson and Quataert, 2007). Electron-ion decoupling is not now included.

The range of validity of the code is approximately from 10 K to 10^9 K. Temperatures outside this range can still be treated, although with greater uncertainty. The code will not permit temperatures below 2.8 K or above 1.001×10^{10} K.

3.6   Density range

There is no formal lower limit to the density that CLOUDY can treat. The set of heavy element fine structure lines, which dominate cooling at low densities, is complete for astrophysically abundant elements, and fine structure line optical depths, continuum pumping, and maser effects are fully treated.

There is no formal high-density limit, although the simulation is less complete at high $(n \cong 10^{10}$ cm^{-3} densities. Line radiative transfer processes are fully treated with either escape or Sobolev approximations (see Kalkofen, W., 1987 and Avrett and Loeser, 1988) and collisional-radiative ionization processes for excited levels of the heavy elements are treated. All species of H-like and He-like isoelectronic sequences are treated as many-level atoms, including all of the physical processes that allow the approach to LTE (see, for example, Mihalas, 1978). Tests with a hydrogen density of 10^{19} cm^{-3} show that the hydrogen and helium atoms and the hydrogen molecules go to LTE at high densities. They go to strict thermodynamic equilibrium when exposed to a true black body. The treatment of Stark broadening for hydrogen lines follows Puetter (1981), so line radiative transfer is treated correctly (in the context of the escape probability formalism) for densities above $\sim 10^{10}$ cm^{-3}.

The treatment of the other 28 isoelectronic sequences is presently not as complete as the H and He-like sequences. Three-body recombination is included as a general recombination process, so the treatment of these elements is approximately correct at high densities.

CLOUDY has been tested at densities of 10^{-8} cm^{-3} and 10^{19} cm^{-3} on 32-bit machines. The numerical (not physical) limit to the density will actually be set by the limits to the range of the

floating point numbers allowed by the machine in use. The physics incorporated in the code imposes no lower limit to the density. The physical high-density limit is roughly $10^{15} \, \text{cm}^{-3}$, and is set by the approximate treatment of three-body recombination - collisional ionization for the heavy elements.

3.7 Radiative transfer

Several tests presented in the test suite show that the continuum transfer methods are in excellent agreement with known exact results.

Line intensities are predicted with stellar atmosphere conditions in mind. Radiative transfer effects, including continuum pumping and possible maser emission, are treated. Nebular approximations, such as the approximation that all atoms are in the ground state, are not made. Collisional effects, including excitation and de-excitation, continuum fluorescence, recombination, etc, are all included as general line excitation mechanisms. The treatment of level populations is designed to go to LTE in the high particle or photon density cases.

The transfer of subordinate lines, those where both upper and lower level of the transition occur in excited state, of the H-like and He-like iso sequences at high ($n \gg 10^{10} \, \text{cm}^{-3}$) densities is a concern. Resonance lines should be fine.

3.8 Reliability

Several issues affect the general question of the reliability of the code. The first is the effects of the bugs that surely must exist in a code the size of CLOUDY. I have seldom found bugs in sections of the code older than roughly two to three years. Younger sections of the code sometimes contain bugs that only manifest themselves in exceptional situations. The issue of reliability in the face of complexity will increasingly be the single major problem limiting the development of large-scale numerical simulations (Ferland, 2001). New methods of writing code will have to be developed if we are to take full advantage of the power of future machines. Machines are getting faster more quickly than people are getting smarter.

The second issue is the validity of the numerical methods used to simulate conditions in the plasma. Fundamental uncertainties arise for cases where the density is high ($n \gg 10^{10} \, \text{cm}^{-3}$). The line radiative transfer techniques used by CLOUDY are approximate (see the discussion by Avrett and Loeser, 1988). Unfortunately, no definitive calculation now exists for the complete non-LTE equilibrium and line emission for an intermediate density $\sim 10^{13} \, \text{cm}^{-3}$ cloud. For less extreme conditions ($n < 10^{10} \, \text{cm}^{-3}$) nebular approximations are valid.

Test cases that are designed to exercise the code in well-posed limits and for certain standard nebulae are in the test suite. The code is well behaved and agrees with predictions of similar codes in these limits. The discussion presented in Ferland (1995) and Péquignot et al. (2001) suggests that 10% accuracy can be reached for the intensities of the stronger lines.

The uncertainties can probably best be judged by looking at both the dispersions among the various photoionization calculations presented in Ferland (1995) and Péquignot et al. (2001), and by the changes in the test suite. Much of the dispersion is due to improvements in the atomic database.

There can be little better way to close a discussion of reliability than to quote the warning included in the description by (Kurucz, 1970, see page xiii) of ATLAS5, a code more than an order of magnitude smaller than CLOUDY:

> ### *WARNING*
>
> "There is no way to guarantee that ATLAS5 does not contain errors. In fact, it is almost certain that it does, since the code is so long.... We also point out that the computation of a model atmosphere should be considered a physical experiment. The program may not be able to calculate a model for conditions that do not occur in real stars or for conditions that violate the initial assumptions on which the program is based."

3.9   The future

The eventual goal is for CLOUDY to give reliable results for all extremes of conditions between and including the intergalactic medium and stellar atmospheres. I estimate that the code is now well over halfway complete.

Current work centers on taking advantage of parallel machines. Once finished, we will be able to use this additional computational power to improve the treatment of hydrodynamics, line radiative transfer, and higher-order dimensionality.

Chapter 4

THE CONTINUUM

4.1 Overview

Under most circumstances the radiation field produced by the central object is the only source of heat and ionization. This section describes how this continuum is treated.

4.2 Attenuation of the incident continuum

In an ***open geometry*** scattering attenuates the incident continuum as

$$I = I_o(1 + 0.5\, d\tau_{scat})^{-1}.$$
(4.1)

Scattering does not affect the continuum in a ***closed geometry***. Absorption attenuates the incident continuum as

$$I = I_o \exp(-d\tau_{abs}).$$
(4.2)

for both geometries.

4.3 Recombination equilibrium

4.3.1 On-the-spot approximation

A modified version of the "on-the-spot" (OTS) approximation is used in the treatment of sources of diffuse ionizing radiation when the **diffuse OTS** command is used. Were no other opacity sources present, then, for a closed geometry that is optically thick in the Lyman continuum, all recombinations of hydrogen or helium to the ground state would produce ionizing photons. Other atoms of the recombined species would quickly absorb these. In this case OTS is an excellent approximation (van Blerkom and Hummer, 1967; Baessgen et al., 1988). However, other opacity sources are present, and these compete in absorbing protons produced by recombinations, making the recombination process more efficient than the OTS approximation would suggest.

The recombination coefficients are modified by the presence of all other opacity sources, such as grains, free-free or H^- absorption, and the heavy element opacities, in the following manner. The net effective recombination rate coefficient ($cm^3\ s^{-1}$) to level n, $\hat{\alpha}(T_e, n)$, is written in terms

of the spontaneous radiative recombination rate coefficient $\alpha(T_e, n)$ and the opacities (cm^{-1}) κ_n and κ_o for the level n and other opacity sources respectively, as

$$\hat{\alpha}(T_e, n) = \alpha(T_e, n) \left\{ P_c(n) + [1 - P_c(n)] \left(\frac{\kappa_o}{\kappa_o + \kappa_n} \right) \right\}, \tag{4.3}$$

where $P_c(n)$ is the continuum escape probability. In general, $P_c(n)$ varies between 0 and 0.5 for an optically thick open geometry (see, for example Davidson, 1977), $P_c \sim 1$ if the gas is optically thin, and $P_c \sim 0$ for ground states if the gas is optically thick and the geometry is closed. All computed opacity sources are included in κ_o.

These recombination continua produce a flux of local on-the-spot photons, φ_{OTS} $(\text{cm}^{-2}\,\text{s}^{-1})$. The OTS photoabsorption rate Γ_{OTS} (s^{-1}), used to determine the ionization or heating rate for the gas or grain constituents, is then $\Gamma_{OTS} = \alpha_v \varphi_{OTS}$ where α_v is the absorption cross section at frequency v. The OTS flux is related to the spontaneous recombination rate coefficient by

$$\varphi_{OTS} = \alpha(T_e, n)\ n_e n_{ion} \left[\frac{1 - P_c(\tau)}{\kappa_o + \kappa_n} \right] \quad \text{cm}^{-2}\ \text{s}^{-1} \tag{4.4}$$

where n_{ion} is the density of the ion. These are stored in the vectors *otscon* and *otslin*, which map one-to-one onto the vectors *flux* and *anu*.

4.3.2   Outward only approximation

A composite "outward-only" (Tarter, 1967) –"on-the-spot" approximation is used in the treatment of sources of diffuse ionizing radiation when the **diffuse outward** command is used. This is the default assumption. The escaping radiation is then propagated in the outward direction (all for the spherical case, and half for an open geometry).

4.4   Continuous opacities

The cloud is divided into a large number of concentric shells (zones) and the attenuated and diffuse continua and physical conditions are then determined within each.

The main opacity sources in the ultraviolet continuum are generally photoelectric and free-free (inverse bremsstrahlung) absorption, grain opacity, electron scattering (of both bound and free electrons), and the damping wings of Lyman lines (Rayleigh scattering). The main reemission mechanisms are generally free-free (bremsstrahlung), grain emission, free-bound, and two-photon emission. Grains are not present by default but can be added as an option. Continuous absorption and reemission by all ground states, and many excited states, of all ionization stages of the 30 elements in the calculation are explicitly included. Great care is taken to ensure that each absorption mechanism is balanced by a reemission process, and vice versa, so that energy balance in the strict thermodynamic equilibrium limit can be achieved.

4.4.1   Total opacity arrays

Total absorption opacities (cm^{-1}) are storied in the vector *opac*. Total scattering opacities (cm^{-1}) are stored in *scatop*. The opacities are evaluated in routine *ConvIonizeOpacityDo* and are within the *opac* structure (defined in `opacity.h`).

4.4.2 Cross-section array

Storage. The cross sections per particle (cm^2) for individual species (atoms, ions, molecules, etc) are stored within the array *OpacStack*, a stack array with a single dimension. These cross sections are evaluated when the code is initialized in routine *OpacityCreateAll*.

Array indices. Each species has an associated array index that defines the offset between the origin of *OpacStack*, the frequency array *anu*, and the opacity at the threshold. If this offset has the name *ioff*, for instance, then the cross section at threshold will be given by array element *OpacStack[ioff]*. If *ip* is the index to the threshold energy within *anu*, then the array index to the cross section at energy *i* will be $i - ip + ioff$.

Individual cross-sections. The function *csphot* returns the cross section at a specific frequency for any species. It has three arguments, 1) the pointer to the frequency in *anu* where the cross section is to be evaluated, 2) the pointer to the threshold for the species, and 3) the *ioff* offset described above. All are integer variables.

4.4.3 Photoionization rates

Photoionization rates (units s^{-1}) can be computed by several functions. Which is used at a particular time is determined by circumstances.

GammaK This computes the photoionization rate with allowance for an arbitrary fluorescence yield. This routine is a major pacesetter for the code since it is used to evaluate the continuum rates in the majority of the cases. The photoionization rate is given by

$$\Gamma_n = 4\pi \int_{\nu_o}^{\infty} \frac{J_\nu}{h\nu} \alpha_\nu \, d\nu \quad [s^{-1}]. \tag{4.5}$$

where α_ν is the photoionization cross section [cm^{-2}]. The routine has three integer arguments, the *anu* pointers to the lower and upper energies and the offset to the opacity array *ioff* (described above).

GammaPrt This is a special version of *GammaK* that writes the step by step results of the integration on any open file. The output lists the product of the photon flux and the cross section, the photon flux, and the opacity.

GammaBn This is a special version of *GammaK* that is used when the correction for stimulated emission or induced recombination is important. The photoionization rate is given by

$$\Gamma_n = 4\pi \int_{\nu_o}^{\infty} \frac{J_\nu}{h\nu} \alpha_\nu \, d\nu \quad [s^{-1}] \tag{4.6}$$

and the rate for induced recombination and its associated cooling is computed as

$$\alpha(ind) = P_n^* 4\pi \int_{\nu_o}^{\infty} \frac{J_\nu}{h\nu} \alpha_\nu \exp(-h\nu/kT) \, d\nu \quad [cm^3 \, s^{-1}]. \tag{4.7}$$

where P^* is the LTE population.

GammaPrtRate will print photo rates for all shells of an ion and element. It is called with three arguments, a file handle, followed by the ionization stage and element number on the C scale (0 for H or atoms, etc).

4.4.4   Attenuation within the zone

A correction must be made to account for the attenuation of the continuum across the zone (Netzer and Ferland, 1984). Assuming that the continuum varies across the zone as

$$\frac{I(\nu, \delta r)}{I_o(\nu)} = \exp\left(-\kappa(\nu)\, f(r)\, \delta r\right) \tag{4.8}$$

then the intensity averaged over a zone with thickness δr is

$$\left\langle \frac{I(\nu, \delta r)}{I_o(\nu)} \right\rangle = \frac{1 - \exp\left(-\kappa(\nu)\, f(r)\, \delta r\right)}{\kappa(\nu)\, f(r)\, \delta r} \tag{4.9}$$

where $\kappa(\nu)$ is the absorption opacity and $f(r)$ is the filling factor. The coefficients giving this ratio as a function of energy are stored in the vector *tmn*, and are evaluated in routine *radinc*. The continuum stored in *flux* is multiplied by these factors in the same subroutine.

4.4.5   Rayleigh scattering

Clouds with neutral hydrogen column densities greater than $10^{23}\,\mathrm{cm}^{-2}$ are optically thick to Rayleigh scattering at wavelengths near $L\alpha$, and this process is a major scattering opacity source at short wavelengths for grain-free environments.

Rayleigh scattering cross sections given by Gavrila (1967) are used, joined with expressions for the radiative damping wings of Lyman lines (Mihalas, 1978). For wavelengths longward of 1410 Å a power-law fit to Gavrila's quantal calculations is used;

$$\sigma_{Ray} = 8.41 \times 10^{-25}\varepsilon^4 + 3.37 \times 10^{-24}\varepsilon^6 + 4.71 \times 10^{-22}\varepsilon^{14} \quad \mathrm{cm}^2 \tag{4.10}$$

where $\varepsilon \equiv /cR_\infty$ is the photon energy in Rydbergs. This fit is accurate to typically a percent, with occasional errors as large as 4 percent.

For wavelengths between 1410 Å and the Lyman limit, radiative broadening of the Lyman lines is assumed (Mihalas, 1978);

$$\sigma_{Ray} = \sum_{i=2}^{4} \left(\frac{q_e^2\, f_{1,i}}{m_e c} \right) \frac{\Gamma/4\pi}{(\nu - \nu_{1,i})^2} \quad \mathrm{cm}^2 \tag{4.11}$$

where Γ is the reciprocal lifetime of the upper level i and the sum is over the first four Lyman lines. This expression gives cross sections in excellent agreement with Gavrila (1967) for these wavelengths.

4.4.6   Free-free opacity

The main opacity source in the infrared-radio spectral region for many conditions is free-free opacity with a cross section given by

$$\alpha_\nu(ff) = 3.69 \times 10^8\, \bar{g}_{III}(\nu, T)\, f(r)\, \nu^{-3}\, T^{-1/2} \left\{1 - \exp\left(-h\nu/kT\right)\right\} \sum_A \sum_z z^2\, n_A^{+z} \quad [\mathrm{cm}^{-2}] \tag{4.12}$$
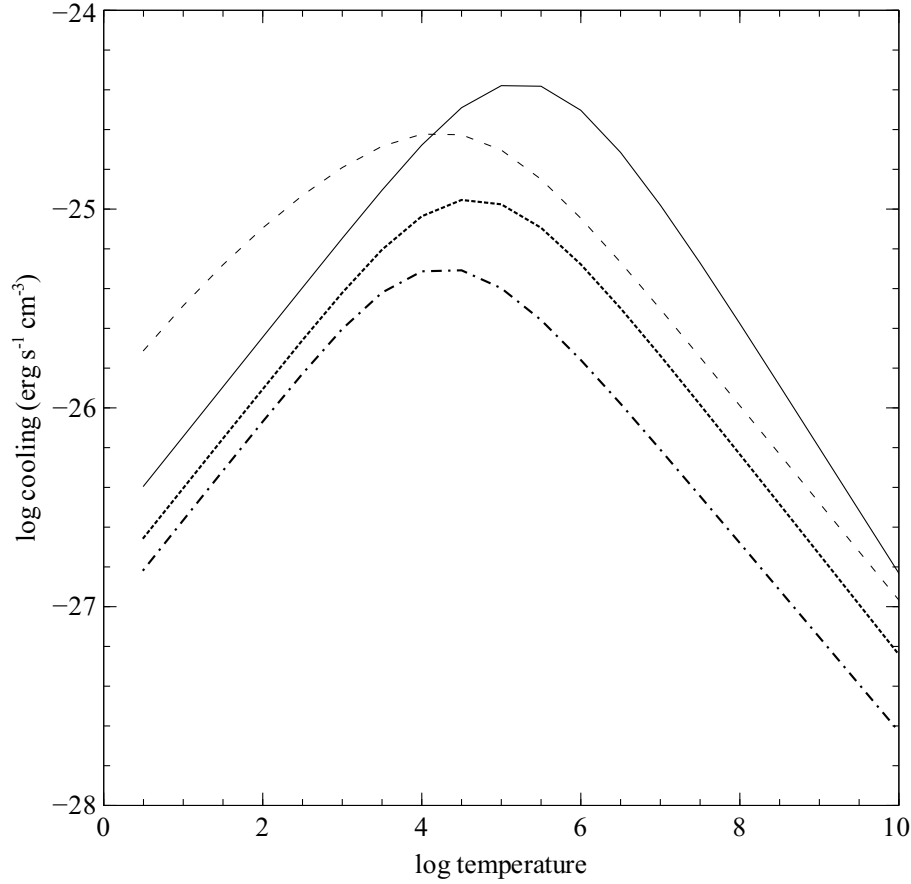
Figure 4.1: Thermally averaged free-free gaunt factor. The gaunt factor is shown as a function of photon energy and temperature.

(see, for example, Mihalas, 1978). The sum is over all ions n^{+z} of element A and over all elements. The temperature averaged gaunt factor $\bar{g}_{III}(\nu, T)$ is taken from Hummer (1988) (see also Karzas and Latter, 1961) and are evaluated in routine *gffsub* that was originally written by D. Hummer.

This routine did not extend to energies that could be treated by asymptotic expansions of the gaunt factor. *gffsub* was modified by J. Ferguson to extend over the full temperature and energy range considered by CLOUDY, and later extensively rewritten by Ryan Porter. Figure 4.1 shows the gaunt factors as functions of photon energy and temperature.

4.4.7 Bound-free opacity

Continuum optical depths for photoabsorption from level l are given by

$$d\tau_l(\nu) = \alpha_\nu(n) \, n_l \left[1 - \exp(-h\nu/kT)/b_l \right] \, f(r) \, \delta r \tag{4.13}$$

where b_l is the departure coefficient for level l and α_ν is the absorption cross section [cm^{-2}].

4.4.8   Plasma frequency

The plasma frequency, the energy where the index of refraction of an ionized medium goes to zero, is given by

$$v_{pl} = \left(\frac{n_e q_e^2}{\pi m_e} \right)^{1/2} = 8.978 \times 10^3 \, n_e^{1/2} \, s^{-1} = 2.729 \times 10^{-12} \, n_e^{1/2} \, \text{Ryd} \qquad (4.14)$$

An ionized gas will reflect the incident continuum for energies smaller than this. This shielding becomes important for the energy range considered by CLOUDY for electron densities greater than $\sim 10^7 \text{cm}^{-3}$. For higher densities this process is treated by setting the intensity of the incident continuum to zero for energies below the plasma frequency, adding this portion of the incident continuum to the reflected continuum, and not allowing emission or absorption for any processes that occur below the plasma frequency.

4.4.9   Pressure lowering of ionization potential

The electric field of nearby charges in the continuum acts to lower the ionization potential. The amount by which it is lowered is determined by the electron density. Ryan Porter extended the code to consider all species treated with the iso-electronic model atoms.

4.5   Continuum range

The energy interval $1.001 \times 10^{-8} \, \text{Ryd} - 7.354 \times 10^6 \, \text{Ryd}$ is divided into a large number of energy cells with nearly logarithmically increasing widths.

4.6   The continuum mesh

4.6.1   Continuum mesh logic

The central frequencies of two cells are related by

$$\frac{v_{i+1}}{v_i} = \exp(r) \qquad (4.15)$$

where r is the resolution, $\delta v / v$. Then the nth cell energy is related to the first cell energy by

$$v_n = v_0 \exp(nr). \qquad (4.16)$$

The cell corresponding to energy v_n is then

$$n = \frac{\log(v_n/v_0)}{r}. \qquad (4.17)$$

4.6.2   Defining the continuum energy mesh

The array *anu* gives the energy of the center of each continuum cell, in Rydbergs. This energy scale is defined in routine *ContCreatePointers*.

4.6.3 Changing the energy resolution of the mesh

The file `continuum_mesh.ini` contains ordered pairs of continuum energies and resolving powers (defined as $1/r$) that are read by *ContCreatePointers* to set the continuum mesh when calling *fill*. Change the contents of `continuum_mesh.ini` to change the resolution of the continuum mesh. The file explains how to do this.

   If the energy resolving power is increased then the code will require more mesh points to cover the full continuum and will run more slowly, but the predicted continuum will have greater detail.

4.7   Continuum generation

The continuum is generated by the function *ffun*. *ffun* has a single argument, the energy in Rydbergs, and it returns the number of photons per unit area, time, and Rydberg, at that energy. *ffun* sums over all the specified continua and applies the appropriate normalization factors. Another function, *ffun1*, evaluates each individual continuum, and is normally called only by *ffun*.

   The units, and their conversion to other measures of the continuum, are given below. The photon flux density is:

$$\varphi_\nu(\nu) = \text{ffun}(\nu) \quad [\text{photons cm}^{-2}\,\text{s}^{-1}\,\text{Ryd}^{-1}]. \tag{4.18}$$

This is stored in the photon array:

$$flux(\nu_i) = \varphi_\nu(\nu)\,\delta\nu_i = \text{ffun}(\nu_i) \times widflx(\nu) \quad [\text{photons cm}^{-2}\,\text{s}^{-1}] \tag{4.19}$$

where *widflx* is an array containing the width of each continuum bin. Finally, the energy flux density is given by

$$f_\nu(\nu) = \text{ffun}(\nu)\,h\left(\frac{\nu}{\nu_{912}}\right) \quad [\text{erg cm}^{-2}\,\text{s}^{-1}\,\text{Hz}^{-1}] \tag{4.20}$$

and

$$\nu f_\nu(\nu) = \text{ffun}(\nu)\,h\left(\frac{\nu}{\nu_{912}}\right)\nu_{912}\,h\nu_{Ryd} \quad [\text{erg cm}^{-2}\,\text{s}^{-1}]. \tag{4.21}$$

4.8   Energy units; the Rydberg

Continuum energies are usually given in Rydbergs. The energy of level n of a hydrogenic atom is given by

$$E_n = -\frac{\mathscr{R}_H}{n^2}, n = 1,2,3,\ldots, \quad [\text{units of } \mathscr{R}_H] \tag{4.22}$$

where the Rydberg energy is given by

$$\mathscr{R} = \frac{\mu q_e^4}{2\hbar^2} = \frac{\mu}{m_e}\mathscr{R}_\infty \tag{4.23}$$

and the reduced mass of the orbiting electron μ is related to the electron and nuclear mass m_e and m_{nuc} by

$$\mu = \frac{m_e m_{nuc}}{m_e + m_{nuc}} = \frac{m_e}{1 + m_e/m_{nuc}} \quad [\text{g}], \tag{4.24}$$

(Friedrich, 1998). Note that as $m_{nuc} \to \infty$, $\mu \to m_e$. This Rydberg energy $\mathscr{R}_H$ is smaller than the energy for an infinite mass nucleus $\mathscr{R}_\infty$ by the ratio μ/m_e. Using the 1998 CODATA revision of the fundamental constants (see Cohen and Taylor, 1987; Mohr and Taylor, 2000) the infinite mass Rydberg energy is given by

$$\mathscr{R}_\infty = \frac{m_e q_e^4}{2\hbar^2} = 13.605698 [\text{eV}] \tag{4.25}$$

and

$$\begin{aligned}
\mathscr{R}_\infty/(2\pi\hbar c) &= 109737.315686 \text{ cm}^{-1} \\
\mathscr{R}_\infty/(2\pi\hbar) &= 3.28984196038 \times 10^{15} \text{ Hz } . \\
(2\pi\hbar c)/\mathscr{R}_\infty &= 91.126732 \text{ nm}
\end{aligned} \tag{4.26}$$

The ionization potential of hydrogen $\mathscr{R}_H$ is μ/m_e or ~ 0.99946 times smaller than $\mathscr{R}_\infty$. [1] We have

$$\begin{aligned}
\mathscr{R}_H &= 13.59842 \text{ eV} \\
\mathscr{R}_H &= 2.178728 \times 10^{11} \text{ erg} \\
(2\pi\hbar)/\mathscr{R}_H &= 91.176430 \text{ n}m \\
\mathscr{R}_H &= 109677.576 \text{ c}m^{-1}
\end{aligned} \tag{4.27}$$

The difference between $\mathscr{R}_H$ and $\mathscr{R}_\infty$ is significant since it enters as the third power in the photon phase-space conversion factor $2h\nu^3/c^2$.

The Bohr radius is for an infinite mass nucleus is given by

$$\begin{aligned}
a_o &= \hbar^2/(m_e q_e^2) = 0.5291772083 \times 10^{-8} \\
a &= a_o/\mu
\end{aligned} \quad [\text{cm}]. \tag{4.28}$$

In the "atomic units" system of measuring quantities lengths are given in terms of the Bohr radius and energies in "Hartree" units. One Hartree is twice the Rydberg energy.

---

[1] $\mathscr{R}_H$ was the Rydberg unit used by CLOUDY before 1988.

Chapter 5

LINE ATOMIC PARAMETERS

5.1 Overview

Many atomic physics quantities describe how matter and light interact. This section goes over these quantities and how they are related to one another. The inter-relations between these quantities are described in most spectroscopy texts, at about the same depth as is given below. Hilborn (1982) gives a far more formal description, often tracing quantities back to basic E&M concepts. Highly recommended.

5.2 Spectroscopic notation

A great deal of confusion exists over the difference between the designation of a spectrum and an ion. "Atom II" denotes a spectrum, a collection of photons, while "Atom$^+$" denotes a baryon, the element "Atom" with a single electron removed.

   Much of the notation in today's atomic physics was developed in the second half of the 19th century. Physicists noticed that the spectrum of a gas would change dramatically when it was heated to high temperatures. They did not understand the reason why this happened, the electron had not yet been discovered, but they developed the notation that "Atom I" was the normal spectrum, this spectrum changed to "Atom II" when the gas was heated, and became the "Atom III" spectrum if heated still further. The "Atom II" spectrum was often called the "enhanced" spectrum of Atom. This is the reason why, in classical novae, the appearance of broad absorption lines of singly ionized species is called the "diffuse enhanced phase".

   The electron was discovered well after this notation had been developed. By the early 20th century it was understood that the "I" spectrum was produced by the atom, Atom^0. At high temperatures the first ion, Atom$^+$, formed and produced the "Atom II" spectrum.

   During the first half of the 20th century astrophysicists mainly studied stellar absorption lines. This led to the commonly-used notation that the "Atom II" spectrum, for instance, measured Atom$^+$. It is unambiguously true that the equivalent width of an Atom II absorption line is proportional to the column density of Atom$^+$. However there is an ambiguity in emission lines. The Lα H I line can be produced by impact excitation of H^0 or by recombination of H$^+$. In both cases the line is an H I line, but H I is produced by either H^0 or H$^+$, depending on details.

   To be unambiguously correct you should refer to Atom I or Atom II when discussing the

spectrum. When discussing a column density or particle density you should refer to the ion, as in $Atom^0$ or $Atom^+$. It is not correct to refer to the column density of H I. The $L\alpha$ H I emission line measures the column density of either H^0 or H^+, depending on whether the line forms by impact excitation or recombination. The notation is not ambiguous in absorption lines, which is probably why we have such confusion today. Be unambiguous and correct - use the right notation!

5.3   Line absorption

5.3.1   Line optical depths

The optical depth for a transition $u - l$, where u and l are the upper and lower levels, is given by

$$d\tau_{l,u} = \alpha_v \left(n_l - n_u g_l / g_u \right) f(r) \, dr [\text{Napier}]. \tag{5.1}$$

Here $f(r)$ is the filling factor and α_v is the atomic absorption cross section [cm^2].

The term in parenthesis is the population [cm^{-3}] of the lower level, with a correction for stimulated emission. This term is the only place where stimulated emission enters in the radiative balance equations (Elitzur et al., 1983).

5.3.2   Oscillator strengths

The oscillator strength f is a dimensionless number of order unity that can be thought of as a correction factor to make the expression for a classical oscillator agree with the quantum mechanical value. Sections below relate the oscillator strength to other line parameters such as the absorption coefficient and the transition probability. The absorption (f_{abs}, called $f_{l,u}$ here) and emission (f_{em}, called $f_{u,l}$) oscillator strengths are related by

$$g_l f_{l,u} = -g_u f_{u,l} \tag{5.2}$$

where the g's are the statistical weights. This product is symmetric, neglecting sign, and the code tries to use gf's throughout. The convention is that emission lines have negative oscillator strength.

5.3.3   Absorption cross section

The line-center absorption cross section [cm^2] is related to the dimensionless absorption oscillator strength f_{lu} or f_{abs} by

$$
\begin{aligned}
\alpha_v &= \frac{\pi^{1/2} q_e^2 \lambda \, f_{l,u}}{m_e c u_{Dop}} \varphi_v(x) \\
&= 0.014974 f_{l,u} \frac{\lambda_{cm}}{u_{Dop}} \varphi_v(x) = 1.4974 \times 10^{-6} f_{l,u} \frac{\lambda_{\mu m}}{u_{Dop}} \varphi_v(x)
\end{aligned}
\quad [cm^2] \tag{5.3}
$$

with the relative line displacement given by

$$x \equiv \frac{v - v_o}{\Delta v_{Dop}}, \tag{5.4}$$

$\varphi_v(x)$ is the Voigt function, and u_{Dop} is the Doppler velocity width (cm s^{-1}), the point where the line profile falls to 1/e of its peak. With this definition of the relative line displacement, the line profile due to thermal motions alone is $exp(-x^2)$. Equation 5.3 is evaluated in routine *abscf*.

5.4 The line profile function

5.4.1 Velocities in a thermal distribution

The distribution function for a Maxwellian velocity distribution is given by

$$\frac{n(u)\,du}{n} = \frac{1}{\pi^{3/2}} \exp\left[-u^2 m_A/2kT\right] \left(\frac{m}{2kT}\right)^{3/2} 4\pi u^2\,du \quad [\text{cm s}^{-1}] \tag{5.5}$$

(Novotny, 1973; p 122). There are three mean speeds in a thermal velocity distribution. The ***most probable speed*** is the peak of the velocity distribution, with a value

$$u_{mean}^2 = 2kT/m_A \quad [\text{cm}^2\ \text{s}^{-2}]. \tag{5.6}$$

This is found by setting the derivative of the distribution function to zero (Novotny, 1973, p 122). The velocity distribution function can be expressed in terms of the ***mean speed*** as

$$\frac{n(u)\,du}{n} = \frac{1}{\pi^{3/2}} \exp\left[-u^2/u_{mean}^2\right] \frac{4\pi u^2}{u_{mean}^3}\,du \quad [\text{cm s}^{-1}]. \tag{5.7}$$

The ***average speed*** is obtained by averaging over this function and is given by

$$u_{average}^2 = 8kT/\pi m_A \quad [\text{cm}^2\ \text{s}^{-2}]. \tag{5.8}$$

The ***thermal Doppler velocity width*** is the velocity averaged over the projected line of sight, given by (Novotny, 1973; p 204)

$$u_{th}^2 = 2kT/m_A \quad [\text{cm}^2\ \text{s}^{-2}]. \tag{5.9}$$

This is the distance from line center where the line profile falls to e^{-1} of its central value. So it turns out that the most probable speed is equal to the Doppler velocity width.

The ***velocity dispersion*** σ is

$$\sigma = u/\sqrt{2} \quad [\text{cm s}^{-1}] \tag{5.10}$$

and appears in the Gaussian profile function as

$$\varphi(\delta u) = \exp\left(-\delta u^2/2\sigma^2\right). \tag{5.11}$$

5.4.2 Micro vs macro turbulence

Micro-turbulence (hereafter, just turbulence) is due to any additional motions that occur over scales that are smaller than a photon mean free path. Micro-turbulence changes the line transfer since the line opacity is distributed over a broader range of velocities. Macro-turbulence is due to motion that occurs over such large scale lengths that they not change the optical depth through the emitting region. An example might be cloud orbital motions. The transfer within a cloud is not changed by its bulk motion and so would be considered macro-turbulence.

5.4.3   Line Widths

If a non-thermal micro-turbulent component of motions is present then equation 5.3, the total Doppler velocity width [cm s^{-1}] including turbulence, is given by

$$u_{tot}^2 = 2kT/m_A + u_{turb}^2 \quad [\text{cm}^2 \text{ s}^{-2}] \tag{5.12}$$

as determined by the local kinetic temperature T. Within the code the micro-turbulent velocity u_{turb} is assumed to be zero unless it is reset with the **turbulence** command.[1]

In *GetDopplerWidth* the Doppler velocity width is evaluated as

$$u_{Dop} = \sqrt{2kT/m_A + u_{turb}^2} = \sqrt{1.651 \times 10^8 \, T/m_{AMU} + u_{turb}^2} \quad [\text{cm s}^{-1}]. \tag{5.13}$$

The atomic weight is in atomic mass units.

The Doppler velocity width is related to the half width at half maximum by (Novotny, 1973, eqns 5-18; p 205)

$$\Delta u_{1/2} = (\ln 2)^{1/2} u_{Dop} = 0.832555 \, u_{Dop} \quad [\text{cm s}^{-1}] \tag{5.14}$$

and the FWHM is given by

$$\Delta u_{FWHM} = 2(\ln 2)^{1/2} u_{Dop} \quad [\text{cm s}^{-1}]. \tag{5.15}$$

5.4.4   The Doppler b parameter

Much of the literature refers to the Doppler b parameter. This is the Doppler velocity width or velocity dispersion with turbulence included, and is given by

$$b^2 = u_{Dop}^2 = 2kT/m_A + u_{turb}^2 \quad [\text{cm}^2\text{s}^{-2}]. \tag{5.16}$$

With these definitions

$$b = u_{Dop} = \Delta u_{FWHM} / \left[2(\ln 2)^{1/2} \right] \quad [\text{cm s}^{-1}]. \tag{5.17}$$

5.4.5   Voigt function

Optical depths a relative displacement x away from line center are related to the line center optical depth τ_0 by

$$\tau(x) = \tau_0 \varphi_v(x). \tag{5.18}$$

The relative displacement is given by equation 5.4 above. The Voigt function is normalized to unity at line center and is approximately given by

$$\varphi_v(x) \approx \exp(-x^2) + a/(\pi^{1/2} x^2). \tag{5.19}$$

Here a is the damping constant and the expression is valid only for small a. At line center $\varphi_v(a,x) \sim (1+a)^{-1}$, while the core of the line is roughly $x \sim (1+a)$ wide. The line center optical depth thus depends on a when $a \geq 1$, while the mean optical depth is constant.

---

[1]Note that the **turbulence** command accepts u_{turb} in km s^{-1} but converts it into cm s^{-1}, the units used throughout the code.

5.4.6 Mean vs. line center optical depths

CLOUDY works with line center optical depths throughout (see, for example, Mihalas, 1978). However, mean optical depths are reported in the final printout. For comparison, the line center optical depth is $\pi^{1/2}$ times *smaller* than the mean optical depth when the damping constant a is small, $a \ll 1$.

In many places routines or approximations using *mean* optical depths are encountered (e.g., Hummer and Kunasz, 1980). The difference is in how equation 5.19 is normalized.

5.5 The Einstein coefficients

The dimensionless oscillator strength gf is related to the transition probability A_{ul} [s^{-1}] by

$$g_l f_{l,u} = g_u f_{u,l} = \frac{m_e c \lambda_{cm}^2}{8\pi^2 q_e^2} g_u A_{u,l} = 1.4992 g_u A_{u,l} \lambda_{cm}^2 = 1.4992 \times 10^{-8} g_u A_{u,l} \lambda_{\mu m}^2 \qquad (5.20)$$

where $\lambda_{\mu m}$ is the wavelength in microns and λ_{cm} the wavelength in centimeters. The absorption oscillator strength is related to the transition probability by

$$f_{l,u} = \frac{m_e c \lambda_{cm}^2}{8\pi^2 q_e^2} \frac{g_u}{g_l} A_{u,l} = 1.4992 \times 10^{-8} A_{u,l} \lambda_{\mu m}^2 \frac{g_u}{g_l} \qquad (5.21)$$

or

$$A_{u,l} = \frac{8\pi^2 q_e^2}{mc\lambda_{cm}^2} \frac{g_l}{g_u} f_{abs} = \frac{f_{l,u}}{1.4992 \times 10^{-8}} \lambda_{\mu m}^{-2} \frac{g_l}{g_u} \quad [\text{s}^{-1}]. \qquad (5.22)$$

Combining equations 5.3 and 5.21 we obtain an expression relating the transition probability and the absorption cross section;

$$\begin{aligned}
\alpha_v &= \frac{\lambda^3 g_u}{8\pi g_l} \frac{A_{u,l}}{\pi^{1/2} u_{Dop}} \varphi_v(x) \\
&= \frac{\lambda^3 g_u}{8\pi g_l} \frac{A_{u,l}}{\pi^{1/2} u_{Dop}} \varphi_v(x) \qquad [\text{cm}^2]. \\
&= 2.24484 \times 10^{-14} A_{u,l} \lambda_{\mu m}^3 \frac{g_u}{g_l} \frac{\varphi_v(x)}{u_{Dop}}
\end{aligned} \qquad (5.23)$$

The coefficient for induced emission, B_{ul}, is related to A_{ul} by the phase space factor $2h\nu^3/c^2$;

$$A_{u,l} = \frac{2h\nu^3}{c^2} B_{ul} \quad [\text{s}^{-1}] \qquad (5.24)$$

and the induced emission and absorption probabilities are related by

$$g_l B_{l,u} = g_u B_{u,l} \qquad (5.25)$$

The absorption cross section α_v is related to $B_{l,u}$ by

$$\alpha_v = \frac{hc}{4\pi^{3/2}} \frac{B_{l,u}}{u_{Dop}} \varphi_v(x) \quad [\text{cm}^2]. \qquad (5.26)$$

In these terms the optical depth increment (equation 5.1) is given by

$$\begin{aligned}
d\tau_{l,u} &= \alpha_v (n_l - n_u g_l/g_u) f(r) \, dr \\
&= \frac{hc}{4\pi^{3/2}} \frac{B_{l,u}}{u_{Dop}} \varphi_v(x) (n_l - n_u g_l/g_u) f(r) \, dr
\end{aligned} \quad . \qquad (5.27)$$

5.6   Continuum pumping

5.6.1   Photon occupation number

The intensity of a radiation field can be thought of as two parts, the available volume of phase space $2h\nu^3/c^2$ and a dimensionless occupation number η giving the fraction of that space that is filled. Occupation numbers can be larger than unity for photons, which are Bose-Einstein particles.

   For reference, the Planck function is given by

$$B_\nu = I_\nu = \frac{F_\nu}{\pi} = \frac{2h\nu^3}{c^2}\frac{1}{\exp\left(h\nu/kT\right)-1} \quad [\mathrm{erg\ cm^{-2}\ s^{-1}\ sr^{-1}\ Hz^{-1}}] \qquad (5.28)$$

where F_ν is the single-hemisphere emittance from an opaque surface. The photon occupation number of a blackbody is then

$$\eta_\nu = \frac{1}{\exp\left(h\nu/kT\right)-1} \qquad (5.29)$$

The dimensionless occupation number for any continuum with a mean intensity J_ν (erg cm^{-2} s^{-1} Hz^{-1} sr^{-1}) at a frequency ν is defined as

$$\eta_\nu \equiv J_\nu/\left(2h\nu^3/c^2\right) \equiv \left[\exp\left(h\nu/kT_{ex}\right)-1\right]^{-1}. \qquad (5.30)$$

Here T_{ex} is the excitation temperature of the continuum at the frequency.

5.6.2   Pumping rates

Continuum fluorescence is treated as in Ferland and Rees (1988) and Ferland (1992). The rate of induced radiative excitation by continuum photons (continuum pumping) is given by

$$r_{l,u} = n_l B_{l,u} J_{l,u} = n_l A_{u,l}\frac{J_{l,u}}{2h\nu^3/c^2}\frac{g_u}{g_l} = n_l A_{u,l}\eta_\nu\frac{g_u}{g_l} \quad [\mathrm{cm^{-3}\ s^{-1}}] \qquad (5.31)$$

where η_ν is the dimensionless continuum occupation number at the line energy. The rate of induced radiative de-excitation is related by detailed balance and is given by

$$r_{u,l} = r_{l,u}\frac{g_l}{g_u} \quad [\mathrm{cm^{-3}\ s^{-1}}]. \qquad (5.32)$$

The occupation number has the advantage that the Einstein B's do not enter any rate equations. All radiative rates can be expressed in terms of an A and ν.

5.6.3   Optical depth effects

The line becomes self-shielding when the optical depth is greater than unity. The line optical depth between the current position and the illuminated face of the slab is used to evaluate the inward-looking escape probability, the probability that a line photon will travel this distance in a single scattering. Line optical depths do not directly affect η_c, only continuous opacities do. The final form of the continuum pumping rate is

$$r_{l,u} = n_l A_{u,l}\eta_\nu\frac{g_u}{g_l}\gamma_{l,u}(\tau) \quad [\mathrm{cm^{-3}\ s^{-1}}] \qquad (5.33)$$
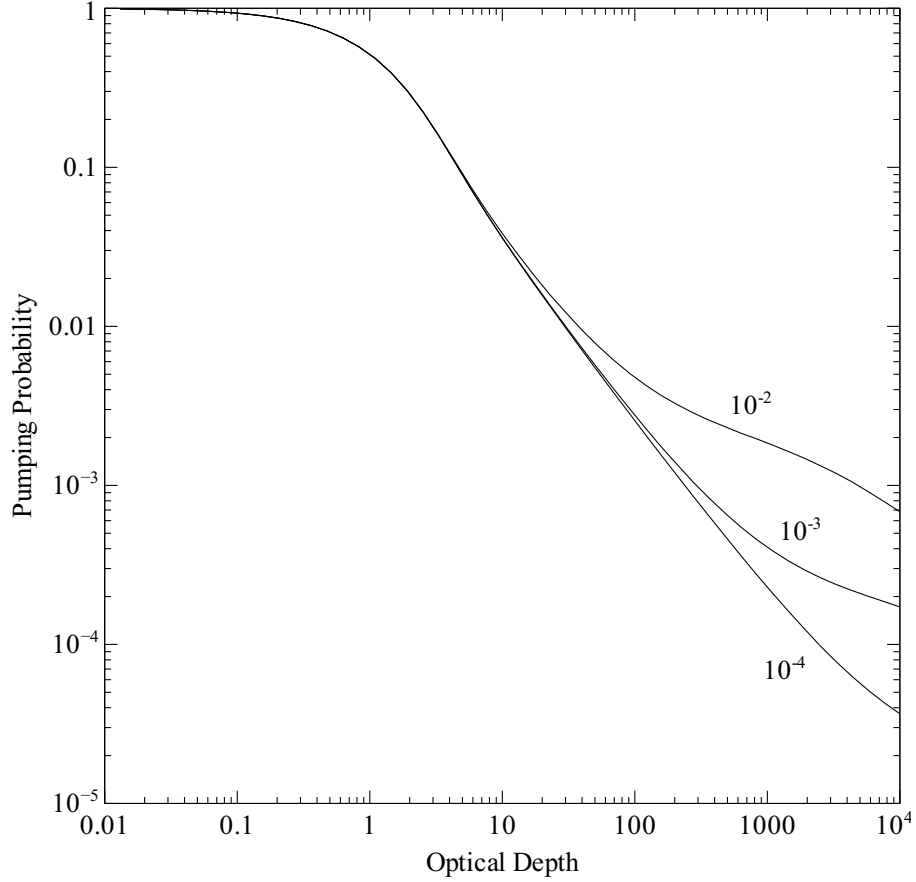
Figure 5.1: This figure shows the probability that a photon will penetrate to the line center optical depth shown on the x-axis, and then be absorbed by the line. The curves are for various values of the damping constant a (the ratio of damping width to Doppler width), as indicated on the figure.

where $\gamma_{l,u}$ is the probability that continuum photons penetrate an optical depth τ_o and are then absorbed by an atom:

$$\gamma_{l,u} = \int_0^\infty \varphi_v \, \exp\left(-\tau_o \varphi_v\right) dv \bigg/ \int_0^\infty \varphi_v \, dv. \tag{5.34}$$

where φ_v is the Voigt function. Figure 5.1, taken from Ferland (1992), shows $\gamma_{l,u}$ for a wide variety of values of the damping constant a.

The code works in terms of the flux of photons per energy mesh point. The transmitted continuum has a flux of photons φ_v (photons cm^{-2} s^{-1} Ryd^{-1}). The photon occupation number of the attenuated continuum is given by equation 5.29 above, here written as

$$\eta_v = \varphi_v \, \frac{c^2}{8\pi \, v_1^3 \, v_{Ryd}^2} \tag{5.35}$$

where v_{Ryd} is the frequency in Rydbergs, v_1 is the frequency of 1 Rydberg, and the other symbols have their usual meaning. Continuum pumping is included among the general line excitation processes for all lines considered by the code.

5.7   Kirchhoff's Law

Kirchhoff's law is the statement that, in thermodynamic equilibrium, the energy emitted is equal to the energy absorbed. If the emission and absorption coefficients are j_ν and κ_ν then

$$j_\nu = \kappa_\nu B_\nu(T) \tag{5.36}$$

where $B_\nu(T)$ is Planck's function.

5.8   The line source function and mean intensity

The source function for a line is defined as

$$S_l(T_{exc}) \equiv B_l(T_{exc}) \equiv \frac{j_l}{\kappa_l} = \frac{A_{u,l} n_u}{B_{l,u}(n_l - n_u g_l/g_u)} \quad [\text{erg Hz}^{-1}\ \text{sr}^{-1}\ \text{s}^{-1}]. \tag{5.37}$$

where T_{exc} is the line excitation temperature

$$\frac{n_u/g_u}{n_l/g_l} = \exp\left[-h\nu/kT_{exc}\right] \tag{5.38}$$

$B_l(T_{exc})$ is the Planck function at the line excitation temperature and the line emission and absorption coefficients jl and kl enter through Kirchhoff's law. Combining with the definitions of the Einstein relations we find the relation

$$S_l(T_{exc}) = \frac{2h\nu^3}{c^2}\frac{n_u/g_u}{(n_l/g_l - n_u/g_u)} \quad [\text{erg Hz}^{-1}\ \text{sr}^{-1}\ \text{s}^{-1}]. \tag{5.39}$$

The radiation field within the line is given by the mean intensity $\bar{J}$. $\bar{J}$ and S_l are related by the net radiative bracket, which we approximate as the escape probability P_{esc}:

$$P_{esc} \equiv 1 - \bar{J}/S_l. \tag{5.40}$$

The mean intensity is then give by

$$\bar{J} = S_l(1 - P_{esc}). \tag{5.41}$$

and the line center photon occupation number is

$$\eta_l = \frac{n_u/g_u}{(n_l/g_l - n_u/g_u)}(1 - P_{esc}). \tag{5.42}$$

5.9   Level populations in radiative equilibrium limit

For a two-level system where collisions can be neglected, in the optically thin limit, the balance equation relating the populations of a two-level system is given by

$$\frac{dn_u}{dt} = n_l B_{lu}J - n_u(A_{ul} + B_{ul}J) \quad [\text{cm}^{-3}\ \text{s}^{-1}]. \tag{5.43}$$

In the time-steady limit, where the time derivative is zero, the balance can be rewritten in terms of the transition probabilities as

$$n_u \left(A_{ul} + A_{ul} \eta \right) = n_u A_{ul} \left(1 + \eta \right) = n_l A_{ul} \eta \frac{g_u}{g_l} \quad [\text{cm}^{-3} \text{ s}^{-1}]. \tag{5.44}$$

In the limit where η is small every photoexcitation is followed by spontaneous decay, while in the limit where η is large the level populations are given by the ratio of statistical weights, i.e., T_{exc} is infinite.

Chapter 6

LINE DETAILS

6.1 Overview

The effects of optical depths, continuum pumping, collisions, and destruction by background opacity, are computed for *all* permitted and intercombination lines. The cooling is usually distributed among many lines in high-density models, and these lines are usually optically thick. This section describes the methods and data structures used within the code to accomplish this.

6.2 Line Boltzmann factors

The Boltzmann factor $h\nu/kT$ for a line with a known wavelength or energy is given by Table 6.1. The table lists the ratio $h\nu/k$ for various units of the line energy. Vacuum, not air, wavelengths, must be used for all quantities involving wavelengths.

6.3 Air vs vacuum wavelengths

The convention across physics and astronomy is to give line wavelengths in vacuum for $\lambda \leq 2000$ Å and in air for $\lambda > 2000$ Å. There is no choice—if you observe visible light with HST the wavelengths must be quoted in air.

Air wavelengths are smaller than vacuum wavelengths because the wavefronts are crushed as then enter the denser medium with its higher index of refraction. The frequency is unchanged.

Table 6.1: Line Boltzmann Factors	
Line Energy Units	$h\nu/k$ (K)
Angstroms	$1.43877(+8)/\lambda(\text{Å})$
microns	$1.43877(+4)/\lambda(\mu)$
wavenumbers	$1.43877 \times \sigma$
Rydbergs	$1.5788866(+5) \times E$

6.4   The line escape probability functions

6.4.1   Escape probability formalism vs exact radiative transfer

Radiative transport effects are approximated with the escape probability formalism (EPF). This includes line pumping by the incident continuum, photon destruction by collisional deactivation or by continuous opacities, and line overlap in special cases. This section describes how the escape probability is related to the net radiative bracket, the formally correct term in the transfer equation.

The full balance equation for radiative losses and gains for the upper level of a two level atom is given by

$$n_u A_{ul} + n_u B_{ul} \bar{J} - n_l B_{lu} \bar{J} \equiv n_u A_{ul} \rho_{ul} \approx n_u A_{ul} P_{ul} \tag{6.1}$$

where A and B are the Einstein coefficients, $\bar{J}$ is the mean intensity averaged over the line, and ρ_{ul} is the net radiative bracket, defined as

$$\rho_{ul} \equiv 1 - \bar{J}/S \tag{6.2}$$

where S is the line source function. The essence of the EPF is to replace ρ_{ul} with the escape probability P_{ul} on the argument that the difference between J and S is due to photons leaking away from the region. (Elitzur et al., 1983; Sec 2.6) shows that this is exact if S is constant across the line-forming region. In the code ρ_{ul} is replaced with P_{ul}.

6.4.2   Redistribution functions

At low densities, line scattering for a two-level atom is coherent in the atom's reference frame and the line profile function is described by the incomplete redistribution function. At high densities the Stark effect can broaden the line. When the radiation density is high, scattering within excited states can inhibit the broadening of resonance lines such as Lβ (line interlocking), destroying the coherence of the scattering process. In these cases complete redistribution in a Doppler core more closely describes the scattering process.

CLOUDY uses several escape probability functions to take these processes into account. Strong resonance lines are treated with partial redistribution with a Voigt profile. Subordinate lines are treated with complete redistribution in a Doppler core.

6.4.3   Incomplete redistribution

Incomplete redistribution is assumed for resonance transitions such as C IV $\lambda 1549$ and the Lα transitions of hydrogen and helium. Two studies of line formation using this approximation are those of Bonilha et al. (1979) and Hummer and Kunasz (1980). Both studies suggest escape probabilities of the form

$$P_l(\tau) = \{1 + b(\tau)\,\tau\}^{-1} \tag{6.3}$$

but there is substantial disagreement in the form and value of the factor $b(\tau)$, sometimes by more than a factor of 2. (This is after due allowance for the different definitions of line opacities in the two papers.) CLOUDY uses the Hummer and Kunasz (1980) results for H I, He I, and He II Lα and strong resonance lines such as C IV $\lambda 1549$. Their tabulated values were fitted by interpolation.

6.4.4 Damping constant

The damping constant a is given by

$$a = \frac{\Gamma}{4\pi\,\Delta\nu_D} = \frac{\lambda_{cm}\sum A}{4\pi\,u_{Dop}} = \frac{\lambda_{cm}7.958\times10^{-2}\sum A}{u_{Dop}} = \frac{\lambda_{\mu m}7.958\times10^{-6}\sum A}{u_{Dop}} \tag{6.4}$$

where Γ is the inverse lifetime of the level (the sum of the A's from the upper level), $\Delta\nu_D$ is the Doppler width in frequency units (Mihalas, 1978), λ_{cm} and $\lambda_{\mu m}$ are the wavelengths in cm and microns respectively, and u_{Dop} is the Doppler width in cm s^{-1}. The ratio $\Gamma\lambda/4\pi$ is stored in the line structures and the a's are evaluated using this ratio and the current Doppler width.

6.4.5 Background opacity and Destruction probability

The ratio of continuous to total opacity is X_c parameterized as

$$X_c = \frac{\sum \kappa_c\, n_c}{\kappa_l n_l + \sum \kappa_c\, n_c} \tag{6.5}$$

where the κ_l's are the line center absorption opacities and the n's the number of absorbers.

6.4.6 Complete redistribution

Lines arising from excited states (hydrogen Balmer, Paschen, etc.) and Lyman lines with $n_u > 2$ are treated assuming complete redistribution in a Doppler core (i.e., the damping constant a is assumed to be zero). This assumption can be changed with the **atom redistribution** command. If the total optical depth of the slab is T, then the escape probability at a depth τ from the illuminated face is given by;

$$P_{u,l}(\tau, T, X_c) = [1 - X_c F(X_c)]\frac{1}{2}[K_2(\tau, X_c) + K_2(T - \tau, X_c)] \quad, \tag{6.6}$$

and the destruction probability is

$$D_{u,l}(X_c) = X_c F(X_c). \tag{6.7}$$

The function is

$$F(X_c) = \int_{-\infty}^{\infty} \frac{\varphi(x)}{X_c + \varphi(x)}\, dx, \tag{6.8}$$

where in these expressions (and in this part of the code) the *mean opacity is used*, and $\varphi(x) \approx \pi^{1/2}\exp(-x^2)$ is the Voigt function. $F(X_c)$ is interpolated from the tables presented by Hummer (1968). The function

$$K_2(\tau, X_c) \equiv \frac{1}{1 - X_c F(X_c)}\int_{-\infty}^{\infty} \frac{\varphi^2(x)}{X_c + \varphi(x)}E_2\left[(X_c + \varphi(x))\,\tau\right]\, d\tau \tag{6.9}$$

is evaluated numerically.

The complete redistribution escape probabilities are corrected for finite damping parameters by including a separate wing escape probability. The resulting total escape probabilities have been

tested against numerical integrations of the exact formulae in Avrett and Loeser (1966); Hummer and Rybicki (1982) – note the different definitions of K_2 in these papers – verified by comparison by the tabulations and fits in these papers. The overall fit to the escape probability is in error by at most 25% in the range $a = 10^{-3}$–10^3, with the largest error at $\tau \simeq 1$, and correct asymptotic behaviour in the low τ and high τ limits. This level of accuracy is thought to be sufficient, given the other approximations inherent in the escape probability method.

6.4.7   Masing lines

A line mases when its optical depth is negative. The escape probability is (Elitzur, 1992; p 32)

$$P_{u,l} = \frac{1 - \exp(-\tau)}{\tau}. \tag{6.10}$$

The code will generate a comment if strong maser action occurs for any transition.

6.4.8   Stark broadening

Distant collisions with charged particles broaden the upper levels of lines, and in the limit of very high densities this will make the scattering process completely non-coherent even for Lα (i.e., complete redistribution obtains). Cloudy closely follows the treatment of Puetter (1981) in treating Stark broadening.

6.4.9   Net escape probability

A total escape probability $P_{l,tot}$, given by

$$P_{u,l} = \min\left(P_{inc} + P_{Stark}\,,\, P_{com}\right), \tag{6.11}$$

is defined for transitions described by incomplete redistribution. The escape probabilities are those for incomplete, Stark, and complete redistribution respectively. The total effective escape probability is not allowed to exceed the complete redistribution value for $\tau > a^{-1}$.

   If τ is the optical depth in the direction towards the source of ionizing radiation and T is the total optical depth computed in a previous iteration, then the escape probability entering the balance equations is

$$P_{u,l}(\tau, T) = \left\{ P_{u,l}(\tau) + P_{u,l}(T - \tau) \right\}/2. \tag{6.12}$$

In general the total optical depth T is only known after the first iteration, so more than one iteration must be performed when radiative transfer is important.

6.5   Optical depths and the geometry

The terms open and closed geometry are defined in a section in Part I. The treatment of transfer in these two limits is described here.

6.5.1 Open geometry

This is the default. During the first iteration the line optical depth is defined using only optical depths accumulated in the inward direction. This optical depth is initialized to a very small number at the start of the calculation. At the end of the first iteration the total optical depth is set to the optical depth accumulated in the inward direction. At the end of subsequent iterations the total optical depth is defined as a mean of the new and old inward optical depths.

6.5.2 Closed geometry overview

Continuum photons are assumed to interact with gas fully covering the continuum source. At the end of the first iteration the total continuum optical depths are set equal to twice the computed optical depths, and the inner optical depths reset to the computed optical depths. The same recipe is followed on subsequent iterations, except that means of old and newly computed optical depths are used.

Closed expanding geometry This is the default if the **sphere** command is entered. In this case it is assumed that line photons do not interact with lines on the "other" side of the expanding spherical nebula. The treatment of line optical depths is entirely analogous to that described for an open geometry, since the presence of the distant material has no effect on line transfer.

Closed static geometry This is assumed if the **sphere static** command is entered. In this case line photons from all parts of the spherical shell do interact. As a result the optical depth scale is poorly defined on the first iteration, and more than one iteration is required. On second and later iterations the total line optical depth is set to twice the optical depth of the computed structure, and the optical depth at the illuminated face of the shell is set to half of this. The optical depth scale is only reliably defined after at least a second iteration.

6.5.3 Wind

The model is a large velocity gradient ($v \propto R$ Sobolev approximation) wind. This is described further in Part 2 of this document.

6.6 Collision strengths

I have tried to follow the Opacity Project notation throughout this document (Lanzafame et al., 1993). The energy-specific collision strength Ω_{lu} for a transition between upper and lower levels u and l is related to the excitation cross section Q_{lu} by

$$Q_{lu} = \frac{\pi \Omega_{lu}}{g l k_{lu}^2} \quad [\text{cm}^2] \tag{6.13}$$

where k_{lu}^2 is the wavenumber of the collision energy. If the collisions are with thermal electrons having a Maxwellian velocity distribution $f(u)$ and velocity u then the rate coefficient q_{lu} is given by

$$q_{lu} = \int_0^\infty f(u) u Q_{lu} \, du = \frac{2\pi^{1/2}\hbar^2}{g_l m_e} a_o \left(\frac{R_\infty}{kT}\right) \Upsilon_{lu} \exp\left(-\frac{E_{lu}}{kT}\right) \sqrt{\frac{2kT}{m_e}} \quad [\text{cm}^3 \text{s}^{-1}]. \tag{6.14}$$

E_{ul} is the transition energy in Rydbergs, a_o is the Bohr radius,

$$a_o = \frac{\hbar^2}{m_e q_e^2} = 0.529177249 \times 10^{-8} \text{ cm} \tag{6.15}$$

and R_∞ is the Rydberg energy. Then the thermally-averaged collision strength is given by

$$\Upsilon_{lu} = \int_0^\infty \Omega_{lu} \exp\left(-\frac{\varepsilon}{kT}\right) d\left(\frac{\varepsilon}{kT}\right). \tag{6.16}$$

The rate coefficient for collisional de-excitation is then given by

$$q_{ul} = \frac{\Upsilon}{g_u \sqrt{T_e}} \left(\frac{2\pi}{k}\right)^{1/2} \frac{\hbar^2}{m_e^{3/2}} = \frac{\Upsilon 8.6291 \times 10^{-6}}{g_u \sqrt{T_e}} \quad [\text{cm}^3 \text{s}^{-1}]. \tag{6.17}$$

The rate coefficient for excitation follows from detailed balance:

$$q_{lu} = q_{ul} \frac{g_u}{g_l} \exp(-\chi) = \frac{\Upsilon 8.6291 \times 10^{-6}}{g_l \sqrt{T_e}} \exp(-\chi) \quad [\text{cm}^3 \text{s}^{-1}]. \tag{6.18}$$

6.7   Born approximation

The Born approximation is valid for energies much larger than the excitation energy of the transition. The energy specific collision strength is given by Bethe (1930) as

$$\Omega_{lu} \approx \frac{4 g_l f_{lu}}{E_{lu}} \ln\left(\frac{4\varepsilon}{E_{lu}}\right) \tag{6.19}$$

where $f_{l,u}$ is the absorption oscillator strength of the permitted transition.

6.8   The g-bar approximation

The g-bar or van Regemorter (1962) approximation relates the collision strength to the transition probability A_{ul} and wavelength λ (in μm). Here, the collision strength for the downward transition Υ_{ul} is approximately given by

$$
\begin{aligned}
\Upsilon_{u,l} &= \frac{2\pi}{\sqrt{3}} \frac{m^2 e^2}{h^3} \lambda_{\mu m}^3 10^{-12} g_u A_{u,l} \bar{g} \\
&\approx 2.388 \times 10^{-6} \lambda_{\mu m}^3 g_u A_{u,l} \bar{g} \\
&\approx 159 \lambda_{\mu m} g_l f_{abs} \bar{g}
\end{aligned}
\tag{6.20}
$$

where g_u and g_l are the statistical weights of the upper and lower levels and f_{abs} is the absorption oscillator strength. For energies of interest in astrophysical plasmas, where $kT < h\nu$, $\bar{g}$ is approximately given by

$$\bar{g} \approx \begin{cases} 0.2; & \text{positive ions} \\ (kT/h\nu)/10; & \text{neutrals} \end{cases} \tag{6.21}$$

(van Regemorter 1962). These approximations are generally accurate to better than 1 dex. Gaetz and Salpeter (1983) give improved forms of the approximation.

6.9 The critical density

The critical density is defined as the density at which the radiative de-excitation rate $A_{ul}P_{ul}$ (where A is the transition probability and P is the escape probability) equals the collisional de-excitation rate $q_{ul}n_e$. Setting

$$A_{ul}P_{ul} = C_{ul} = q_{ul}n_e = \Upsilon\frac{8.629 \times 10^{-6}}{g_u\sqrt{T_e}}n_e \quad [\text{s}^{-1}] \tag{6.22}$$

where Υ is the thermally averaged collision strength, the critical density is given by

$$n_{crit} \sim \frac{A_{ul}P_{ul}g_u\sqrt{T_e}}{\Upsilon 8.629 \times 10^{-6}} \quad [\text{cm}^{-3}]. \tag{6.23}$$

For an optically allowed transition, in which the g-bar approximation may apply, this density is approximately given by

$$n_{crit} = \frac{4.8 \times 10^{10}\sqrt{T_e}}{\lambda_{\mu m}^3\bar{g}} \quad [\text{cm}^{-3}]. \tag{6.24}$$

6.10 Line thermalization length

Line radiative transfer will affect the thermal equilibrium of the gas when the collision time scale approaches an effective lifetime $\tau \sim A_{ul}/n_{scat})^{-1}$, where A_{ul} is the transition probability and n_{scat} is the number of scatterings a line photon undergoes before escape. For permitted metal lines (which often have optical depths $\sim 10^4 - 10^6$) line thermalization becomes important at densities $n_e > 10^{15}/\tau 10^{10}\text{cm}^{-3}$. These effects are important for hydrogen at considerably lower densities due to its greater abundance. Additionally, continuum transfer affects the ionization and thermal equilibrium of the gas at all densities.

6.11 Averaging levels into terms

6.11.1 Collision strengths

Often cases are encountered in which a multiplet consisting of many lines can be treated as the equivalent two-level atom with a single transition. In these cases it is necessary to define "effective" collision strengths and transition probabilities. If the collision strength from an individual level i is Υ_i, and the statistical weights of the level and term are g_i and g_{tot} respectively, then the effective collision strength Υ_{eff} is related to Υ_i by a simple argument. The collision rate q_i is proportional to the ratio

$$n_iq_i \propto n_i\frac{\Upsilon_i}{g_i} \quad [\text{s}^{-1}] \tag{6.25}$$

so that

$$n_{tot}q_{tot} = \sum_i n_iq_i \propto \sum_i n_i\frac{\Upsilon_i}{g_i} \quad [\text{s}^{-1}]. \tag{6.26}$$

In many cases it is valid to assume that the levels within the term are populated according to their statistical weight, viz.,

$$n_i = n_{tot} \frac{g_i}{g_{tot}} \quad [\text{cm}^{-3}]. \tag{6.27}$$

Then, the effective collision strength Υ_{tot} is operationally defined by the relations

$$n_{tot} \frac{\Upsilon_{tot}}{g_{tot}} = \sum_i n_i \frac{\Upsilon_i}{g_i} = n_{tot} \sum_i \frac{g_i}{g_{tot}} \frac{\Upsilon_i}{g_i} = n_{tot} \frac{\sum_i \Upsilon_i}{g_{tot}}. \tag{6.28}$$

So, the effective collision strength of the entire multiplet is

$$\Upsilon_{tot} = \sum_i \Upsilon_i. \tag{6.29}$$

6.11.2   Transition probabilities

Under similar circumstances an effective transition probability A_{eff} may be defined as

$$n_{tot} A_{tot} = \sum_i n_i A_i = n_{tot} \sum_i \frac{g_i}{g_{tot}} A_i \tag{6.30}$$

so that the effective transition probability is

$$A_{tot} = \sum_i \frac{g_i}{g_{tot}} A_i. \tag{6.31}$$

So collision strengths are added, and transition probabilities averaged.

6.12   Level populations with collisions

Both escape and destruction probabilities enter in the calculation of a level population and line emissivity. The escape probability $P_{u,l}$ is the probability that a line photon will escape in a single scattering (Elitzur et al., 1983; Elitzur, 1984). The destruction probability $D_{u,l}$ is the probability that a line photon will be destroyed in a single scattering.

The line de-excitation rate is given by

$$\left(\frac{dn_u}{dt}\right)_{rad} = n_u A_{u,l} \left(P_{u,l} + D_{u,l}\right) - n_l A_{u,l} \eta \gamma_{u,l} + n_u C_{ul} - n_l C_{lu} \quad [\text{cm}^{-3}\text{s}^{-1}] \tag{6.32}$$

where η is the photon occupation number of the attenuated external radiation field, C is the collision rate (s^{-1}), and $\gamma_{u,l}$ is the fluorescence probability.

The net emission from a transition between the level n to a lower level l and escaping to the surface is then

$$4\pi j(n,l) = n_n A_{n,l} \, h\nu_{n,l} \, P_{u,l}(\tau_{n,l}) \, f(r) \quad [\text{erg cm}^{-3}\text{s}^{-1}] \tag{6.33}$$

where $f(r)$ is the filling factor. The total emission from the gas is then

$$e(n,l) = \int_V n_n A_{n,l} \, h\nu_{n,l} \, P_{u,l}(\tau_{n,l}) \, f(r) \, dV \quad [\text{erg cm}^{-2}\text{s}^{-1}\text{or erg s}^{-1}] \tag{6.34}$$

depending on whether the intensity or luminosity case is chosen. The local cooling rate (erg cm^{-3} s^{-1}) due to the line is related to the level populations by

$$\Lambda_{u,l} = \left(n_l C_{l,u} - n_u C_{u,l} \right) \, f(r) \, h\nu \quad [\text{erg cm}^{-3}\text{s}^{-1}] \tag{6.35}$$

and the local flux (cm^{-2} s^{-1}) of "on-the-spot" (OTS) photons caused by line loss (used to compute heating or photoionization rates for the sources of the background opacity) is

$$\varphi_{OTS} = \frac{n_u A_{u,l} D_{u,l}(X_c)}{\sum \kappa_c \, n(c)}. \tag{6.36}$$

The ratio of inward to total line intensity is then given by

$$\frac{4\pi \, j(in)}{4\pi \, j(total)} = \frac{P_{u,l}(\tau)}{\left[P_{u,l}(\tau) + P_{u,l}(T - \tau) \right]}. \tag{6.37}$$

Chapter 7

CLOUDY AS A STANDALONE PROGRAM

CLOUDY can be used to run a single model, to create large grids of calculations, or to compute a number of simulations while varying one or more parameters to match an observed spectrum. This Chapter describes how to use CLOUDY as a self-contained program to read in the parameters for the simulation and compute the result. The next Chapter discusses the case in which the code is called as a subroutine of another larger code.

7.1 Running a single model with a shell script

The code reads from an input file and can create a large number of output files. The latter include both the main output (described in Chapter 1, *OUTPUT*) and the ancillary "save" files (described in the Chapter 15 of Hazy 1, *CONTROLLING OUTPUT*).

It is a good idea to follow a naming convention for these files. The convention I use is the style "`basename.type`" where `basename` explains the astrophysical context (for instance, "quasar" or "IGM") and `type` gives the type of information in the file. For instance, a model of a planetary nebula may have the base name "`pn_halo`", the input script might be `pn_halo.in`, the output would be in `pn_halo.out`, and the file created by the **save overview** command might be `pn_halo.ovr`. Then all of these files could be located with a simple "**ls pn**.*" and all overview files with a "**ls *.ovr**" on a Linux system.

The `pn.in` file contains the input commands that tell the program what to do. A typical example might be the following:

```
// log of the hydrogen density (cm^-3)
hden 4
// log of the inner radius (cm)
radius 17
// black body temperature and total luminosity
black body 1e5 K, luminosity 38
save overview "pn.ovr"
```

CLOUDY stops reading the input steam when it reaches either an empty line or the end of file. Nothing special is needed at the end of the input file.

I have a shell script named `run` which is in my "`bin`" directory, which I include on my path. The shell script `run` consists of the following on Linux:

```
cloudy.exe < $1.in > $1.out
```

Under Windows it would have the name `run.bat` and would contain the following

```
cloudy.exe < %1.in > %1.out
```

If `run` is executed by typing

```
run pn
```

it would read the input stream in `pn.in` and create an output file called `pn.out`.

7.2   Running a single model from the command line

The code also has two command-line options that will accomplish the same thing as the shell script described in the previous section. If you create an executable called `cloudy.exe`, then the command

```
cloudy.exe model.in
```

will read input from `model.in`, write output to `model.out`, and add the prefix `model` to all the save files. This can also be written

```
cloudy.exe -p model
```

Alternatively, you can use the command

```
cloudy.exe -r model
```

which does the same thing, except that it will *not* add the prefix `model` to all the save files (mnemonic: `-p` will Prefix Save file names, while `-r` will only Redirect input and output).

By typing the command

```
cloudy.exe -h
```

you can a list of all the command line flags that CLOUDY supports. There are two additional flags, **-a** and **-g**, that are used for debugging the code and internally in grid runs. They should never be used in normal situations.

7.3   Running grids of simulations or optimizing a simulation

The greatest insight is gained by creating grids of simulations which vary one or more of the input parameters to show how various predictions change, or to optimize the agreement between the predicted and observed values. One example, the predicted C IV *lambda*1549 equivalent width as a function of the flux of ionizing photons and cloud density, is shown in Figure 8.1. See Baldwin et al. (1995c) for more details.

Such grids of simulations can be made parallel on distributed memory parallel machines because each point in the grid is independent of the other points.

7.3.1 Grids of simulations

The **grid** command is described in Chapter 17 *THE GRID COMMAND* of Hazy 1.

7.3.2 Optimizing a simulation

The **optimize** command is described in Chapter 16 *THE OPTIMIZE COMMAND* of Hazy 1.

7.4 Parallel processing with MPI

Peter van Hoof created a wrapper for the **grid** and **optimize** commands. CLOUDY will automatically run these commands using all available processors if the code is compiled with an MPI aware compiler and run on an appropriate computer.

Chapter 8

CLOUDY AS A SUBROUTINE

8.1 Overview

CLOUDY is designed to be used as a subroutine of other, much larger, codes. When used this way a series of subroutine calls, described next, are used to initialize the code, specify the initial conditions, do the simulation, and finally examine the predictions.

A common strategy is to call the code to compute line intensities for a large matrix of parameters. The results of one such calculation is shown in Figure 8.1 (Baldwin et al., 1995b; Ferland, 2003). Such grids can be computed in a few dozen hours on modern workstations, and offer far greater insight to physical effects of changing model parameters than does a single model.

Much of this can be done without writing a program, by using the **grid** command. This command was introduced by Ryan Porter in C07.02, and makes it possible to create large grids of models like those shown in Figure 8.1 with commands. The **grid** command is described in Part 1 of this document. There are several **save** output commands that are designed to save information from grid calculations.

8.1.1 ♺ Host languages

CLOUDY is written in C++, and it assumed for most of this section that CLOUDY will be called from an external C or C++ program.

However, there is experimental support for a number of other languages.

You can call a C++ program like CLOUDY from a Fortran program by using the *cfortran.h* header file described at http://www-zeus.desy.de/~burow/cfortran/. I have never tried this. Good luck.

In the subdirectory sys_gcc_shared, an additional set of Makefile targets are set up to use the subroutine API described below for Perl and Python scripts, using the SWIG interface generator. In this directory, there is also an experimental direct interface to the Lua scripting language.
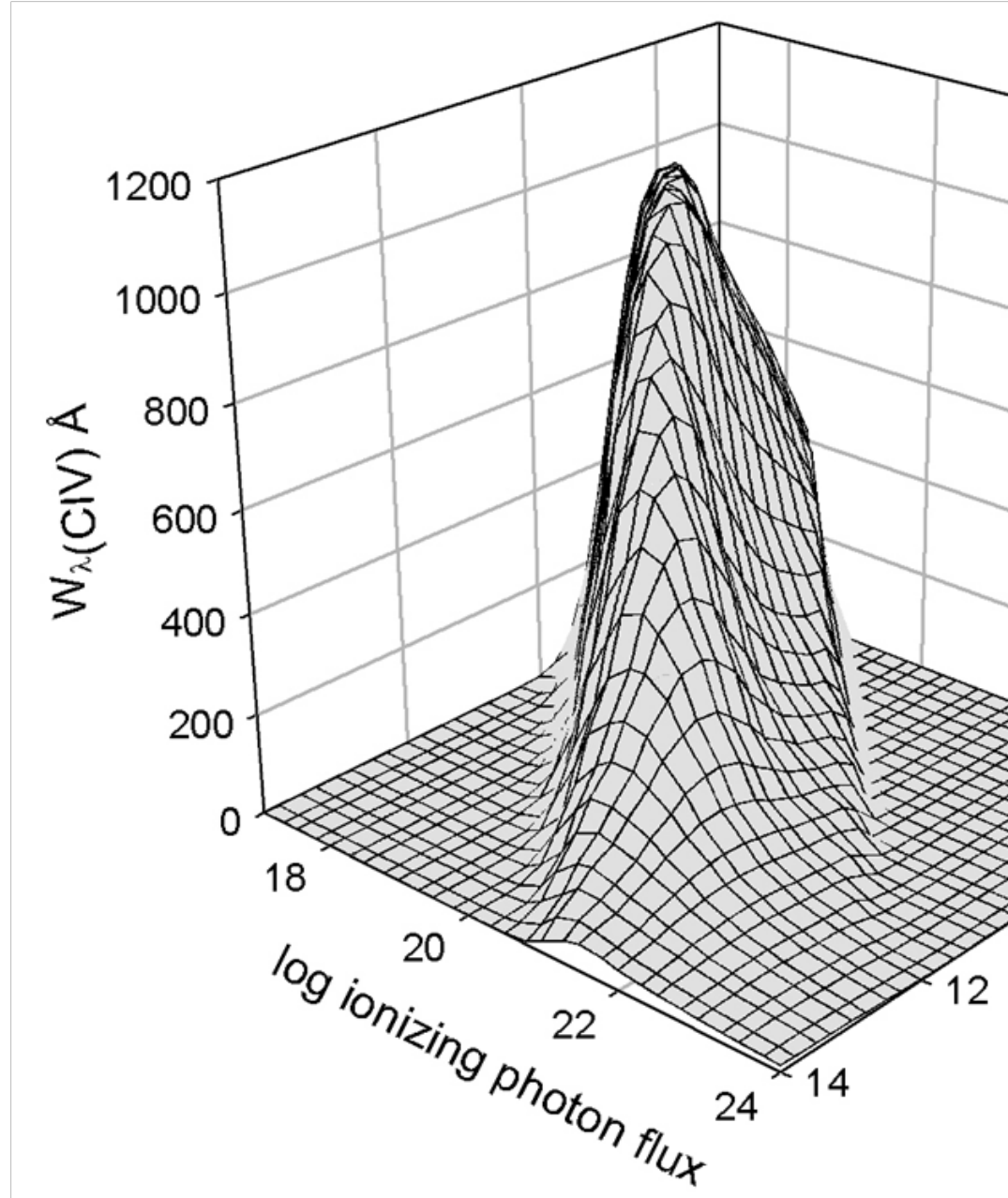
Figure 8.1: The results of a large grid of quasar emission-line cloud calculations are shown. The x-y plane shows the logs of the hydrogen density ($cm^{3)}$ and flux of ionizing photons (cm^{-2} s^{-1}). The z axis is the predicted line equivalent width.

8.1.2 Creating a new main program

In C++ there must be exactly one main program and it must be called *main*. This routine is within the file `maincl.cpp` in the source downloaded from the web. You need to replace the existing CLOUDY main program with one that you write. The file `maincl.cpp` that is included in the distribution must be deleted so that the program you write will be loaded instead. The remaining routines are then compiled with a command like the following:

```
g++ -O3 -c *.cpp
```

which will create a large number of object files. The new main program will then need to be linked with these object files with a command something like

```
g++ -O3 -o mycloudy.exe newmain.cpp *.o -lm
```

Alternatively, if you are using the Makefile build process and have already built CLOUDY, you can use the library file `libcloudy.a` which contains all the object files with the command

```
g++ -O3 -o mycloudy.exe newmain.cpp -L. -lcloudy -lm
```

The following subsections outline how to write code for this new main program.

8.1.3 The cddefines.h and cddrive.h header files

The file `cddrive.h` contains definitions of all public routines, the routines that a user would call to drive CLOUDY. That file is the definitive reference for the material contained in this section and is more up to date than this document. Comments within that file explain all routines and their parameters.

The header `cddefines.h` should come before `cddrive.h` since it includes many definitions and includes the standard C++ header files that are needed to drive the code. The first two header files in the new main routine should be the following:

```
#include "cddefines.h"
#include "cddrive.h"
```

8.1.4 The template for main programs

C++ exceptions are used by CLOUDY. The main program must catch these exceptions. If it does not then the code will crash with an unhandled exception and output files will not be complete. This requires that the main program start with a C++ "try" block and that exception handlers be included within the main program. The adventurous user may enjoy creating their own try block/exception handlers.

A sample template, in the file `template.cpp`, includes the needed code and can be used to create your own programs. It is included in the `programs` directory below the `tsuite` directory in the program download. That version includes a simple call to run the code's smoke test. You would replace the existing code with your own.

8.1.5   A note on return conditions

Some of the routines return a value to indicate success or failure. I try to follow the C and Unix conventions to indicate success with zero (or *false*) and trouble with a non-zero return (or *true*). This rule is not always followed (it is not followed by the important routine *cdLine*), however, and `cddrive.h` should be consulted to make sure the return conditions are understood.

8.2   Initializing the code

Many variables must be initialized at the beginning of the calculation. Calling routine *cdInit* does this.

```
cdInit();
```

Routine *cdInit* must be called every time a new calculation is to be performed, *before* calling any of the following subroutines, but after the results of any previous calculations have been read. (The results of any previous calculations are lost when *cdInit* is called.)

8.3   Handling input and output

8.3.1   cdTalk—produce output?

CLOUDY normally speaks what's on its mind. This would generate too much output in a large grid. It does have a quiet mode in which nothing at all is printed. This quiet mode is set by the logical argument to subroutine *cdTalk*.

```
#include "cddefines.h"
#include "cddrive.h"
cdInit();
/*set no output at all*/
cdTalk( false )
/*have the code produce the normal printout*/
cdTalk( true )
```

The default is for CLOUDY to produce output. *cdTalk* does not have to be called if this is what you want. It needs to be called with the logical variable *false* if the quiet mode is desired.

8.3.2   cdOutput—sending output to a file

CLOUDY normally writes its standard output on the system's `stdout`. This can be changed to another file by calling the routine *cdOutput*, which has two arguments. The first is the name of the file, and the second the mode with which the file should be opened (this is identical to the second parameter of the fopen() routine). Both parameters have default values. The default mode for opening the file is "w", while the default filename is the empty string. Supplying an empty string as the file name means that the output will be directed back to `stdout` again. The output file will be automatically closed on exit of the program. If you call *cdOutput* multiple times with different file names, the file from the previous call will also be closed automatically.

```
#include "cddefines.h"
#include "cddrive.h"

/* open the file output.txt for writing with mode "w" */
/* this will overwrite an existing file */
cdOutput("output.txt");

/* if you want to append to a file, use this instead */
cdOutput("output.txt","a");

/* this will close output.txt and open output2.txt with mode "w" */
cdOutput("output2.txt");

/* if you now want to redirect to stdout again, use this */
/* this will close output2.txt */
cdOutput();
```

8.3.3 cdInput—reading input from a file

CLOUDY normally reads input from the system's `stdin`. This can be changed to another file by calling the routine *cdInput*, which works the same way as the *cdOutput* command described above with the following two exceptions: the default open mode is "r", and supplying an empty string as the file name redirects input back to `stdin`.

8.3.4 cdRead—entering Commands

Command lines are entered by successive calls to routine *cdRead*. The argument of *cdRead* is a null-terminated string containing valid commands. These commands must obey all the rules outlined in Part I.

In the examples below some commands are directly entered as strings (this works when the string is a constant) while other strings are created by writing variables through *sprintf* (a standard C io function). This is necessary when the value of a variable needs to be placed into a string.

```
 char chLine[200];/* this string will hold the command lines we will generate*/

/* this example sends the string straight to cdRead */
nleft = cdRead("title a series of constant pressure models" );

/* write variable to a string then sends the string to cdRead */
hden = 5.4;
sprintf( chLine , "hden %5.2f ", hden );
nleft = cdRead(chLine );

/* this example sends a string that contains double quotes,
 * and so must "escape" them with doubled backslashes */
nleft = cdRead("save overview \"test.ovr\" " );

sprintf( chLine , "coronal %5.2f ", temp );
nleft = cdRead(chLine );

nleft = cdRead("stop zone 1 " );
```

cdRead returns the number of commands that can still be entered before exceeding the size of the storage arrays. The return value was ignored in the examples above. So this routine is an exception to the general rule that a zero return condition indicates success—here it indicates a problem—no further commands can be entered.

It is not now possible to read in more than 4000 command lines because of limits to the size of the character arrays used to store them. This limit is stored as the variable *NKRD*. If more than 4000 lines are read in by calling *cdRead* then *cdRead* will stop after explaining why. It will be necessary to increase *NKRD* if more than 4000 command lines are needed.

8.4   Executing the code

8.4.1   cdDrive—calling the Code

The calculation is performed when routine *cdDrive* is called. *cdDrive* returns a bool indicating whether the calculation aborted. The value *false* indicates a successful calculation. The following shows an example of its use.

```
if( cdDrive() )
{
    printf("problems!\n");
    cdEXIT(EXIT_FAILURE);
}
```

If problems occurred and the results cannot be trusted then the return value is *true*. This will only be set if the calculation suffered a complete meltdown. Routine *cdNwcns* can be called to find out about any problems.

8.4.2   cdNoExec—checking without Computing

If routine *cdNoExec* is called after *cdInit* but before *cdDrive* then only the initial parts of a calculation will be performed when routine *cdDrive* is called.

```
cdInit();

/*read in commands */
cdRead( . . .);

/*tell it not to execute */
cdNoExec();

/*call the code */
if( cdDrive() )
    cdEXIT(EXIT_FAILURE);
```

When *cdDrive* is called after *cdNoExec* the code will generate the incident continuum, set the initial density, and derive the chemical composition. It will then stop just before the initial search for the physical conditions in the first zone. All of the initial printout, summarizing properties of the composition and continuum, will be generated. This provides a quick way to check that a large grid of models will be specified correctly without actually fully computing the grid.

8.5   Ending the code

The code must end by calling *cdEXIT*. This routine will close any open output files and do other needed jobs before exiting. The full output may not be produced if this routine is not called. Routine *cdEXIT* takes a single parameter, *EXIT_SUCCESS* or *EXIT_FAILURE*, standard macros that indicate how the program ended.

The routine *cdEXIT* is a pair with *DEBUG_ENTRY*. Any routine that calls *cdEXIT* must also have a call to *DEBUG_ENTRY* statement at the start. The routine *DEBUG_ENTRY* has a single parameter, a string giving the name of the routine (which should be "main()") in the case of the main program.

8.6   Checking Predictions

This section describes a series of routines that allow predicted quantities to be obtained after the calculation is complete.

8.6.1   cdB21cm—mean magnetic field

The return value is the mean magnetic field weighted by $n(\mathrm{H}^0)dr/T_{spin}$. This is related to the field measured with 21 cm Zeeman observations. A tangled magnetic field is assumed. A magnetic field is not included by default but can be added with the **magnetic field** command, described in Part 1 of this document.

8.6.2   cdColm—the computed column densities

The predicted column densities of some species can be obtained by calling routine *cdColm*:

```
/* want N(C+2) */
if(cdColm("carb", 3 , &colum))
{
    printf(" could not find C+2\n");
}
else
{
    printf("The predicted C+2 column density is %e\n", column );
}
```

The routine returns zero if it found the species, and 1 if it could not. It returns the predicted column density $[\,\mathrm{cm}^{-2}\,]$ as the third argument. The first argument *chLabel* is a four-character string that must agree with the first four characters (upper or lower case) of the name used to indicate the element in the printout. The second (integer) variable *ion* is the spectroscopic designation of the level of ionization, i.e., 1 indicates the atom C^0, 3 indicates C^{+2}, etc.

The ion stage of 0 indicates a special case, a molecule or an excited level of an atom or ion. The label determines the species in this case. Table 8.1 gives the levels and molecules that are recognized. Many of the molecules have fewer than four characters. The label must still contain four characters and spaces are used to fill out the four.

Table 8.1: Special cases cdColm Column Densities

Excited states		molecules
Label	Level	Label
He1*	$He^0 \, 2^3S$	H2
CII*	$C^+ J = 3/2$	H-
C11*	$C^0 J = 0$	H2+
C12*	$C^0 J = 1$	H3+
C13*	$C^0 J = 2$	H2g
C30*	$C^{2+} J = 0$	H2*
C31*	$C^{2+} J = 1$	HeH+
C32*	$C^{2+} J = 2$	CO
O11*	$O^0 J = 2$	OH
O12*	$O^0 J = 1$	H2O
O13*	$O^0 J = 0$	O2
Si2*	$Si^+ J = 3/2$	SiO
		H-
		C2
		C3
		CN
		CH
		CH^+
H2vJ	H_2 any v, J	H2

A large and complex model of the H_2 molecule is computed when the **atom H2** command is included. It is possible to obtain column densities in any v, J level of H_2 by calling routine ***cdH2_colden***.

8.6.3   cdCooling_last—last zone's cooling

The return value is the total cooling rate (erg cm^{-3} s^{-1}) for the last computed zone.

8.6.4   cdDepth_depth—the depth structure of the cloud

This routine returns a vector giving the zone depths (in cm) of the previous iteration. The code uses adaptive logic to control the radial zoning of the model. Neither the number of depth points nor their structure is known in advance. This routine is called with a double precision vector with enough space to hold the structure. The number of depth points is determined by calling *cdnZone* and space must be allocated by the calling routine. Each element of the vector is the depth from the illuminated face to the center of zone *n*.

8.6.5   cdEDEN_last—electron density of last zone

This returns the electron density (cm^{-3}) of the last zone.

8.6.6   cdEmis—emissivity of lines

cdEmis has the same arguments as *cdLine* but returns the local emissivity (erg cm^{-3} s^{-1} for unit filling factor) of the line for the last computed zone. The return value is the index of the line within the line stack if it was found, and the negative of the number of lines in the stack if the line could not be found.

8.6.7   cdGetLineList—sets of emission lines

The routine *cdGetLineList* provides a way to automatically access a list of emission lines.

First enter a list of emission lines into a file. One emission line occurs on each line of the file. It starts with a line label like "H 1" followed by the wavelength of the line. A set of such files is included in the data directory of the distribution files. They have names `LineList*.dat`. You can use these as examples to create your own files.

The first argument to routine *cdGetLineList* is the name of the file containing the line list. If a null string is passed ("") then `LineList_BLR.dat` is used. The code will first try to open the file in the current directory, and if is not present, will try on the path.

cdInit must be called before *cdGetLineList* is called. Next *cdGetLineList* is called, and finally, the actual grid of calculations begins. The predicted intensities of a set of lines are then extracted by calling *cdLine*.

The second and third parameters are a pair of vectors that are defined by the calling program. When the routine *cdGetLineList* is called it uses these vectors to store the labels and wavelengths. Space for the lines is allocated by *cdGetLineList* after it determines how many lines are in the file.

These string and float vectors will then contain the label and wavelength used to identify the lines.
The function returns the number of lines in the list. If problems occurred then -1 is returned.

The following shows an example of getting the lines from `LineList_NLR.dat`, executing
the code, and then obtaining the predicted intensities of all lines listed in `LineList_NLR.dat`
by calling *cdLine*.

```
/* define variables */
exit_type exit_status = EXIT_SUCCESS;
vector<char*> chLabel;
vector<float> wl;
/* initialize the code */
cdInit();
/* get list of lines from a line list included in the distribution */
if( (nLines=cdGetLineList("LineList_NLR.dat",chLabel,wl)) < 0 )
{
    /* this is the error exit - could not obtain the lines */
    cdEXIT(EXIT_FAILURE);
}
- - - - - - missing code
/* now do the calculation */
cdInit();
/* missing commands here, then call the code */
- - - - - - missing code
if( cdDrive() )
    exit_status = EXIT_FAILURE;
- - - - - - missing commands go here
/* now print the predicted emission line intensities */
for( long n=0; n < nLines; ++n )
{
    if( cdLine( chLabel[n], wl[n], &relative, &absolute ) < 0 )
    {
        fprintf(stderr,"did not find %4s%5li\n",chLabel[n],wl[n]);
    }
    print("%.3e\n", relative);
}
/* free the memory that was allocated in cdGetLineList */
for( size_t i=0; i < chLabel.size(); ++i )
    delete[] chLabel[i];
chLabel.clear();
cdEXIT(exit_status);
```

8.6.8   cdH2_colden—state-specific column densities of H_2

This returns the column density of any level in the X ground electronic state of H_2. This command
only works when the large H_2 molecule is included with the **atom H2** command. It has two
integer arguments, the vibration and rotation quantum numbers of a level in X. If both are zero or
greater the routine returns the column density in that level. If the vibration quantum number is
negative then a summed column density is turned. If $v < 0$ and $J = 0$ it returns the total H_2 column
density, if $J = 1$ it returns the ortho column density, and if $J = 2$ the para column density. If the
indices do not make sense the routine prints a message and returns -1.

Here are some examples:

```
/* total H2 column density */
total = cdH2_colden( -1 , 0 );
/* ortho column density */
ortho = cdH2_colden( -1 , 1 );
/* para column density */
para = cdH2_colden( -1 , 2 );
/* column density in 0, 0 */
total00 = cdH2_colden( 0 , 0 );
```

8.6.9   cdH2_Line—an H_2 emission line intensity

More than half a million H_2 lines are predicted and there will be instances where two H_2 lines have nearly the same wavelength. Identification of a particular transition within the list of lines can be ambiguous. This command finds the intensity and luminosity of a transition by specifying its upper and lower n, v, J levels. The first six arguments give the n, v, J indices of the upper and lower levels in that order. The last two variables are double pointers that return the intensity and luminosity of the transition. The function returns 1 if it finds the line and 0 if it did not. (This behavior follows that of *cdLine* rather than the standard C++ conventions on function return values.) Currently this only works for the ground electronic state.

   Here is an example:

```
double xInten , xLumin;
/* the 1-0 S(1) at 2.121 microns */
if( cdH2_Lines( 0,1,3 , 0,0,1 , &xInten , &xLumin ) == 0 )
{
    printf("could not find line.\n");
}
```

8.6.10   cdHeating_last—last zone's heating

The total heating rate (erg cm^{-3} s^{-1}) for the last computed zone is returned.

8.6.11   cdIonFrac—the computed ionization fractions

The predicted ionization fractions,[1] averaged over radius, area, or volume, can be accessed by calling the subroutine *cdIonFrac*. The average over radius is defined as

$$\left\langle \frac{n\left(S^{+n}\right)}{n\left(S_{gas}\right)} \right\rangle = \frac{\int n(S^{+n})\, f(l)\, dl}{\int n(S_{gas})\, f(l)\, dl} \tag{8.1}$$

where $n(S_{gas})$ is the total gas phase density of the element. The average over area is defined as

$$\left\langle \frac{n\left(S^{+n}\right)}{n\left(S_{gas}\right)} \right\rangle = \frac{\int n(S^{+n})\, f(r)\, r\, dr}{\int n(S_{gas})\, f(r)\, r\, dr}. \tag{8.2}$$

---

[1]Before version 96 the ionization fractions only included atoms and ions. They now also include molecules. The sum of the atomic and ionic fractions will not add up to unity if a significant fraction of the element is in molecules.

The average over volume is defined as

$$\left\langle \frac{n(S^{+n})}{n(S_{gas})} \right\rangle = \frac{\int n(S^{+n})\, f(r)\, r^2\, dr}{\int n(S_{gas})\, f(r)\, r^2\, dr}. \tag{8.3}$$

Two sample calls to the routine follow:

```
/* false below means to not include electrons in the mean */
if( cdIonFrac("carb", 2, &frac, "radius", false) )
{
    cdEXIT(EXIT_FAILURE);
}
printf("The predicted ionization fraction over radius is %g\n", frac);
/* true below means to include electrons in the mean */
if( cdIonFrac("carb", 2, &frac, "radius", true) )
{
    cdEXIT(EXIT_FAILURE);
}
printf("Ionization fraction wrt radius end elec den is %g\n", frac);
```

The first argument is a four-character identifier that must agree with the first four characters (upper or lower case) used to indicate the element in the printout. The second integer argument is the spectroscopic designation of the level of ionization, i.e., 1 indicates C^0, 3 indicates the second ion C^{+2}, etc. The third argument returns the predicted ionization fraction S^{+n}/S_{gas}. The 4th argument is a six-character variable (plus end of string sentinel) which must be either "*radius*", "*area*", or "*volume*" (either upper or lower case). This string determines whether the ionization fraction returned is weighted with respect to radius, area, or volume. These would be appropriate to compare to pencil beam observations, integrated long slit observations, or observations of the whole nebula, respectively. The last argument determines whether (*true*) or not (*false*) the ionization fraction is also weighted with respect to the electron density. The function returns zero if the ion was found and non-zero if an error occurred.

An ionization stage of zero will request the fraction of an element within a molecule. If the element name is "H2 " (the letters H2 followed by two spaces) then the fraction of hydrogen in H_2, $2n(H_2)/n(H_{tot})$, will be returned. Currently only H_2 is implemented.

8.6.12   cdLine—emission-line intensities

This finds the intensity or luminosity of any line. The label and wavelength of the line are specified and the routine returns the relative intensity and the log of the absolute intensity or luminosity. The following is an example.

```
#include "cddefines.h"
#include "cddrive.h"
double relint , absint;
long nType=0;

if( cdLine( "totl" , 4861 , &relint , &absint , nType ) < 0 )
    printf("did not find this line\n");
```

The first argument in the call is the line label, the four-character null-terminated string (upper or lower case) used by the code to identify the line in the main emission-line printout. The second variable gives the wavelength of the line in Angstroms. Both of these must exactly match the label and wavelength used by CLOUDY to identify the line (see the chapter *Lines* for a full description). The third variable (*relint* in the above example) is a double-precision pointer to the relative intensity of the line (relative to the normalization line, usually Hβ, and reset with the **normalize** command). The log of the intensity (erg cm^{-2} s^{-1}) or luminosity (erg s^{-1}) of the line is returned as the fourth double-precision pointer (*absint* in the above example). If the intensity of the line is zero or the line was not found then this variable will be set to -37.

The last (optional) parameter indicates which type of line prediction to return. The integer types are 0 intrinsic, 1 emergent, 2 intrinsic cumulative, 3 emergent cumulative intensities.

The intensities returned by when the last parameter is missing or it is zero are those printed with the heading *Intrinsic Intensities*. This does not include the effects of possible reflection off a background molecular cloud or of absorption from outside the emission-line region. If the last parameter is 1 then emergent intensities are reported. The values 2 and 3 are the equivalent quantities for the cumulative intensities in a time dependent model.

If *cdLine* finds the line it returns the index of the line within the stack of emission lines. So a positive return value indicates success. It returns the negative of the total number of lines in the stack if the line is not found. This may occur if the line wavelength or label was mistyped. This is an exception to the C++ function return convention in which a normal return is zero and an abnormal return is non-zero. A positive value indicates a successful return.

There is a special version, *cdLine_ip*, which takes the line index as the first argument and return the relative and absolute intensities as the second and their quantities. The line index is not a constant for a given version of the code. It depends on the number of elements that are included, the sizes of the iso-sequence model atoms, and whether other models like H$_2$ are included.

8.6.13   cdnZone—how many zones in the last iteration?

The routine returns the number of zones in the previous iteration.

8.6.14   cdPressure_depth—pressure structure of the last iteration

The pressure as a function of depth, for the last iteration, is obtained by calling routine *cdPressure_depth*. This routine has three arguments, pointers to vectors giving the total pressure, the gas pressure, and the line radiation pressure. All are double precision vectors and the calling routine must have allocated space for these before calling the routine. The total number of elements needed for each vector is the number of zones in the last iteration and is obtained by calling routine *cdnZone*.

8.6.15   cdPressure_last—pressure of the last zone

The pressure for the last zone is obtained by calling routine *cdPresure_last*. This routine has three arguments, pointers to the total pressure, the gas pressure, and the line radiation pressure. All are double precision variables. The total pressure includes all contributors to the pressure and may include magnetic, radiation, and turbulent pressure in addition to thermal pressure.

8.6.16   cdSPEC—get predicted spectrum

This routine provides an interface between CLOUDY and Keith Arnaud's X-Ray spectral analysis program *XSPEC*. Depending on which option is used, it will return the incident continuum, the attenuated incident continuum, the reflected continuum, the diffuse continuous continuum, outward direction diffuse continuous emission, reflected lines, or outward lines. All are $4\pi\nu J_\nu$ [erg cm^{-2} s^{-1}] and assume full coverage of the continuum source. Details are given in `cddrive.h`.

   This is no longer being developed and will soon be removed. Use the **save FITS** command, described in Part 1, instead.

8.6.17   cdTemp—the computed mean temperature

Routine *cdTemp* returns the mean electron temperature weighted with respect to some species. The first parameter is a four character null-terminated string giving the first four letters (upper or lower case) of the name of an element as spelled by the code. The second parameter is the ionization stage, with 1 for the atom, 2 the first ion, etc. The third parameter will be the computed mean temperature. The last parameter is a six-character null-terminated string, either "radius", "area", or "volume", that says whether the temperature should be weighted with respect to radius, area, or volume. These would be appropriate to compare to pencil beam observations, integrated long slit observations, or observations of the whole nebula, respectively. The mean temperature weighted by ion $+n$ of element S over radius is defined as

$$\langle T\left(S^{+n}\right)\rangle = \frac{\int T\, n(S^{+n})\, f(l)\, dl}{\int n(S^{+n})\, f(l)\, dl} \quad [\mathrm{K}], \tag{8.4}$$

the average over area is

$$\langle T\left(S^{+n}\right)\rangle = \frac{\int T\, n(S^{+n})\, f(r)\, r\, dr}{\int n(S^{+n})\, f(r)\, r\, dr} \quad [\mathrm{K}], \tag{8.5}$$

and the average over volume is

$$\langle T\left(S^{+n}\right)\rangle = \frac{\int T\, n(S^{+n})\, f(r)\, r^2\, dr}{\int n(S^{+n})\, f(r)\, r^2\, dr} \quad [\mathrm{K}]. \tag{8.6}$$

The routine returns 0 if it finds the species, and 1 if it could not find the species. The following is an example of a call:

```
if( cdTemp("carb", 2, &temp, "radius") )
{
    cdEXIT(EXIT_FAILURE);
}
printf("The mean C+2 temperature is %g\n", temp);
```

   An ionization stage of zero requests one of the following special temperatures. All these can be weighted by radius, area, or volume.

21 cm-related temperatures: The routine will return one of three temperatures related to 21 cm observations. The label "*21cm*" will return the mean of $n(\mathrm{H}^0)/T_{kin}$, the harmonic mean gas

kinetic temperature weighted with respect to the atomic hydrogen density. The label "**spin**" will return the mean weighted by $n(H^0)/T_{spin}$, the harmonic mean of the 21 cm spin temperature weighted with respect to the atomic hydrogen density. Finally the label "**opti**" returns the temperature derived from the ratio of Lα to 21 cm optical depths (see AGN3 section 5.5).

Molecular hydrogen: The label "**H2_ _**" (notice that there were two spaces after the 2) and an ionization stage of zero will return the mean temperature weighted with respect to the H_2 density.

Simple mean temperature: If the label consists of four spaces, as in "   ", and the ionization stage is zero, the routine will return the mean temperature, but not weighted by any species. If the label is "**TeNe**" then the routine will return the mean temperature weighted by the electron density.

8.6.18   cdTemp_last—the temperature of the last zone

The kinetic temperature of the last zone is obtained by calling the function *cdLastTemp*. The function has no arguments and its return value is the temperature.

8.6.19   cdTimescales—several timescales.

This routine has three arguments, pointers to doubles that return the timescales [s] for several processes. These are the thermal timescale, the hydrogen recombination timescale, and the H_2 formation timescale.

8.7   Other information

8.7.1   cdDate(cdString)

The date when the current version of the code was released will be placed as a null-terminated string. The string is passed as an argument and the calling program must have allocated enough room for the string.

8.7.2   cdVersion(cdString)

The code's version number will be placed as a null-terminated string into the string passed as an argument. The version number is a string rather than a float since it can end with a letter of the alphabet. The calling program must allocate enough space for the string.

8.7.3   double cdExecTime()

This returns the time that has elapsed since the previous call to *cdInit*.

8.8   Printing comments

CLOUDY is designed to be autonomous and self aware. It constantly monitors itself to make sure that it is doing the physics correctly. After the calculation is complete, but before the emission lines are printed, it generates a series of statements that indicate warnings, cautions, comments, and surprises. These should be examined to confirm that the calculation is likely to be valid. A series of routines allows the driving code to determine whether these comments were generated, what type they were, and to print then into an arbitrary open file.

8.8.1   Were comments generated?

Routine *cdNwcns* will return the number of warnings, cautions, surprises, notes, and temperature and pressure failures:

```
cdNwcns( &lgAbort , &nw , &nc , &nn , &ns , &nte , &npe , &nione, &neden )
```

where the first variable is a *bool* flag indicating whether the calculation aborted, *nw* is the number of warnings generated (if this number is non-zero, then the calculation has serious problems), *nc* is the number of cautions generated (these are less severe than warnings, but are still a cause of concern), and *nn* and *ns* are the number of notes and surprises. The next two arguments are the number of temperature and pressure failures. The last two are the number of ionization and electron density failures. There should not be any failures in a successful calculation.

   If either of the first two variables are non-zero then the code ran into serious problems. An abort is far more serious than a warning since it indicates catastrophic failure. I would appreciate learning about these. Please post details on the code's discussion board.

8.8.2   Printing the comments.

A series of comments normally appear after the last zone. These may be printed into any file by calling the series of subroutines described here. In all cases the routines take as an argument a file handle which must point to a file that has already been opened for writing.

```
/* output the comments into a file
 * first define the file handle, then open the file for writing */
FILE *ioOUT;
if( (IoOUT = fopen( "comments.txt", "w") ) == NULL )
{
    printf("error creating comments.txt file\n");
    cdEXIT(EXIT_FAILURE);
}
--- missing code to do calculation  ---
/*print the reason the calculation stopped, and geometry*/
cdReasonGeo(ioOUT);
/*print the warnings*/
cdWarnings(ioOUT);
/*next print the cautions*/
cdCautions(ioOUT);
/*now print any surprising results*/
cdSurprises(ioOUT);
```

```
/*now print the notes*/
cdNotes(ioOUT);
fclose(ioOUT);
```

cdReasonGeo(FILE *io) It is very important to understand why the calculation stopped. The first two lines after the last zone results give the reason the calculation stopped and the type of geometry. This information will be printed into the file whose handle is the argument.

cdWarnings(FILE *io) All warnings (denoted by "W-") will be printed.

cdCautions(FILE *io) All cautions (denoted by a "C-") will be printed.

cdSurprises(FILE *io) All surprises (denoted by a "!") are printed.

cdNotes(FILE *io) The notes concerning the calculation will be printed.

8.8.3 cdErrors(FILE *io)—printing a summary of any problems

Routine *cdErrors(FILE *io)* will generate a summary of any problems that happened during a calculation. The argument is a pointer to the output file where the summary will be placed. The calling program must have already opened the file for writing. If problems occurred in the calculation, such as temperature or pressure failures, warnings, or cautions, will be printed along with the calculation's title.

8.8.4 cdPrintCommands(FILE *io)—print the command stack

The entire series of input commands will be written into the file. The single argument is a file handle that points to a previously opened file. The commands are preceded and followed by lines that begin with "c =====" to easily identify their start and end.

8.8.5 setbuf or the no buffering command

Programs produce output by writing into a buffer. They only place information on the disk once the buffer is full. If a C++ program crashes before this buffer is "flushed" the information within the buffer will be lost. This poses a problem if the printout generated just before the crash is needed for debugging. The C++ io library provides a routine, *setbuf*, that can turn file buffering off. The following sequence would open a file and turn buffering off:

```
ioDATA = open_data("test.out","w",AS_LOCAL_ONLY);
/* turn off buffering so we see results as then happen */
setbuf( ioDATA , NULL );
```

Note that the call to *setbuf must* come immediately after the call to *open_data* before any output is written. The **no buffering** command, described in Part 1 of this document, will do the same thing. Note that turning off buffering will make the code run far more slowly.

8.9   Example Call as a Subroutine

The file template.cpp shown below contains a skeleton program showing how you can call
Cloudy as a subroutine. You can find the program in the directory tsuite/programs. Use this
as a starting point to write your own program. The area where you can insert your own code is
clearly indicated in the template.

```
/* This file is part of Cloudy and is copyright (C)1978-2010 by Gary J. Ferland and
 * others.  For conditions of distribution and use see copyright notice in license.txt */
/* Template main program for calling Cloudy as a subroutine
 * Routine returns 0 if model is ok, and 1 if problems occurred. */
#include "cddefines.h"
#include "cddrive.h"

int main( int argc, char *argv[] )
{
    exit_type exit_status = ES_SUCCESS;

    DEBUG_ENTRY( "main()" ); // do not remove this!

    try {
        // ========================================================================
        // START ENTERING YOUR CODE AFTER THIS LINE
        // inside this try block you can enter your code to call Cloudy as a subroutine
        // replace the code between here and the next ===== line with your own program

        long nleft;

        // you can call Cloudy in a loop if you want...
        for( int i=0; i < 1; ++i )
        {
            // the code always needs to be initialized first
            cdInit();
            // replace this with a series of calls to cdRead to define the input script
            // this particular command line exercises the smoke test
            nleft = cdRead( "test" );
            // this calls Cloudy to execute the input script you defined above
            if( cdDrive() )
                exit_status = ES_FAILURE;
        }

        cdEXIT(exit_status); // always exit with cdEXIT, this assures files are properly closed.
        // ========================================================================
        // THIS IS THE END OF THE PROGRAM YOU WROTE
        // this ends the try block - your code calling Cloudy ends above this line
    }
    // here we catch all the possible exceptions that the code can throw, do not remove this!
    catch( bad_alloc )
    {
        fprintf( ioQQQ, " DISASTER - A memory allocation has failed. Most likely your computer "
            "ran out of memory.\n Try monitoring the memory use of your run. Bailing out...\n" );
        exit_status = ES_BAD_ALLOC;
    }
    catch( out_of_range& e )
    {
        fprintf( ioQQQ, " DISASTER - An out_of_range exception was caught, what() = %s. Bailing out...\n",
            e.what() );
        exit_status = ES_OUT_OF_RANGE;
    }
    catch( bad_assert& e )
    {
        MyAssert( e.file(), e.line() , e.comment() );
        exit_status = ES_BAD_ASSERT;
    }
#ifdef CATCH_SIGNAL
    catch( bad_signal& e )
    {
        if( ioQQQ != NULL )
        {
            if( e.sig() == SIGINT || e.sig() == SIGQUIT )
            {
                fprintf( ioQQQ, " User interrupt request. Bailing out...\n" );
                exit_status = ES_USER_INTERRUPT;
            }
            else if( e.sig() == SIGTERM )
            {
                fprintf( ioQQQ, " Termination request. Bailing out...\n" );
                exit_status = ES_TERMINATION_REQUEST;
            }
            else if( e.sig() == SIGILL )
            {
                fprintf( ioQQQ, " DISASTER - An illegal instruction was found. Bailing out...\n" );
                exit_status = ES_ILLEGAL_INSTRUCTION;
            }
            else if( e.sig() == SIGFPE )
            {
                fprintf( ioQQQ, " DISASTER - A floating point exception occurred. Bailing out...\n" );
```

```
                exit_status = ES_FP_EXCEPTION;
            }
            else if( e.sig() == SIGSEGV )
            {
                fprintf( ioQQQ, " DISASTER - A segmentation violation occurred. Bailing out...\n" );
                exit_status = ES_SEGFAULT;
            }
#           ifdef SIGBUS
            else if( e.sig() == SIGBUS )
            {
                fprintf( ioQQQ, " DISASTER - A bus error occurred. Bailing out...\n" );
                exit_status = ES_BUS_ERROR;
            }
#           endif
            else
            {
                fprintf( ioQQQ, " DISASTER - A signal %d was caught. Bailing out...\n", e.sig() );
                exit_status = ES_UNKNOWN_SIGNAL;
            }

        }
    }
#endif
    catch( cloudy_exit& e )
    {
        if( ioQQQ != NULL )
        {
            ostringstream oss;
            oss << " [Stop in " << e.routine();
            oss << " at " << e.file() << ":" << e.line();
            if( e.exit_status() == 0 )
                oss << ", Cloudy exited OK]";
            else
                oss << ", something went wrong]";
            fprintf( ioQQQ, "%s\n", oss.str().c_str() );
        }
        exit_status = e.exit_status();
    }
    catch( std::exception& e )
    {
        fprintf( ioQQQ, " DISASTER - An unknown exception was caught, what() = %s. Bailing out...\n",
            e.what() );
        exit_status = ES_UNKNOWN_EXCEPTION;
    }
    // generic catch-all in case we forget any specific exception above... so this MUST be the last one.
    catch( ... )
    {
        fprintf( ioQQQ, " DISASTER - An unknown exception was caught. Bailing out...\n" );
        exit_status = ES_UNKNOWN_EXCEPTION;
    }

    cdPrepareExit(exit_status);

    return exit_status;
}
```

8.10 Computing Grids of Calculations

Today I usually use the code to compute results, extract information, and save desired quantities.
The following example illustrates producing a series of models with increasing gas temperature
and density. The program calculates the line strength of various diagnostic [O III] lines
normalized to [O III] $\lambda 5007$ and writes these to a file. This example is the program
vary_nete.cpp, which you can find in the directory tsuite/programs/vary_nete.

  This example only saves results for a few lines. Often a large number of lines are needed. A call
to *cdLineList* provides an easy way to obtain large numbers of lines whose labels are stored in a
file.

```
/* This file is part of Cloudy and is copyright (C)1978-2010 by Gary J. Ferland and
 * others.  For conditions of distribution and use see copyright notice in license.txt */
/*main program that calls cloudy when used as a stand-alone program */
#include "cddefines.h"
#include "cddrive.h"

/*int main( int argc, char *argv[] )*/
int main( void )
{
    exit_type exit_status = ES_SUCCESS;

    DEBUG_ENTRY( "main()" );
```

```c
try {
    double hdenLimit , hdenInit , hden, TeInc ,
        r5007,r4363,r1665,ro3_88,ro3_52, r1661 ,
        absolute , temp, TeLimit, TeInit , hdenInc ;

    FILE *ioDATA ;
    char chLine[100];

    /* this is limit to the number of command chLines we can still put in */
    long int nleft;

    cdOutput( "vary_nete.out" );

    /* calculation's results will go to this file*/
    if( (ioDATA = fopen("vary_nete.txt","w")) == NULL )
    {
        printf(" could not open varyTF.txt for writing.\n");
        cdEXIT(EXIT_FAILURE);
    }

    /* the range of density, and the increment in density, for this grid */
    hdenInit = 0.;
    hdenLimit = 7.;
    hdenInc = 1.;

    /* the range of log temperatures for the grid */
    TeInit = 3.5;
    TeLimit = 4.5;
    /* multiplicative inc - log */
    TeInc = .2;

    /* set the density and temperature to the initial values */
    hden = hdenInit;
    temp = TeInit;

    /* print the header for the data file */
    fprintf(ioDATA,
        "density\ttemp\t[O III] 5007\t[O III] 4363\t[O III] 1665\t[O III] 88\t[O III] 52\t[O III] 1661\n");

    while( hden < 1.01 * hdenLimit )
    {
        while( temp < 1.01*TeLimit )
        {

            /* initialize the code for this run */
            cdInit();

            /* option to not execute the code, uncomment when debugging setup */
            /*cdNoexec( );*/

            /* gas temperature for this calculation */
            sprintf(chLine,"constant temperature %f", temp);
            nleft = cdRead( chLine  );

            /* gas density for this calculation */
            sprintf(chLine,"hden %f", hden);
            nleft = cdRead( chLine  );

            /* only want the first zone */
            nleft = cdRead( "stop zone 1 "  );

            /* speed up calculation by not included some elements and lines */
            nleft = cdRead( "init file \"fast.ini\" "  );

            nleft = cdRead( "normalize to \"o  3\" 5007 "  );

            /* an incident continuum must be specified to get the code
             * to run at all - not very important since we will set
             * the gas temperature */
            nleft = cdRead( "blackbody 40000  "  );
            nleft = cdRead( "ionization parameter -3  "  );

            /* actually call the code */
            if( cdDrive() )
                exit_status = ES_FAILURE;
            /* flush the output so we see it on the screen */
            fflush(ioQQQ);

            fprintf(ioDATA,"%.3e\t%.5f",    hden, temp );
            fprintf(stderr,"%.3e\t%.5f",    hden, temp );

            /*********************** O III ******************/
            if( cdLine( "O  3", 5007 , &r5007 , &absolute ) < 0 )
            {
                printf("did not find O  3 5007\n");
                exit_status = ES_FAILURE;
            }
            /* this should be unity */
            fprintf(ioDATA,"\t%.3e",    r5007 );
            fprintf(stderr,"\t%.3e",    r5007 );
```

```
                if( cdLine( "totl", 4363 , &r4363 , &absolute ) < 0 )
                {
                    printf("did not find totl 4363\n");
                    exit_status = ES_FAILURE;
                }
                fprintf(ioDATA,"\t%.3e",    r4363 );
                fprintf(stderr,"\t%.3e",    r4363 );

                if( cdLine( "totl", 1665 , &r1665 , &absolute ) < 0 )
                {
                    printf("did not find totl 1665\n");
                    exit_status = ES_FAILURE;
                }
                fprintf(ioDATA,"\t%.3e",    r1665 );
                fprintf(stderr,"\t%.3e",    r1665 );

                if( cdLine( "O  3", 883300 , &ro3_88 , &absolute ) < 0 )
                {
                    printf("did not find O  3 88\n");
                    exit_status = ES_FAILURE;
                }
                fprintf(ioDATA,"\t%.3e",    ro3_88 );
                fprintf(stderr,"\t%.3e",    ro3_88 );

                if( cdLine( "O  3", 518000 , &ro3_52 , &absolute ) < 0 )
                {
                    printf("did not find O  3 52\n");
                    exit_status = ES_FAILURE;
                }
                fprintf(ioDATA,"\t%.3e",    ro3_52 );
                fprintf(stderr,"\t%.3e",    ro3_52 );

                if( cdLine( "O  3", 1661 , &r1661 , &absolute ) < 0 )
                {
                    printf("did not find O  3 1661\n");
                    exit_status = ES_FAILURE;
                }
                fprintf(ioDATA,"\t%.3e",    r1661 );
                fprintf(stderr,"\t%.3e",    r1661 );


                /*********************** end lines with lf and flush it ***************/
                fprintf(ioDATA,"\n" );
                fflush(ioDATA );
                fprintf(stderr,"\n" );

                temp += TeInc;
            }
            temp = TeInit;
            hden += hdenInc;
        }

        cdEXIT(exit_status);
    }
    catch( bad_alloc )
    {
        fprintf( ioQQQ, " DISASTER - A memory allocation has failed. Most likely your computer "
            "ran out of memory.\n Try monitoring the memory use of your run. Bailing out...\n" );
        exit_status = ES_BAD_ALLOC;
    }
    catch( out_of_range& e )
    {
        fprintf( ioQQQ, " DISASTER - An out_of_range exception was caught, what() = %s. Bailing out...\n",
            e.what() );
        exit_status = ES_OUT_OF_RANGE;
    }
    catch( bad_assert& e )
    {
        MyAssert( e.file(), e.line() , e.comment() );
        exit_status = ES_BAD_ASSERT;
    }
#ifdef CATCH_SIGNAL
    catch( bad_signal& e )
    {
        if( ioQQQ != NULL )
        {
            if( e.sig() == SIGINT || e.sig() == SIGQUIT )
            {
                fprintf( ioQQQ, " User interrupt request. Bailing out...\n" );
                exit_status = ES_USER_INTERRUPT;
            }
            else if( e.sig() == SIGTERM )
            {
                fprintf( ioQQQ, " Termination request. Bailing out...\n" );
                exit_status = ES_TERMINATION_REQUEST;
            }
            else if( e.sig() == SIGILL )
            {
                fprintf( ioQQQ, " DISASTER - An illegal instruction was found. Bailing out...\n" );
                exit_status = ES_ILLEGAL_INSTRUCTION;
            }
            else if( e.sig() == SIGFPE )
```

```
            {
                fprintf( ioQQQ, " DISASTER - A floating point exception occurred. Bailing out...\n" );
                exit_status = ES_FP_EXCEPTION;
            }
            else if( e.sig() == SIGSEGV )
            {
                fprintf( ioQQQ, " DISASTER - A segmentation violation occurred. Bailing out...\n" );
                exit_status = ES_SEGFAULT;
            }
#           ifdef SIGBUS
            else if( e.sig() == SIGBUS )
            {
                fprintf( ioQQQ, " DISASTER - A bus error occurred. Bailing out...\n" );
                exit_status = ES_BUS_ERROR;
            }
#           endif
            else
            {
                fprintf( ioQQQ, " DISASTER - A signal %d was caught. Bailing out...\n", e.sig() );
                exit_status = ES_UNKNOWN_SIGNAL;
            }

        }
    }
#endif
    catch( cloudy_exit& e )
    {
        if( ioQQQ != NULL )
        {
            ostringstream oss;
            oss << " [Stop in " << e.routine();
            oss << " at " << e.file() << ":" << e.line();
            if( e.exit_status() == 0 )
                oss << ", Cloudy exited OK]";
            else
                oss << ", something went wrong]";
            fprintf( ioQQQ, "%s\n", oss.str().c_str() );
        }
        exit_status = e.exit_status();
    }
    catch( std::exception& e )
    {
        fprintf( ioQQQ, " DISASTER - An unknown exception was caught, what() = %s. Bailing out...\n",
            e.what() );
        exit_status = ES_UNKNOWN_EXCEPTION;
    }
    // generic catch-all in case we forget any specific exception above... so this MUST be the last one.
    catch( ... )
    {
        fprintf( ioQQQ, " DISASTER - An unknown exception was caught. Bailing out...\n" );
        exit_status = ES_UNKNOWN_EXCEPTION;
    }

    cdPrepareExit(exit_status);

    return exit_status;
}
```

Chapter 9

THE EMISSION LINES

9.1 Overview

The following sections outline the emission lines predicted by CLOUDY. Before version 90 of the code all lines were listed in the sub-section immediately following this section. The code is being modified to bring all lines into a common line class, as the code moves to C++ and objects. This chapter will remain incomplete until this work is finished.

9.2 The main emission-line printout

The main emission line printout was briefly described in the Chapter *OUTPUT*. This section goes into more detail.

Output organization. The printed list is sorted into four large groups of columns, with each large column sub-divided into four smaller sub-columns. The first sub-column is either the spectroscopic designation of the ion producing the line or an indication of how the line is formed. The second sub-column is the line wavelength, with a 0 to indicate a continuum. The third sub-column is the log of the power in the line, in the units given in the header (erg s^{-1} into either 4π sr or cm^{-2}). The last sub-column is the intensity of the line relative to the reference line, usually Hβ , unless this is reset with the **normalize** command.

These lines can be printed as a single large column, and can be sorted by wavelength or intensity. These options are controlled by the **print line** command described in Part I of this document.

9.2.1 Intrinsic and emergent line intensities

The computed emission-line spectrum is divided into two groups. The first group of lines, called "Intrinsic line intensities", gives the intrinsic intensity of the lines, and does not include the reddening effects of internal grains due to the photon's passage out of the nebula. The second group includes the effects of grain scattering and absorption and has the header "Emergent Line Intensities". The intensities are the total intensities observed from the illuminated face, including both absorption and scattering by grains.

9.2.2   Line identification

Line wavelengths. These are given in various units. Numbers ending in "A" are wavelengths in Angstroms (Å). For instance, Hβ is given by "H 1 4861A". Wavelengths in microns are indicated by "m", an example, the strong [O III] IR line, is "O 3 51.80m".

   The code follows the convention that wavelengths longward of 2000Å are given in air and shorter wavelengths in vacuum. Continua are usually indicated by a wavelength of zero.

9.2.3   Blocks of lines. . .

Lines are organized by common origin with a comment, ending in a series of periods ". . .", beginning the section. As an example, the first commented block of lines begins with "general properties. . .". The following subsections give overviews of the lines.

9.2.4   General properties. . .

This mainly summarizes heating and cooling agents for the model.

TOTL 4861 and TOTL 1216, are the total intensities of Hβ and Lα, as predicted by the multi-level H atom. These intensities are the results of calculations that include all collisional, radiative, and optical depth effects.

Inci The total energy in the incident continuum. This entry will not be included if the **aperture** command is in effect.

TotH and TotC give the total heating and cooling. These will be nearly equal in equilibrium.

BFH1 and BFHx are the heating due to photoionization of ground state and excited state hydrogen respectively.

He1i, 3He1 , heating due to ground state He and the triplets.

BFHe and TotM are the heating due to helium and metal photoionization.

Pair heating due to pair production.

ComH , ComC , Compton heating, cooling.

CT H CT C charge transfer heating and cooling.

extH extC "extra" heating or cooling added to model.

e-e+ 511 The positron line.

Expn , expansion, or adiabatic, cooling

H FB , H radiative recombination cooling

HFBc, HFBc , hydrogen net free-bound cooling and heating

Iind , cooling due to induced recombination of hydrogen

3He2 , cooling due to induced recombination of fully ionized helium

Cycn , cyclotron cooling

9.2.5 Continua...

These give intensities of various continua. These are either the total integrated continuum or the product νF_ν at certain energies.

Continuum bands

The file `continuum_bands.ini` in the data directory specifies a set of wavelength bands. The code will integrate over these bands to find the total radiated luminosity and enter this into the main emission-line stack. Currently this is a simple sum of the energy emitted between the upper and lower bounds of the band, and does not take into account the transmission function of particular instruments. The `continuum_bands.ini` file can be edited to change the number of bands or their detailed properties. Table 9.1 lists the bands in the file at the time of this writing. The first and second columns give the label and wavelength as they appear in the printout. The last column gives the wavelength range for the integration. These entries will not be included if the **aperture** command is in effect. Please consult the file to see its current contents and feel free to add your own bands.

Other continua

Bac 3646 residual flux at head of Balmer continuum, νF_ν. This entry will not be included if the **aperture** command is in effect.

cout 3646 cref 3646 , outward, reflected continuum at peak of Balmer Jump. These entries will not be included if the **aperture** command is in effect.

thin 3646 , residual flux at head of Balmer continuum, optically thin limit. This entry will not be included if the **aperture** command is in effect.

Inci 4860, Inci 1215 , incident continua near Hα and Lα. These entries will not be included if the **aperture** command is in effect.

Ba C 0 , integrated Balmer continuum

PA C 0 , integrated Paschen continuum

HeFF 0 , He brems emission

HeFB 0 , He recombination cooling

MeFB 0 , heavy element recombination cooling

MeFF 0 , metal brems emission

Table 9.1: Default continuum bands

Label	Wavelength	Wavelength Range
FIR	83.00m	40.00m – 500.0m
TIR	1800m	500.0m – 3100m
NIRa	2.850m	7000A – 40.00m
NIRb	3.000m	10000A – 5.000m
MIRa	15.00m	5.000m – 25.00m
MIRb	22.50m	5.000m – 40.00m
NMIR	21.75m	7000A – 40.00m
TFIR	611.2m	122.5m – 1100m
TALL	10000A	0.010A – 10000m
F12	12.00m	8.500m – 15.00m
F25	25.00m	19.00m – 30.00m
F60	60.00m	40.00m – 80.00m
F100	100.0m	83.00m – 120.0m
MIPS	24.00m	20.80m – 26.10m
MIPS	70.00m	61.00m – 80.00m
MIPS	160.0m	140.0m – 174.0m
IRAC	3.600m	3.160m – 3.920m
IRAC	4.500m	4.000m – 5.020m
IRAC	5.800m	5.000m – 6.400m
IRAC	8.000m	6.500m – 9.300m
SPR1	250.0m	212.0m – 288.0m
SPR2	350.0m	297.0m – 405.0m
SPR3	500.0m	414.0m – 600.0m
PAC1	70.00m	60.00m – 82.00m
PAC2	100.0m	84.00m – 122.0m
PAC3	160.0m	130.0m – 198.0m
PAH	3.300m	3.250m – 3.350m
PAHC	3.230m	3.200m – 3.250m
PAHC	3.370m	3.350m – 3.400m
PAH	6.200m	5.900m – 6.400m
PAHC	5.650m	5.400m – 5.900m
PAH	7.900m	7.400m – 8.400m
PAHC	6.900m	6.400m – 7.400m
PAH	11.30m	11.10m – 11.50m
PAHC	10.90m	10.70m – 11.10m
PAH	11.80m	11.60m – 12.30m
PAHC	12.65m	12.30m – 13.00m
PAH	13.30m	12.90m – 13.70m
PAHC	14.10m	13.70m – 14.50m
Bcon	3640A	911.6A – 3646A
Pcon	5000A	3646A – 8204A

ToFF 0 , total brems emission

FF x , part of H brems, in x-ray beyond 0.5KeV

eeff , electron - electron brems

   nFnu 122m, *nInu 122m*, *InwT 122m*, *InwC 122m*, a large list of continua at selected wavelengths will be printed if the **print continuum** command is entered. These entries will not be included if the **aperture** command is in effect. The first is the sum of various continua at the wavelength, given as νF_ν (see the **print continuum** and **set nFnu** commands in Part 1 for a discussion of what is included in the *nFnu* entry). *nInu* is the transmitted and reflected incident continuum. *InwT* is the total reflected continuum. *InwC* is the reflected incident continuum.

9.2.6   Molecules...

H2dC , is the cooling due to collisional dissociation of H_2.

H2dH , heating by H_2 dissociation by Lyman continuum

H2vH , heating by coll deexcit of vib-excited H_2

H2vC , cooling by coll deexcit of vib-excited H_2

H2 v , line emission by vib-excited H_2

H-FB and H-FF are the free-bound and free-free continua of the H- ion.

H-CT 6563 , H-alpha produce by H- mutual neutralization

H- H 0, H- heating

H-Hc 0, H- heating

H2+ and HEH+ are the cooling due to formation of H_2^+ and HeH^+.

Codh , carbon monoxide photodissociation heating

CO C 12 , C12O16 cooling

CO C 13 , C13O16 cooling

9.2.7   Grains...

Information in this block concerns emission, absorption, heating, and cooling by any grains included in the calculation.

GrGH , gas heating by grain photoionization

GrTH , gas heating by thermionic emissions of grains

GrGC , gas cooling by collisions with grains

GraT , This is the total grain heating by all sources, lines, collisions, incident continuum. If the grain emission is optically thin limit then this is equal to the total intensity in grain emission.

GraI , grain heating by incident continuum

GraL 1216 , grain heating due to destruction of Lα

GraC , grain heating due to collisions with gas

GraD , grain heating due to diffuse fields, may also have grain emission

Grain emission is included in the predicted total emitted continuum. A machine readable form of the continuum can be produced with the **save continuum** command, also described in Part I of this document.

9.2.8   H-like iso-seq...

This block includes all hydrogen-like isoelectronic species. The **atom H-like** command, described in Part 1 of this document, adjusts properties of this sequence.

HFFc 0 , net free-free cooling, nearly cancels with cooling in lte

HFFh 0 , net free-free heating, nearly cancels with cooling in lte

H FF 0 , H brems (free-free) cooling

FF H 0 , total free-free heating

Clin 912 , total collisional cooling due to all hydrogen lines

Hlin 912 , total collisional heating due to all hydrogen lines

Cool 1216 , collisionally excited La cooling

Heat 1216 , collisionally de-excited La heating

Crst 960 , cooling due to n¿2 Lyman lines

Hrst 960 , heating due to n¿2 Lyman lines

Crst 4861 , cooling due to n¿3 Balmer lines

Hrst 4861 , heating due to n¿3 Balmer lines

Crst 0 , cooling due to higher Paschen lines

Hrst 0 , heating due to higher Paschen lines

LA X 1216 , Lα contribution from suprathermal secondaries from ground

Ind2 1216 , Lα produced by induced two photon

Pump 4861 , Hβ produced by continuum pumping in optically thin ld limit

CION 0 , net col ionz-3 body heat collision ionization cooling of hydrogen

3bHt 0 , heating due to 3-body recombination

Strk 1216 , Stark broadening component of line

Dest 1216 , part of line destroyed by background opacities

Fe 2 1216 , part of Lα absorbed by Fe II

Q(H) 4861 is the intensity of Hβ predicted from the total number of ionizing photons, $Q(H^0)$, assuming that each hydrogen-ionizing photon produces one hydrogen atom recombination. This entry will not be included if the **aperture** command is in effect.

Q(H) 1216 indicates the Lα intensity produced if each hydrogen ionizing photon results in one Lα photon in the high density limit (i.e., no two-photon emission). This entry will not be included if the **aperture** command is in effect.

CaBo 4861 These are the "old" case B predictions, as printed in versions 90 and before of the code.

Ca B 6563A The entries starting with "Ca B" are the Case B intensities computed from the actual model ionization and temperature structure, but assuming that Hβ emits with its Case B emissivity.

Next the predicted intensities of all lines of the hydrogenic iso-electronic sequence are given. The lines have labels that identify the species and stage of ionization, such as "H 1", "He 2", "Li 3", "C 6", etc. The entries with a wavelength of zero are the total intensities of the $2s - 1s$ two-photon emission.

9.2.9   He iso-sequence...

Atoms and ions of the helium-like iso-electronic sequence are treated as multi-level atoms. All species and stages of ionization are specified by labels like "He 1", "Li 2", "C 5", etc. A wavelength of zero indicates the two-photon continuum. The $2\,^3P$ term is resolved into three levels. Emission of each line of the $2\,^3S - 2\,^3P$ and $1\,^1S - 2\,^3P$ multiplets is predicted along with the sum of the multiplets with label "TOTL". The **atom He-like** command, described in Part 1 of this document, adjusts properties of this sequence. Further details are given in Bauman et al. (2005), Porter et al. (2005), and Porter and Ferland (2007).

9.2.10   level 1 lines...

In the current version of the code, the lines printed under this title include both the lines that have been moved to the *EmLine* class, but also older lines that are still scalar quantities. This part of the code is still in a state of flux, and this is reflected in the current documentation. The remaining part of this subsection outlines the methods used for most of the heavy element atoms. The method for producing a list of transferred lines, those that have been moved to the `EmLine` class, is described in the section beginning on page 99 below.

These lines have accurate collision strengths and wavelengths. Many are two-level atoms, but some are the result of multi-level atoms. The following is a summary of the general approach.

Li-sequence. Examples include C IV $\lambda 1549$, O VI $\lambda 1034$, and Mg II $\lambda 2798$. A three-level atom, with full treatment of optical depths and collisional excitation, is used. The "TOTL" intensity is the sum of both lines in the doublet, and is followed by the individual intensities of each member.

Be-sequence. Examples include C III] λ 1909, O V] $\lambda 1215$, and Si III] 1895. A four-level atom, solving for populations of the individual ^3P_j states, is used. The first printed intensity is the total intensity of the multiplet (both $j = 0,1$ decays) and this is followed by the intensities of individual lines. The intensity of the permitted $^1P_o - ^1S$ transition is also calculated. Optical depth and collisional effects on both the permitted and intercombination lines are included.

B-sequence. Examples include C II and O IV. The ground term is treated as a two level atom, with optical depth and collisional effects included, when the gas is too cool to excite the UV lines. The $^4P - ^2P_o$ lines are also predicted with a full multi-level atom that resolves fine structure. The "TOTL" intensity printed is the total intensity of the multiplet and is followed by individual lines. Pumping by all level-two lines of the same ion is included as an excitation process.

^{3}**P- ground term**. Examples include such spectra as [O III] and [O I]. The infrared fine structure lines are computed with full treatment of collisional and optical depth effects. A comment is printed at the end of the model if these lines mase or become optically thick. The populations of ^1D and ^1S are computed with a three-level atom. The intensity of the $^1D - ^3P$ transition is only that of the individual line (i.e. 5007), not the doublet.

^{4}**S**^{0} **- ground term**. Examples include [O II] and [S II]. They are treated as a five-level atom. Intensities of all individual lines, as well as co-added multiplets, are given.

9.2.11   Recombination...

These are a set of heavy-element recombination lines that are predicted in the low-density limit assuming that the transitions are optically thin. This consists of all recombination lines of C, N., and O, with coefficients taken from Nussbaumer and Storey (1984) and Pequignot et al. (1991).

These predictions are for optically thin pure recombination. These should be accurate for planetary nebulae and H II regions. They will not be accurate for dense environments where optical depths and collisional effects come into play. These are only included in the output if the electron density is less than 10^8 cm^{-3}, a rough upper limit to the range of validity in the original calculations of the coefficients.

There are several instances where more than one line of an ion will have the same wavelength due to the integer Angstrom format used for wavelengths. The worst case is O V 4953, where three lines of the same multiplet have the same wavelength.

9.2.12 Level 2 lines...

These are resonance lines that use Opacity Project wavelengths, which are generally accurate to about 10%. These lines have g-bar collision strengths, which are not very accurate at all.

9.3 The transferred lines

9.3.1 Save line data output

The group of "transferred lines" includes all those that have been moved to the `EmLine` class.

In older versions of this document a large list of emission lines appeared here. This list is now far too large to include here. Rather, the list can be generated by executing the code with the command **save line data "filename.txt"** included. This will create a file that includes the full set of lines that are predicted. Note that the lines that are output are only those that exist when the code is run. It is possible to make many of the model atoms and molecules as large or small as you like, and the actual lines that exist when the **save** command is entered will be output. The test case `func_lines.in` in the test suite includes this save command and generates a list of all transferred lines in the file `func_lines.lis`.

To generate a line list, set up a calculation with the atoms set to whatever size is desired (see the **atom** command in Part I). Then execute this script with the **save line data** command included (described in Part I). The save output will include the line list. This will include the level 1 ,level 2, H_2, Fe II, CO, and recombination lines, but not the scalar forbidden lines. These are described in a list following this subsection.

This contains several groups of lines. All quantities were evaluated at 10^4 K. The description of the command in Part I of this document explains how to evaluate the quantities at other temperatures.

The ion is the first column of the table. This is in a uniform format, beginning with the two character element symbol and followed by an integer indicating the level of ionization. "C 2" is C^+ or C II. This is followed by the integer wavelength label used to identify the line in the printout. The third column, with the label "WL", is the correct wavelength of the line, with units of microns ("m"), Angstroms ("A"), or cm ("c"). The remaining columns give the statistical weights of the lower and upper levels, the product of the statistical weight and the oscillator strength, and then the transition probability.

The last column is the electron collision strength. Usually these collision strengths are for only the indicated transition, although in some cases (the Be sequence) the value is for the entire multiplet.

9.3.2 Output produced for the transferred lines

Because the lines have a common format within their storage vectors, the output has a common format too. Generally only the total intensity of the transition, the result of the solution of a multi-level atom with all processes included, is printed. The approach used to compute the level populations is described in Part II of HAZY, and includes continuum pumping, destruction by background opacities, and trapping.

The total intensity of the transition is printed in a form like "C 2 1335", with the spectroscopic identification given by the first part, as found in the first column of the table, and the wavelength as indicated by the number in the second column of the table.

In a few cases (for instance, the C IV $\lambda\lambda$1548, 1551 doublet), a total intensity is also derived. In these cases the label "TOTL" will appear together with an average wavelength (1549 in this case). These lines are all explicitly shown in a following section.

It is possible to break out various contributors to the lines with options on the **print line** command, described in Part I of this document and in the following. These contributors are printed following the total intensity.

print line heating An emission line will heat rather than cool the gas if it is radiatively excited but collisionally de-excited. The print out will include this agent, with the label "Heat", when this command is given.

print line collisions The collisional contribution to the lines will be printed, with the label "Coll".

print line pump The contribution to the total line, produced by continuum pumping, is printed with the label "Pump". What is observed? Whether or not this is a net emission process contributing to the observed line intensity depends on the geometry, mainly whether or not continuum source is in the beam. At some velocities within the line profile this can be a net emission process, due to absorption at other velocities. If the continuum source is in the beam and gas covers it, this is not a net emission process, since photons are conserved.

print line inward The inwardly directed part of the total emission is printed with the label "Inwd". This can be greater than half of the line intensity if the line is optically thick since these lines tend to be radiated from the hotter illuminated face of the cloud.

print line optical depths At the end of the calculation the optical depths for all optically thick lines will be printed. This is not done by default since it can be quite long.

9.4   Line identifications

The file `line_labels.txt` in the docs directory of the distribution lists line identifications and a brief description of its origin.

9.5   Hydrogen recombination lines

Table 9.2 gives the strongest lines of the lowest series. All lines have the label "H 1". The wavelength column gives the string as it appears in the printout. "m" indicates a wavelength in μm and A in Angstroms. The Case B intensity is taken from the `limit_caseb_h_hs87` test case.

Some IR lines may have the same wavelength if the default line precision is used. The **set line precision** command can be used to increase the number of significant figures in the line wavelength, which should make identification easier.

Table 9.2: Hydrogen emission lines

Series	$n' \to n$	Wavelength	I(Case B)
Lyman	2–1	1216A	-
Balmer Hα	3–2	6563A	2.8463
Hβ	4–2	4861A	1.0000
Hγ	5–2	4340A	0.4692
Hδ	6–2	4102A	0.2596
Paschen Pα	4–3	1.875m	0.3319
Pβ	5–3	1.282m	0.1617
Pγ	6–3	1.094m	0.0899
Pδ	7–3	1.005m	0.0553
Brackett Brα	5–4	4.051m	0.0778
Brβ	6–4	2.625m	0.0445
Brγ	6–4	2.166m	0.0274
Brδ	6–4	1.945m	0.0180
Pfund	6–5	7.458m	0.0246
Humphreys	7–6	12.37m	0.0098
	8–7	19.06m	0.0042
	9–8	27.80m	0.0020

9.6 Molecular hydrogen lines

Table 9.3 gives some of the stronger or more frequently observed H_2 lines. These are only predicted when the large model H_2 molecule is included with the **atom H2** command. The wavelength column gives the string as it appears in the printout. "m" indicated a wavelength in μm. All lines have the label "H2 ".

Many lines will have the same wavelength if the default line precision is used. The **set line precision** command can be used to increase the number of significant figures in the line wavelength, which should make identification easier.

The following is an example which predicts the emissivity of the H_2 2.121 1-0 S(1) μm line.

```
save line emissivity ``lines.ems''
H2   2.121m
end of lines
```

A list of all H2 lines with their labels and excitation energies can be generated with the command **save H2 lines**. The Joint Astronomy Centre in Hawaii give a summary of H_2 lines.

9.7 CO lines

Rotation transitions within the ground vibration level are included. These are listed in Table 9.4 which gives the symbol "m" to indicate μm.

Table 9.3: Molecular hydrogen emission lines

Transition	v_{hi}, J_{hi}	v_{lo}, J_{lo}	λ(label)	χ(hi, K)
0-0 S(0)	0,2	0,0	28.21m	509.8
0-0 S(1)	0,3	0,1	17.03m	1015.1
0-0 S(2)	0,4	0,2	12.28m	1681.6
0-0 S(3)	0,5	0,3	9.662m	2503.8
0-0 S(4)	0,6	0,4	8.024m	3474.3
0-0 S(5)	0,7	0,5	6.907m	4586.2
0-0 S(6)	0,8	0,6	6.107m	5829.5
1-0 O(2)	1,0	0,2	2.626m	5986.9
1-0 Q(1)	1,1	0,1	2.406m	6149.0
1-0 O(3)	1,1	0,3	2.802m	6149.0
1-0 S(0)	1,2	0,0	2.223m	6471.4
1-0 Q(2)	1,2	0,2	2.413m	6471.4
1-0 O(4)	1,2	0,4	3.003m	6471.4
1-0 S(1)	1,3	0,1	2.121m	6951.3
1-0 Q(3)	1,3	0,3	2.423m	6951.3
1-0 O(5)	1,3	0,5	3.234m	6951.3
0-0 S(7)	0,9	0,7	5.510m	7196.7
1-0 S(2)	1,4	0,2	2.033m	7584.3
1-0 Q(4)	1,4	0,4	2.437m	7584.3
1-0 O(6)	1,4	0,6	3.500m	7584.3
1-0 S(3)	1,5	0,3	1.957m	8365.0
1-0 Q(5)	1,5	0,5	2.454m	8365.0
1-0 O(7)	1,5	0,7	3.806m	8365.0
0-0 S(8)	0,10	0,8	5.051m	8677.0
1-0 S(4)	1,6	0,4	1.891m	9286.3
1-0 Q(6)	1,6	0,6	2.475m	9286.3
0-0 S(9)	0,11	0,9	4.693m	10261
1-0 S(5)	1,7	0,5	1.835m	10341
1-0 S(6)	1,8	0,6	1.787m	11521
1-0 S(7)	1,9	0,7	1. 748m	12817

Table 9.4: CO emission lines

Label	wavelength
CO	2600m
CO	1300m
CO	866.7m
CO	650.1m
CO	520.1m
CO	433.4m
CO	371.5m
CO	325.1m
CO	289.0m
CO	260.2m
CO	236.5m
CO	216.9m
CO	200.2m
CO	185.9m
CO	173.6m
CO	162.8m
CO	153.2m
CO	144.7m
CO	137.2m
CO	130.3m
CO	124.2m
CO	118.5m
CO	113.4m
CO	108.7m
CO	104.4m
CO	100.4m
CO	96.75m
CO	93.32m
CO	90.14m
CO	87.17m
CO	84.39m
CO	81.78m
CO	79.34m
CO	77.04m
CO	74.87m
CO	72.82m
CO	70.89m
CO	69.06m
CO	67.32m
CO	65.67m

Chapter 10

PROBLEMS

10.1 Introduction

The code is designed to be autonomous and self-aware. Nonetheless, if can run into problems. This section describes some of the errors that can cause CLOUDY to stop. Floating point errors should never occur. Several other internal errors, which the code is designed to trap and then complain about, could occur. Finally, it is possible that the code will stop because of convergence problems.

The most important single thing to understand about any calculation is why it stopped and whether this affects the predictions. This is discussed further in the section *Stopping Criteria*

If the calculation aborts it will conclude with a request to post the information on the web site's discussion board—please do—we can't fix it if we don't know it's broken.

Please post any problems on the discussion board on the code's discussion board.

10.2 Thermal stability and temperature convergence

This section describes thermal stability problems, how to identify them, and what to do about them.

10.2.1 Types of thermal maps

Three types of thermal maps, showing the heating or cooling of gas as a function of temperature, can be produced by CLOUDY. Each is the answer to a different question.

Figure 10.1 shows the heating and cooling rates as a function of temperature for a photoionized gas in which the gas kinetic temperature was varied. This figure was produced with the test case func_map.in, one of the standard test cases included in the code distribution. Both the gas density and the flux of ionizing photons were held constant and the kinetic temperature was varied. Only one temperature, the point where the two curves cross, occurs in equilibrium. The func_map.in file uses the **save map** command to determine heating and cooling rates at a variety of temperatures. This is exactly what the code does to determine the equilibrium temperature, so this plot can be useful to find out why the code ran into temperature convergence problems. This is why the command was introduced.
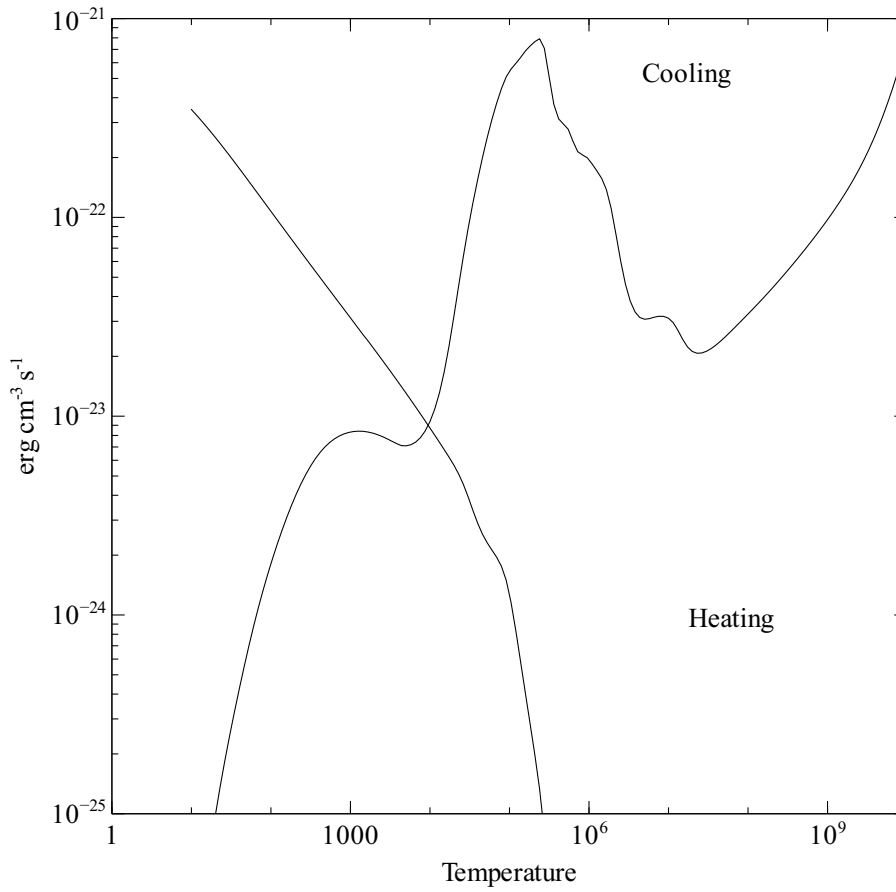
Figure 10.1: A typical heating—cooling function for low density photoionized gas. The cooling and heating rates (erg cm^{-3} s^{-1}) are shown.
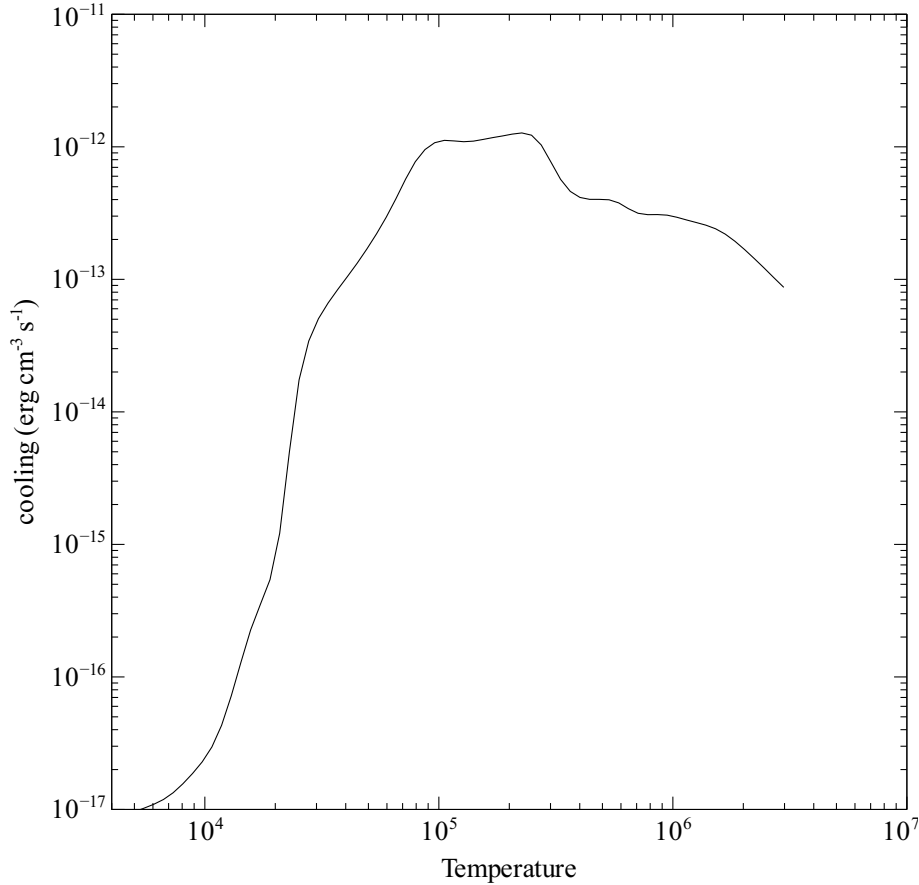
Figure 10.2: A typical cooling function for low density collisionally ionized gas.

Collisionally-ionized gas has a well-defined cooling rate that is only a function of temperature. The sample program `hazy_coolingcurve.cpp` (included in the `programs` directory in the code's distribution) does such a calculation, and Figure 10.2 shows the results. Here the kinetic temperature is set by some physics external to the calculation. The entire ionization solution is valid for each temperature under this assumption. The unspecified heat source would have to provide a local heating rate that is equal to the calculated cooling rate for the solution to be time steady.

The third map is the type of thermal stability map shown by Krolik et al. (1981) and plotted in Figure 10.3. The program that generated these results is given in the file `hazy_kmt.cpp`. Here the equilibrium temperature is determined self-consistently for gas over a wide range of densities, but for a single flux of ionizing photons (or equivalently, distance from the central object).

10.2.2   No Temperature Convergence

A temperature failure occurs when the heating-cooling balance is not within a certain tolerance, set by the **set temperature error** command, after 20 tries. Normally CLOUDY will punt after an excessive number of temperature failures occur. The limit to the number of failures is reset with the **failures** command. If the **failures map** command is entered then the code will first produce a
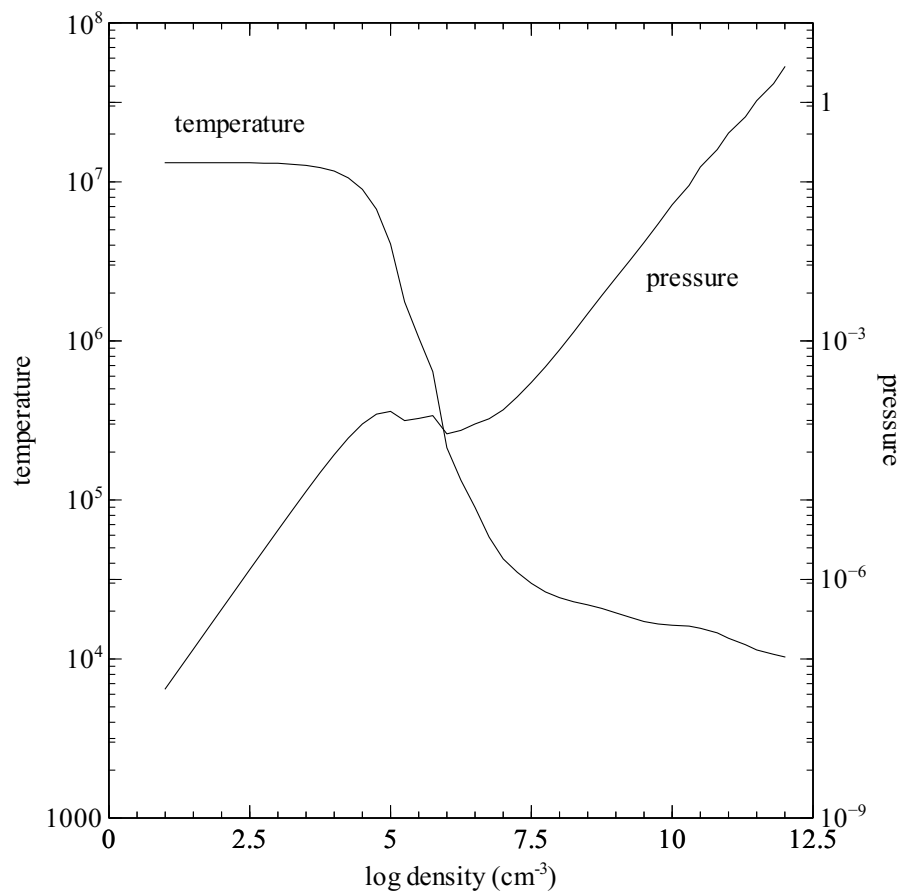
Figure 10.3: Equilibrium temperature as a function of density.

map of heating-cooling space to give an indication of where the equilibrium temperature should have been when excessive failures occur.

Temperature failures most often occur for temperatures in the range 10^2 K to 4×10^3 K, and 10^5 K to 10^6 K. These are where the cooling function permits more that one thermal solution (see, for example, Williams, 1967; Dalgarno and McCray, 1972).

Figure 10.1 shows a typical cooling function for gas in photoionization equilibrium. A peak is reached at a temperature near 10^3 K. This occurs when the fine-structure lines are major coolants. At lower temperatures their cooling rate increases exponentially (as expected), until roughly 10^3 K, when their Boltzmann factors are near unity. Above this temperature their cooling rate is nearly proportional to the Coulomb focusing factor $T^{-1/2}$, and the cooling *decreases* until the temperature is high enough for optical forbidden lines to become important (at roughly 4000 K). A similar phenomenon occurs near the $\sim 10^5$ K to 10^6 K peak in the cooling function.

When failures occur because more than one temperature solution is possible, the reported failures are a physical (not numerical) problem. CLOUDY will try to deal with this by forcing the temperature to values below the peak in the cooling function. Increasing the number of allowed failures (with the **failures** command) to prevent the code from stopping prematurely is permissible as long as the global energy balance is preserved. A warning will be issued at the end of the calculation if the heating-cooling balance is not preserved.

10.2.3 Thermal Stability

The thermal solution may be unstable when the temperature derivative of the net cooling function (cooling minus heating) is negative (Field, 1965). Possibly unstable solutions are indicated by a "u" just before the equilibrium temperature in the zone printout. The temperature derivative is for isochoric (constant density), not isobaric (constant pressure), conditions. Comments are printed at the end of the calculation if possibly unstable thermal solutions are present in the calculation.

10.2.4 Thermal fronts

Just as an ionization front is a region where the level of ionization changes dramatically over a small scale, a thermal front occurs where the temperature changes dramatically over a small scale. This can be caused by a real physical change of state of the gas such as those that occur near the peaks in the cooling curve. An example of a thermal front, taken from Ferland et al. (2002), is shown in Figure 10.4. This type of jump is physical. The gas changes phase and moves to different branches of the cooling curve. The code will generate a caution or comment if the electron temperature changes discontinuously from one zone to the next.

A thermal front can lead to pressure convergence failures when the solution jumps between the high and low temperature branches. Figure 10.5 shows an example case, taken from `orion_hii_pdr_pp.in` in the test suite. This shows the pressure history (output with the **save pressure history** command). The solver adjusts the density trying to make the resulting pressure agree with the desired pressure. The pressure changes continuously with density up to the point where the temperature jumps over the peak in the cooling curve. No solution is possible, and the code announces a pressure failure. In nature the presence of a magnetic field (added with the **magnetic field** command) will cushion the front from large changes in density.
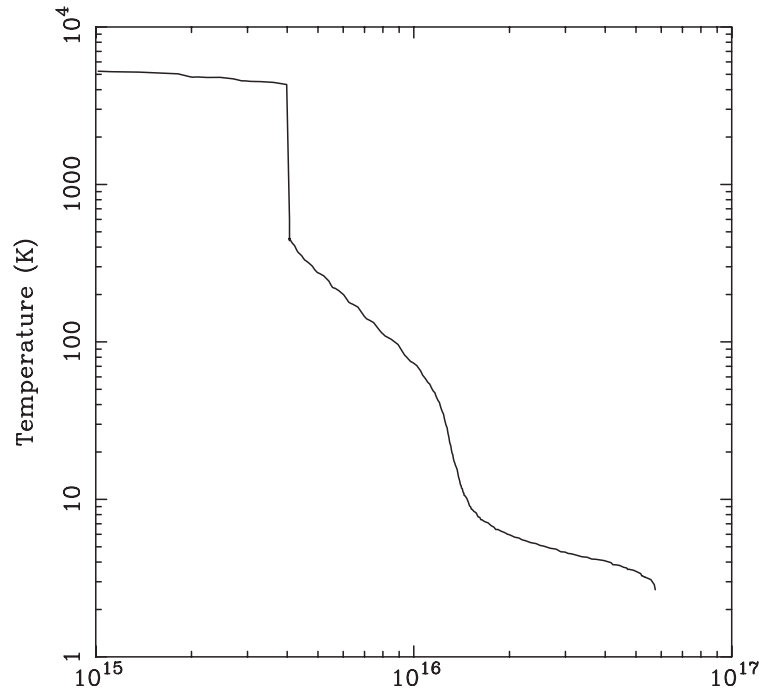
Figure 10.4: An example of a thermal front in a cooling flow cloud (Ferland et al., 2002). The x-axis is the depth into the cloud (cm). The thermal front at $\sim 4 \times 10^{15}$ cm is unresolved.
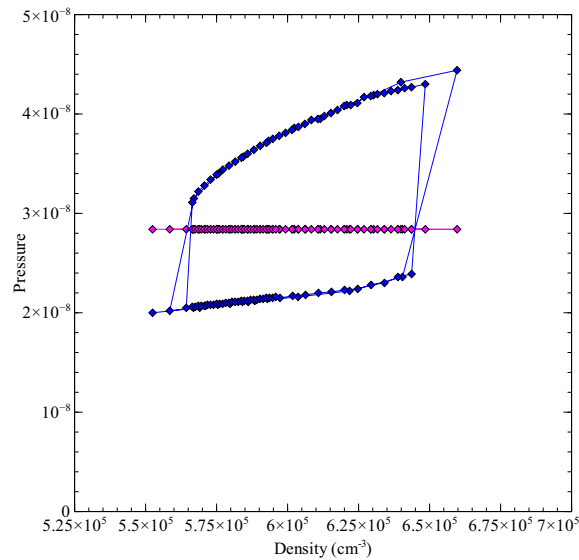
Figure 10.5: A thermal front in a constant pressure simulation. The x-axis gives the density [cm^{-3}] and the y-axis is the pressure [dynes cm^{-2}]. The points forming the large box are the resulting total gas pressure and the horizontal line is the correct pressure. The solution jumps above and below the equilibrium value as the temperature jumps above and below the thermal front.
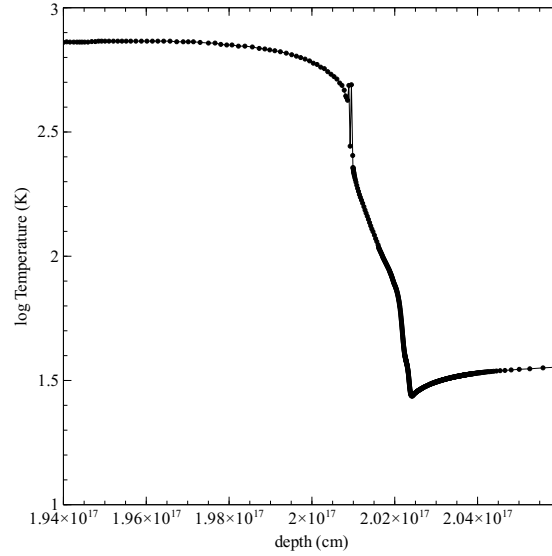
Figure 10.6: A constant-pressure thermal front in a temperature - radius plot. The x-axis gives the radius (cm). The y-axis gives the log of the temperature (K). The solution jumps above and below the equilibrium value, leading to a series of pressure failures, near a depth of 2×10^{17} cm, as it soldiers on through the thermal front.

A serious of pressure failures occur in this simulation when the gas falls to a temperature of ~ 300 K, as shown in Figure 10.6. The code simply presses on with the goal of reaching the cold side of the front.

10.2.5   Map Output

The program stops if an excessive number of temperature failures occur. The default limit is 20. It will produce a map of the heating and cooling as a function of temperature for the last computed zone if the **map** option on the **failures** command is given. The map is described here. The start of the output from the test case func_map.in is shown below.

```
90.02x map of heating vs cooling
 te, heating, cooling.
 Cloudy punts, Te= 9.254E+03 HTOT= 9.123E-24 CTOT= 9.118E-24 nzone=   1
 COOLNG array is
      O   4     25  0.340 O  3   5007  0.182 O  3     88  0.075 H FB      0  0.057 S  4     10
      H ff      0  0.022 S  3     33  0.020 Ne 3     15  0.019 Hefb      0  0.015 N  3     57
      Ne 5     24  0.010 Ne 5    14  0.009 C  3   1910  0.008 Heff      0  0.007 Si 2     34
 Line heating array follows
     Te      Heat------------------> Cool----------------------->     dH/dT     dC/DT
1.0000E+01 3.4774E-22    1    1 0.636 4.6095E-26 H FB     0A 0.723 -8.19E-24  1.56E-27 9.1
1.0209E+01 3.4490E-22    1    1 0.635 4.6814E-26 H FB     0A 0.720 -7.98E-24  1.65E-27 9.1
1.0423E+01 3.4233E-22    1    1 0.635 4.7510E-26 H FB     0A 0.717 -7.74E-24  1.74E-27 9.1
```

The output begins with a listing of the strongest coolants for the last zone. Then the program steps through increasing temperatures and prints the heating, cooling, and ionization of the gas.

From this information it should be possible to determine the temperature where the equilibrium thermal solution should have been. Each solution is completely self-consistent, except that heating and cooling do not balance. Both the local attenuated radiation field and collisional ionization contribute to the ionization balance at each temperature. All processes contribute to the thermal balance, including collisional ionization. The map is at constant density.

The first column gives the temperature. Columns 2 and 6 give the volume heating and cooling. Both have units erg s^{-1} cm^{-3}. Columns 3 and 4 constitute an indication of the main heating source. Columns 7 and 8 give the label and wavelength of the strongest coolant. Columns 5 and 9 give the fraction of the total heating or cooling due to these agents. Columns 10 and 11 give the heating and cooling derivatives. Columns 12 and 13 give the electron and hydrogen densities (cm^{-3}) and the remaining columns give the logs of the hydrogen and helium ionization fractions. The location of the probable thermal solution is indicated by a comment surrounded by dashed lines.

10.3   Convergence problems with dust-free static sphere

Ionization convergence problems can occur with a dust-free static spherical geometry. The default geometry when the **sphere** command is entered is for lines to freely escape after crossing the central hole due to some level of expansion. In a static spherical geometry (set with the **sphere static** command) the total Lα optical depth at the illuminated face of the shell will be very large since the line is scattered by the matter that lies across the entire shell. The line is destroyed when dust is present. If dust is not present the Lα intensity J will become very large. If the total optical depth in Lα is also large then the dominant escape / destruction process for the line will be absorption by atoms of third-row elements or the $n = 2$ level of hydrogen. This can lead to ionization convergence problems due to the extremely large Lα intensity J.

The first question to ask is whether this geometry is appropriate. Dust is nearly always present in the ISM. In dense stellar environments it is unlikely that a spherical geometry will be static. Include dust, use the default expanding spherical geometry, or a wind, and the problem will go away. To the best of my knowledge this geometry does not occur in nature.

10.4   Optical depth convergence problems

The code generally will not converge if it has not done so within ten or so iterations. Convergence problems most commonly occur when the specified column density or thickness is very near a prominent ionization front. In this case very small changes in the physical conditions result in large changes in the optical depths. The code will not have convergence problems if an optical depth is used as a stopping criterion instead.

10.5   Negative populations

It is possible that the code will stop because negative level populations were predicted for atoms, ions, or molecules. This is not supposed to occur, but sometimes happens because of numerical

instabilities in the matrix inversion routine. Please post the input stream and version of CLOUDY on the code's discussion board.

10.6   Floating Point Errors

The code should be compiled and linked with options enabled so that the code will crash on overflow or division by zero, but ignore underflow. The **crash** command described in Part 1 tests this. *Floating point errors should never occur.* The logic within the code is designed to identify problems, and complain, but not fail. The logic is only as good as the tests they were designed to pass. It is inevitable that circumstances will occur for which the logic now in the code is not sufficient. It is possible that the code will fail when these circumstances occur. I would be grateful for reports of any such failures, since they inevitably identify shortcomings in the code, and lead to its improvement. Please post comments on the discussion board on the code's web site.

10.7   We can't fix it if we don't know it's broken

Machines are growing faster far more rapidly than people are getting smarter. Reliability in the face of complexity is the major challenge to the development of any large-scale computer code (Ferland, 2001). There can be little doubt that CLOUDY contains bugs.

   If problems arise or the code crashes then it is likely that you found a problem. We would appreciate learning about such problems since they identify shortcomings which usually lead to improvements in the code (or the documentation). Please post queries and bug reports on the discussion board on the code's discussion board.

Chapter 11

HISTORY AND ACKNOWLEDGEMENTS

11.1   History

CLOUDY was born at the Institute of Astronomy, Cambridge, in August of 1978, in the computing environment described in the web document http://www.nublado.org/gary/computing1970s.htm. Its development has been continued at The University of Kentucky, The Ohio State University, and during extended visits to the Joint Institute for Laboratory Astrophysics, the Royal Greenwich Observatory, IOA Cambridge, Cerro Tololo Interamerican Observatory, and the Canadian Institute for Theoretical Astrophysics.

The code has been through three computer languages. It was originally written in FORTRAN IV and advanced through several dialects, reaching FORTRAN 77 in 1994 (version 84). Version 90 was written in a mix of FORTRAN 77 and MILSPEC extensions. This was the most advanced Fortran that could be used with open source compilers. It moved to ANSI 89 C in 1999 (version 96) and to C++ with the release of 07.02 in 2007.

Moore's Law is due to Gordon Moore, one of the founders of Intel Corporation. He observed that modern CPU's become about twice as powerful every 18 months. This trend has held true for the past twenty years, shows no sign of failing, and seems to be associated with our ability to control complexity. By this standard the growth of CLOUDY has been conservative, in that it is growing slower and complex on the Moore's-law timescale. Figure 11.1 shows the evolution of the code, as indicated by its size as a function of time.[1] As another example, the Meudon 1985 Meeting planetary nebula test (**pn_paris.in** in the test cases) has always taken about one minute to compute.

11.2   Acknowledgments

Comments or suggestions which led to the improvement of CLOUDY were made by the many individuals acknowledged on the web site http://www.nublado.org.

Peter G. Martin and Hagai Netzer had special roles during the early development of the code. Peter added several of the commands that deal with ordering of supplemental line lists and the

---

[1]Before mid-1995 the size was the total number of lines in the distributed source. After 1995 the size only includes the number of lines excluding block data. When the code was converted to C the block data were converted to external data files. These external files are now far larger than the code itself.
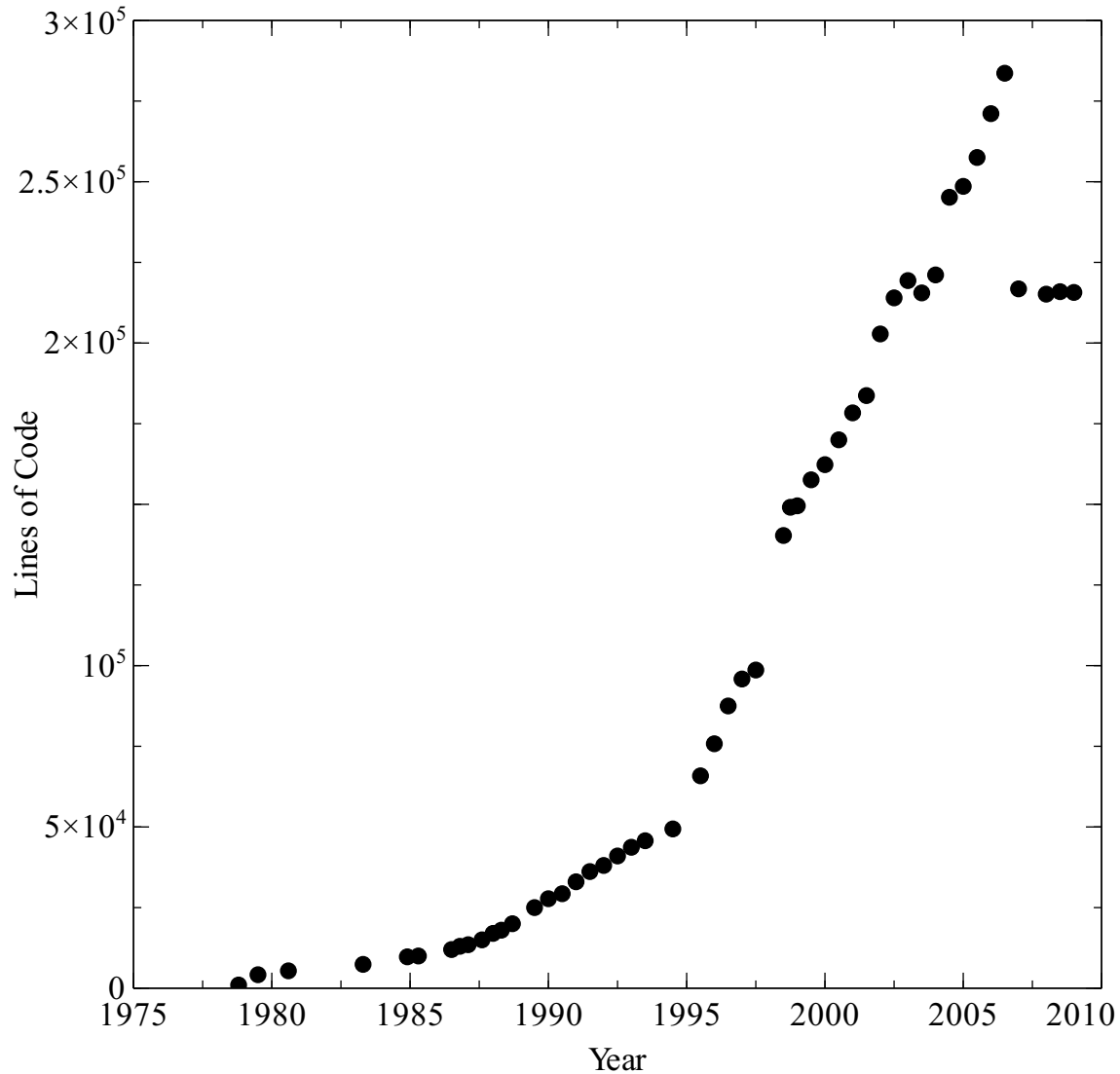
Figure 11.1: The size of the code as a function of time. The code grows roughly 7% larger per year, with growth spurts and slowdowns at times. There are several changes in slope evident - the year and cause are: 1985 - mainframe to Unix; 1993 - Unix to windows; the jump at 1999 - the Fortran to C conversion and the Williams / van Hoof drop in 2006, due to the use of object-like structures and moving converted fortran block data into separate files.

Appendix A

ATOMIC DATA SOURCES

Codes like CLOUDY can only exist because of the large body of work done by the atomic and molecular physics community. This work will only continue to be supported if it is cited in the literature whenever it is used. The following is a partial list of citations for the atomic data used within the code.

This table is generated by the perl script `doc_atomic_data.pl`. This generates the file `doc_atomic_data_refer.txt` which is pasted below.

Appendix B

GLOSSARY OF SYMBOLS

As far as possible, the notation used by HAZY follows standard texts (AGN3; Mihalas, 1978). This is a summary of some of the symbols used.

The fundamental constants used by the code are from the CODATA recommended values. Constants are contained in the header `physconst.h`.

Symbol	Description	Units	Notes
a	Stefan radiation density	erg cm^{-3} K^{-4}	7.56464×10^{-15}
a	damping constant	—	
a_o	Bohr radius	$\hbar/m_e c^2$ cm	$0.5291775 \times 10^{-8}/Z$
A_{rad}	radiative acceleration	cm s^{-2}	
A_{ul}	radiative rate from level u to l	s^{-1}	
b_n	departure coefficient	—	
B	magnetic field	esu	
B_ν	Planck function	erg cm^{-2} s^{-1} Hz^{-1} sr^{-1}	
c	speed of light	cm s^{-1}	2.997925×10^{10}
C	collisional rate	s^{-1}	
C_{ul}	line collision rate	s^{-1}	
D_{ul}	line destruction probability	—	
f	oscillator strength	—	
$f(r)$	filling factor	—	
f_ν	flux density	erg cm^{-2} s^{-1} Hz^{-1}	
F_ν	flux density	erg s^{-1} Hz^{-1}	
g	grain asymmetry factor	—	
g_i	statistical weight	—	
g_{III}	T aver free-free gaunt factor	—	
$g_/$	Solar surface gravity	cm s^{-2}	2.74×10^4
G	gravitational constant	dyne cm^2 g^{-2}	6.673×10^{-8}
G	energy gains, heating	erg cm^{-3} s^{-1}	
h	Planck's constant	erg s	6.6262×10^{-27}
$\hbar$	Planck's constant	erg s	1.0546×10^{-27}
I	integrated intensity	erg s^{-1} sr^{-1} Hz^{-1}	
I_n	ionization potential of level n	erg Ryd	

I_ν	intensity	erg s^{-1} sr^{-1} Hz^{-1}	
J	integrated mean intensity	erg s^{-1} sr^{-1}	
J_ν	mean intensity	erg s^{-1} sr^{-1} Hz^{-1}	
k	Boltzmann constant	eV deg^{-1}	8.6171×10^{-5}
k	Boltzmann constant	erg deg^{-1}	1.38062×10^{-16}
$L_\odot$	luminosity of sun	erg s^{-1}	3.826×10^{33}
m_A	mass of atom A	gm	
m_{AMU}	atomic mass unit	gm	$1.6605402 \times 10^{-24}$
m_e	electron mass	gm	9.10956×10^{-28}
$m_e\,\mathrm{c}^2$	electron energy	Ryd	3.75584×10^4
m_p	proton mass	gm	$1.6726231 \times 10^{-24}$
M_J	Jeans' mass	gm	
$M_\odot$	mass of the sun	gm	1.989×10^{33}
$M_\oplus$	mass of the Earth	gm	5.977×10^{27}
n_e	electron density	cm^{-3}	
n_j	population of level j	cm^{-3}	
n_p	proton density	cm^{-3}	
$n(H)$	total H density, all forms	cm^{-3}	
$n(x)$	density of species x	cm^{-3}	
$n(cr)$	cosmic ray density	cm^{-3}	
n	atom's level	—	
$n(\mathrm{H}_{tot})$	H density, all forms	cm^{-3}	
$N(\mathrm{x})$	column density of species x	cm^{-2}	
$N(\mathrm{H}_{tot})$	total H col den, all forms	cm^{-2}	
N_{eff}	effective H column density	cm^{-2}	
$P*(\mathrm{x})$	LTE relative population	cm^3	
P_{gas}	gas pressure	dyn cm^{-2}	
P_{lines}	line radiation pressure	dyn cm^{-2}	
P_{tot}	total pressure	dyn cm^{-2}	
P_{ul}	line escape probability	—	
$P_{\tau x}(n)$	continuum escape prob	—	
pc	parsec	cm	3.085678×10^{18}
q_{ij}	line collisional rate coefficient	cm^3 s^{-1}	
q_n	collisional rate coefficient	cm^3 s^{-1}	
q_e	electron charge	esu	4.80325×10^{-10}
Q_{abs}	grain absorption efficiency	—	
$Q(\mathrm{H})$	hydrogen ionizing photons	s^{-1}	
r	radius	cm	
$r_{l,u}$	rate	s^{-1}	
R_0	inner radius	cm	
R	total to selective extinction	—	
R_{H}	Rydberg unit for H	—	
R_∞	Rydberg unit for inf mass	—	
R_{AU}	radius of Earth's orbit	cm	1.4959×10^{13}

R_+	radius of the Earth	cm	6.378×10^{18}
$R_\odot$	radius of the sun	cm	6.9599×10^{10}
T_e	electron temperature	cm^{-3}	
$T_{eff}(\odot)$	Sun's effective temperature	K	5770 K
T_{exc}	excitation temperature	K	
T_{color}	color temperature	K	
T_{low}	lowest temp allowed	K	2.8 K
T_u	energy density temperature	K	
u	energy density	erg cm^{-3}	
U_g	grain potential	volt	
u	velocity (mean or projected)	cm s^{-1}	
$\bar{u}$	mean particle speed	cm s^{-1}	
u_{Dop}	Doppler velocity	cm s^{-1}	
u_{exp}	expansion velocity	cm s^{-1}	
u_{th}	thermal velocity	cm s^{-1}	
u_{turb}	turbulent velocity	cm s^{-1}	
V_g	grain potential	eV	
V_n	grain work function	eV	
W	geometric dilution factor	—	
x	relative shift from line center	—	
X_c	continuous to total opacity	—	
$\hat{Y}$	grain photoelectric yield	—	
year		s	3.156×10^7
z	redshift	—	
Z	nuclear charge	—	
α	Fine structure constant	$q_e^2/(\hbar c)$	1/137.036
$\alpha(n,T)$	recombination coefficient	cm^3 s^{-1}	
$\bar{\alpha}(n,T)$	effec recomb coefficient	cm^3 s^{-1}	
α_v	continuous abs cross section	cm^2	
α_{lu}	line absorption cross section	cm^2	
α_β	Case B recomb rate coef	cm^3 s^{-1}	
β	recombination cooling coef	cm^3 s^{-1}	
η_v	photon occupation number	—	
δr	zone thickness	cm	
Δr	depth into cloud	cm	
γ_u, l	continuum pumping probability	—	
Γ_n	photoionization rate	s^{-1}	
Γ	reciprocal lifetime of up level	s^{-1}	
Γ_{OTS}	OTS photoionization rate	s^{-1}	
κ	absorption opacity	cm^{-1}	
κ_{lu}	line absorption opacity	cm^{-1}	
κ_s	continuous scattering opacity	cm^{-1}	
κ_v	continuous absorption opacity	cm^{-1}	
λ_J	Jeans' length	cm	

Λ	energy loss, cooling	erg cm^{-3} s^{-1}	
μ	mean molecular weight	—	
Ω	energy-specific collision strength	—	
Ω	shell coverage	sr	
$\Omega/4\pi$	covering factor	—	
$\Phi(H)$	flux of ionizing photons	cm^{-2} s^{-1}	
$\phi\nu$	photon flux density	cm^{-2} s^{-1} Ryd^{-1}	
ϕ_{OTS}	flux of OTS photons	cm^{-2} s^{-1}	
ρ	mass density	gm cm^{-3}	
πa_o^2	area of first Bohr orbit	cm^2	87.9737×10^{-18}
	Classical electron radius	$q_e^2/(m_e c^2)$ cm	2.818×10^{-13}
σ_T	Thomson cross section	$8\pi/3 \times \left[q_e^2/(m_e c^2)\right]^2$ cm^2	6.6524×10^{-25}
σ_ν	scattering cross section	cm^2	
σ_{Ray}	Rayleigh scat cross section	cm^2	
Σ	projected grain area	cm^2	
τ	optical depth	—	
τ_{abs}	absorption optical depth	—	
τ_{scat}	scattering optical depth	—	
$\tau_{u,l}$	line optical depth	—	
Υ	thermal averaged collision strength	—	
ν	frequency	Hz	
ν_{Ryd}	frequency	Ryd	
δ_ν	line width	Hz	
$\delta\nu_{Dop}$	Doppler width	Hz	
χ_s	hν/kT	—	

Appendix C

CONVERSION FACTORS

Table C.1 gives conversion factors between various common units. The last column of the table gives the variable names for constants that occur within the code. Most are defined as macros within the header file **physconst.h**. These should be used instead of entering the constant directly. In the following all Rydbergs are for infinite mass nuclei.

The fundamental constants used by the code are from the CODATA recommended values and are in the header file physconst.h. Derived quantities should be formed from the fundamental quantities given there, so that any future changes will trickle down into all parts of the code.

Table C.1: Conversion Factors

To convert from	Variable	to	multiply by	Parameter
AU		cm	1.49597870(13)	
Boltzmann constant	BOLTZMANN	1.3806503(-16)		
cm		micron(μm)	10^4	
phot/s/cm^2	flux	f_ν	$\nu_{Ryd}h\nu_1$ (erg)	
phot/Ryd/s/cm^2	flux/widflx	νf_ν	$\nu_{Ryd}^2 h\nu_1$ (erg)	
phot/Ryd/s/cm^2	flux/widflx	J_ν	$\nu_{Ryd} h$	
optical depth	tautot	A_V(mag)	1.08574	
energy (eV)		ergs	1.602192(-12)	
energy (eV)		K	1.1604448(4)	*eVdegK*
energy (keV)		Frequency Hz	2.41799($+17$)	
energy (Ryd)	anu	Kelvin	1.5788866(5)	*Te1ryd*
energy (Ryd)	anu	ergs	2.179874(-11)	*en1ryd*
energy (Ryd)	anu	cm^{-1}	109737.315	*1/WavNRyd*
energy (Ryd)	anu	eV	13.6056981	*evRyd*
energy (Ryd)	anu	Å	911.6/energy(Ryd)	*rydlam*
energy (Ryd), T	anu, Te	hν/kT	1.5788866(5)*anu/Te	*Te1ryd*
temperature (K)	Te	eV	8.617385(-5)	
temperature (K)	Te	ergs	1.38063(-16)	*boltzmann*
temperature (K)	Te	Rydbergs	1/1.5788866(5)	*1/te1ryd*
wavelength (Å)		meters	1(-10)	
wavelength (Å)		ergs	1.9864(-8)/λ(Å)	
wavelength (Å)		degree K	1.43877($+8$)/λ(Å)	
wavelength (cm)	micron	1($+4$)		
wavelength (cm)	Å	1($+8$)		
wavelength (cm)		ergs	1.9864(-12)/λ(cm)	
wavelength (cm)		degree K	1.43877/λ(cm)	
wavelength (cm)		Rydbergs	9.11256(-6)/λ(cm)	
wavelength (micron)		degree K	1.43877($+4$)/λ(μ)	
wavelength (micron)		ergs	1.9864(-12)/λ(μ)	
wavenumbers (cm^{-1})		ergs	1.98648(-16)	
wavenumbers (cm^{-1})		degree K	1.43877	*WavNKelv*
wavenumbers (cm^{-1})		Rydbergs	9.1126732(-6)	*WavNRyd*

Appendix D

THE TEST SUITES

D.1   Introduction

The code must be completely tested every time anything is changed. This is done with the test suite that is included in the distribution. The following pages list the test cases included in the `auto` directory within the test suite directory `tsuite`.

The test suite contains a series of Perl scripts that automate several tasks. The `readme_tests.htm` file included with the tests describes these scripts. The script `doc_tsuite.pl` extracts the test names and the description of each, and creates two files, `doc_tsuite.htm`, a formatted description of each test, and `doc_tsuite.txt`, the table that follows this discussion.

The test cases include a large number of **monitor** commands that allow the code to be automatically tested every night. These **monitor** commands have been removed from the examples below.

The simulations form various classes. The names of the classes and their intention is given in Table .

D.2   Organization of commands

Commands can occur in any order but we have made an effort to enter them in the following order. This is so that they can be studied to see examples of how to use the code.

   c commands controlling continuum
   c commands for density & abundances
   c commands controlling geometry
   c other commands for details
   c commands controlling output As indicated, there are the **print** and **save** commands which control the code's output.
   c commands giving the monitors Each test includes a number of **monitor** commands. These give values of predicted quantities that the code has obtained in the past. Changes to the code may cause predicted values to change, and if they do the **monitor** command will announce it.

Table D.1: Keywords used in the Test Suite.

Class	Function
Blr	Broad emission line region of AGN
Coronal	Collisionally ionized gas with pre-set temperature
Dynamics	Flow
Function	Test various functions of the code
Geometry	Test geometric aspects, aperture command
HII	H II regions
IGM	Intergalactic medium
ISM	Interstellar medium
Limit	Test limiting cases
NLR	Narrow-lined region of AGN
Nova	Aspects of the classical nova explosion
Optimizer	Test the optimizers
PDR	Photodissociation region
PN	Planetary nebulae
Stars	Stellar atmospheres

D.3   The Auto Test Suite

Table D.2 lists the simulations in the `auto` test directory.

Table D.2: The simulations in the auto test suite.

Class	Function
agn_blr_albedo	measure rayleigh scattering of Lya
agn_lex00_u0	intermediate-ionization x-ray ionized cloud from Lexington 2000
agn_lex00_u1	high-ionization x-ray ionized cloud from Lexington 2000
agn_lex00_um1	low-ionization x-ray ionized cloud from Lexington 2000
agn_reflector	model of Compton reflector
agn_S_curve_grid	temperature across Spitzer thermal stability S curve
agn_warm_absorber	simple warm absorber model
aperture_beam_int	test aperture beam command with intensity
aperture_beam_lum	test aperture beam command with luminosity
aperture_slit	test aperture slit command with luminosity
blr_f92	standard blr cloud in Ferland et al. 1992
blr_fp89	final F+P 1989 BLR model table 3
blr_hizqso	high Z quasar cloud
blr_kk81	old blr
blr_level2	test dominant level2 lines
blr_n09_p18	BLR model, density 1e09 cm-3, flux of H-ion phots 1e18 cm2 s-1
blr_n09_p18_Z20	BLR model, density 1e09 cm-3, flux of H-ion phots 1e18 cm2 s-1, Z=20
blr_n09_p20	BLR model, density 1e09 cm-3, flux of H-ion phots 1e20 cm2 s-1
blr_n09_p20_Z20	BLR model, density 1e09 cm-3, flux of H-ion phots 1e20 cm2 s-1, Z=20
blr_n09_p22	BLR model, density 1e09 cm-3, flux of H-ion phots 1e20 cm2 s-1
blr_n09_p22_Z20	BLR model, density 1e09 cm-3, flux of H-ion phots 1e22 cm2 s-1, Z=20
blr_n11_p20	BLR model, density 1e11 cm-3, flux of H-ion phots 1e20 cm2 s-1
blr_n11_p20_Z20	BLR model, density 1e11 cm-3, flux of H-ion phots 1e20 cm2 s-1, Z=20
blr_n12_p19	BLR model, density 1e12 cm-3, flux of H-ion phots 1e19 cm2 s-1
blr_n12_p19_Z20	BLR model, density 1e12 cm-3, flux of H-ion phots 1e19 cm2 s-1, Z=20
blr_n13_p18	BLR model, density 1e13 cm-3, flux of H-ion phots 1e18 cm2 s-1
blr_n13_p18_Z20	BLR model, density 1e13 cm-3, flux of H-ion phots 1e18 cm2 s-1, Z=20
blr_n13_p22	BLR model, density 1e13 cm-3, flux of H-ion phots 1e22 cm2 s-1
blr_n13_p22_Z20	BLR model, density 1e13 cm-3, flux of H-ion phots 1e18 cm2 s-1, Z=20
blr_nf84	early model of blr
blr_nf84_45deg	early model of BLR, with illumination at 45 degree angle
blr_rnfa	table 1 of Rees et al. ApJ 347, 648
blr_rnfb	table 1 of Rees et al. ApJ 347, 648
coll_coronal	model of active region of solar corona
coll_heat_only	test code in limit where ONLY mechanical heating is present
coll_t3	coronal equilibrium at 10^4 K
coll_t4	coronal equilibrium at 10^4 K
coll_t5	coronal equilibrium at 10^5 K
coll_t6	coronal equilibrium at 10^6 K
coll_t7	coronal equilibrium at 10^7 K
dynamics_orion_flow	Orion nebula blister with wind
dynamics_veryfast	very fast wind model
dynamics_veryfast_rec	very fast wind model
dynamics_wind	test of equations of motion in a very highly ionized wind
feii_hin	test feii in high density limit
feii_hirad	feii in case of high radiation density limit
feii_pump	test feii in continuum pumped limit

This HTML file was created by the program doc_tsuite.pl. doc_tsuite.txt contains a tab delimited list of the files.

`agn_blr_albedo` *measure rayleigh scattering of Lya*

```
title measure rayleigh scattering of Lya
c model from Korista, K., & Ferland, G. 1998, ApJ, 495, 672
c
c commands controlling continuum =========
agn 6.683 -1.20 -1.20 -0.90
ionization parameter 1.0
c
c commands for density & abundances =========
init file = "ism.ini"
abundances old solar 84
hden 11.0
c
c commands controlling geometry =========
stop total column density = 23.75
c
c other commands for details =========
iterate
c
c commands controlling output =========
normalize to "FeKa" 1.78
print line faint -1
print diffuse continua
print lines inward
punch continuum last "agn_blr_albedo.con" units kev
punch dr "agn_blr_albedo.dr"
punch reflected continuum last "agn_blr_albedo.ref"
punch emitted continuum last "agn_blr_albedo.emt"
c
c must assert the luminosity since change in this
c line would appear to change the rest of the spectrum
c agn_blr_albedo.in
c class blr
c =======================================
```

This model computes the albedo of a fairly standard BLR cloud. This is the type of model that was presented in the BLR albedo paper by Korista & Ferland, 1998, ApJ 495, 672.

The print diffuse continua command enters continuum fluxs into the emission line stack. The asserts then check that these continua have the expected brightness.

`agn_lex00_u0` *intermediate-ionization x-ray ionized cloud from Lexington 2000*

```
title intermediate-ionization x-ray ionized cloud from Lexington 2000
stop column density 16
print line sort wavelength
print lines column linear
normalise to "H 1" 1216
print line faint -1
```

```
hden 5
iterate
phi(h) 15.477 range 7.353 to 735.3
interpolate (-8 -3)
continue (-6 1.)
continue (-2 14.5)
continue (0.4771 12.7)
continue (0.8663 10.6)
continue (3.8663 7.6)
continue (6 1)
continue (9 -3)
element abundance helium -1
element abundance carbon -3.432
element abundance nitrogen -3.959
element abundance oxygen -3.097
element abundance neon -3.959
element abundance magnesium -4.4318
element abundance silicon -4.456
element abundance sulphur -4.7959
element abundance argon -5.4318
element abundance iron -4.398
element lithium off
element beryllium off
element boron off
element fluorine off
element sodium off
element aluminium off
element phosphrous off
element chlorine off
element potasium off
element calcium off
element scandium off
element titanium off
element vanadium off
element chromium off
element manganese off
element cobalt off
element nickel off
element copper off
element zinc off
c
punch transmitted continuum "agn_lex00_u0.trn" units keV
c
c agn_lex00_u0.in
```

This is one of the "warm absorber" simulations presented at the Lexington 2000 meeting on nebulae. Pequignot et al. summarized in 2001ASPC..247..533P. It is necessary to also include the command no induced processes to obtain the results presented there. This disables UTA ionization, a process that was not included in the calculations presented in the paper.

---

agn_lex00_u1 *high-ionization x-ray ionized cloud from Lexington 2000*

```
title high-ionization x-ray ionized cloud from Lexington 2000
stop column density 16
print lines column linear
normalise to "H 1" 1216
print line faint -1
hden 5
```

```
iterate
phi(h) 16.477 range 7.353 to 735.3
interpolate (-8 -3)
continue (-6 1.)
continue (-2 14.5)
continue (0.4771 12.7)
continue (0.8663 10.6)
continue (3.8663 7.6)
continue (6 1)
continue (9 -3)
element abundance helium -1
element abundance carbon -3.432
element abundance nitrogen -3.959
element abundance oxygen -3.097
element abundance neon -3.959
element abundance magnesium -4.4318
element abundance silicon -4.456
element abundance sulphur -4.7959
element abundance argon -5.4318
element abundance iron -4.398
element lithium off
element beryllium off
element boron off
element fluorine off
element sodium off
element aluminium off
element phosphrous off
element chlorine off
element potasium off
element calcium off
element scandium off
element titanium off
element vanadium off
element chromium off
element manganese off
element cobalt off
element nickel off
element copper off
element zinc off
punch transmitted continuum "agn_lex00_u1.trn" units keV
c
c agn_lex00_u1.in
```

This is one of the "warm absorber" simulations presented at the Lexington 2000 meeting on nebulae. Pequignot et al. summarized in 2001ASPC..247..533P. It is necessary to also include the command no induced processes to obtain the results presented there. This disables UTA ionization, a process that was not included in the calculations presented in the paper.

`agn_lex00_um1` *low-ionization x-ray ionized cloud from Lexington 2000*

```
title low-ionization x-ray ionized cloud from Lexington 2000
stop column density 16
print lines column linear
print lines sort intensity
normalise to "H 1" 1216
print line faint -2
hden 5
iterate
```

```
phi(h) 14.477 range 7.353 to 735.3
interpolate (-8 -3)
continue (-6 1.)
continue (-2 14.5)
continue (0.4771 12.7)
continue (0.8663 10.6)
continue (3.8663 7.6)
continue (6 1)
continue (9 -3)
element abundance helium -1
element abundance carbon -3.432
element abundance nitrogen -3.959
element abundance oxygen -3.097
element abundance neon -3.959
element abundance magnesium -4.4318
element abundance silicon -4.456
element abundance sulphur -4.7959
element abundance argon -5.4318
element abundance iron -4.398
element lithium off
element beryllium off
element boron off
element fluorine off
element sodium off
element aluminium off
element phosphrous off
element chlorine off
element potasium off
element calcium off
element scandium off
element titanium off
element vanadium off
element chromium off
element manganese off
element cobalt off
element nickel off
element copper off
element zinc off
c
punch transmitted continuum "agn_lex00_um1.trn" units keV
c
c agn_lex00_um1.in
```

This is one of the "warm absorber" simulations presented at the Lexington 2000 meeting on nebulae.
Pequignot et al. summarized in 2001ASPC..247..533P. It is necessary to also include the command no
induced processes to obtain the results presented there. This disables UTA ionization, a process that was not
included in the calculations presented in the paper.

---

agn_reflector *model of Compton reflector*

```
title model of Compton reflector
c
c commands controlling continuum =========
c this is a powerlaw that extends into the x-rays, but by default
c has exponential cutoff in FIR
power law -1. hi=1000000000
c the ionization parameter for this powerlaw
ionization parameter 1.
c add background so that FIR/radio does not have zero flux
background
```

```
iterate
c
c commands for density & abundances =========
hden 11
c
c commands controlling geometry =========
stop total column density 24
c
c other commands for details =========
constant temper 5
atom h-like element hydrogen levels resolved 10
atom h-like element helium levels resolved 10
c
c commands controlling output =========
print line faint 1
print line inward
print diffuse continuum
c
c this is to produce lines with lots of contrast with the continuum
set PunchLWidth 100 km/sec
c a fig in part 3 of hazy uses data from this model. the fig shows lines
c with above and below linewidth
c set PunchLWidth c
punch continuum last "agn_reflector.con" units kev
punch continuum reflected last "agn_reflector.alb" units kev
punch overview last "agn_reflector.ovr"
punch dr last "agn_reflector.dr"
c
table lines "LineList_BLR.dat"
c agn_reflector.in
c class blr
c =========================================
c
```

This is a model of the Compton reflector in AGN. It is a constant temperature since models of this region often make that assumption. A plot in Part I of Hazy shows the incident and reflected portions of the continuum. The code will complain that the cloud is Compton thick since it is not really designed to simulate this situation.

`agn_S_curve_grid` *temperature across Spitzer thermal stability S curve*

```
title temperature across Spitzer thermal stability S curve
c
c commands controlling continuum =========
c this is the generic AGN continuum
table agn
c the ionization parameter will be varied
ionization parameter 0.00 vary
grid from 0 to 3 in 0.25 dex steps
c add background so that FIR/radio does not have zero flux
background
iterate
c
c commands for density & abundances =========
c intended to make sim run as fast as possible
hden 8
init file "ism.ini"
element sulphur off
element chlorine off
element argon off
```

```
c
c commands controlling geometry =========
stop zone 1
set dr 0
c
c commands controlling output
c this produces a very large number of lines in the output mostly
c because Hbeta is very weak in hot gas. Choose Lya as the normalization
c line to make line list smaller
normalize to "totl" 1216
print line faint 10 will be varied
c create file with list of grid parameters
punch grid "agn_S_curve_grid.grd" last
c save average H+ temperature
punch average "agn_S_curve_grid.avr" last
temperature hydrogen 1
end of temperatures
c save heating and cooling
punch heating "agn_S_curve_grid.het" last
punch cooling "agn_S_curve_grid.col" last
punch iron element "agn_S_curve_grid.fe" last
punch xspec atable "agn_S_curve_grid.fit"
c
c
c agn_S_grid.in
c class BLR
c =========================================
c
```

This computes a series of models that check the temperature through the S curve in the Fields et al. three-phase model of ISM stability.

`agn_warm_absorber` *simple warm absorber model*

```
title simple warm absorber model
c
c commands controlling continuum =========
table power law
ionization parameter 0
c
c commands for density & abundances =========
hden 6
abundances old solar 84
c
c commands controlling geometry =========
stop column density 22
c
c other commands for details =========
iterate to convergence
c
c commands controlling output =========
print line faint 10
plot continuum range 1 1000
punch continuum "agn_warm_absorber.con" last units kev
punch continuum "agn_warm_absorber.conA" last units Angstroms
punch fine continuum "agn_warm_absorber.finA" last units Angstroms range 61 120 merge 3
punch dr "agn_warm_absorber.dr" last
c this will be the default units, like the printout
punch line optical depths last "agn_warm_absorber_lab.lin"
c this will be the same, in rygbergs
```

```
punch line optical depths last "agn_warm_absorber_ryg.lin" units rydberg
c
c warrm_absorber
c class nlr
c ========================================
c
```

this is a simple warm absorber model. It makes a plot of the transmitted continuum, and generates a list of
lines with significant optical depths

`aperture_beam_int` *test aperture beam command with intensity*

```
title test aperture beam command with intensity
c
c commands for density & abundances =========
hden 0
init file "honly.ini"
c
c other commands for details =========
constant temper 4
c
c commands controlling continuum =========
laser 2
ionization -2
c
c commands controlling geometry =========
stop thickness 10
sphere
aperture beam
c
c commands controlling output =========
c
c aperture_beam_int.in
c class geometry
c ========================================
```

This is a homogeneous sphere that is especially simple. The model is a test of the aperture command, a
command that simulates observing part of an extended object. In this case the aperture is a beam contered on
the center of the nebula, with a line of sight extending through the object.

The code carries along a dummy emission line ("Unit 1") with a constant intensity of 1e-10 erg cm-3 s-1. The
line goes through all of the code's infrastructure, and when the calculation is complete, the program confirms
that the "luminosity" of the line is the emitting volume times 1e-10. The aperture command is verified by
asserting that the emission line has the correct "luminosity". In this case the inner radius is not specified so the
returned value is unity.

`aperture_beam_lum` *test aperture beam command with luminosity*

```
title test aperture beam command with luminosity
c
c commands controlling continuum =========
laser 2
q(h) 31.5
c
```

```
c commands for density & abundances =========
hden 0
init file "honly.ini"
c
c commands controlling geometry =========
stop thickness 10
radius 9
sphere
aperture beam
c
c other commands for details =========
constant temper 4
c
c commands controlling output =========
c
c aperture_beam_lum.in
c class geometry
c =======================================
```

This is a homogeneous sphere that is especially simple. The model is a test of the aperture command, a command that simulates observing part of an extended object. In this case the aperture is a beam contered on the center of the nebula, with a line of sight extending through the object.

The code carries along a dummy emission line ("Unit 1") with a constant intensity of 1e-10 erg cm-3 s-1. The line goes through all of the code's infrastructure, and when the calculation is complete, the program confirms that the "luminosity" of the line is the emitting volume times 1e-10. The aperture command is verified by asserting that this emission line has the correct "luminosity".

`aperture_slit` *test aperture slit command with luminosity*

```
title test aperture slit command with luminosity
c
c commands controlling continuum =========
laser 2
q(h) 31.5
c
c commands for density & abundances =========
hden 0
init file "honly.ini"
c
c commands controlling geometry =========
radius 9 10
sphere
aperture slit
c
c other commands for details =========
constant temper 4
c
c commands controlling output =========
c
c aperture_slit.in
c class geometry
c =======================================
c
```

This is a homogeneous sphere that is especially simple. The model is a test of the aperture command, a command that simulates observing part of an extended object. In this case the aperture is a long slit contered on the center of the nebula, extending beyond the outer reaches of the matter.

The code carries along a dummy emission line ("Unit 1") with a constant intensity of 1e-10 erg cm-3 s-1. The line goes through all of the code's infrastructure, and when the calculation is complete, the program confirms that the "luminosity" of the line is the emitting volume times 1e-10. The aperture command is verified by asserting that the emission line has the correct "luminosity".

`blr_f92` *standard blr cloud in Ferland et al. 1992*

```
title standard blr cloud in Ferland et al. 1992
c
c commands controlling continuum =========
table agn
ionization parameter -1
c
c commands for density & abundances =========
hden 11
abundances old solar 84
c
c commands controlling geometry =========
stop column density 25
c
c other commands for details =========
atom h-like element hydrogen levels resolved 10
atom h-like element helium levels resolved 10
iterate to convergence
c
c commands controlling output =========
print h-like departure hydrogen
print h-like populations hydrogen
print h-like departure helium
print h-like populations helium
plot continuum
normalize to "H 1" 1216 = 105.90
print line faint -2
punch overview last "blr_f92.ovr"
punch convergence reason "blr_f92.cvr"
punch element nitrogen last "blr_f92.nit"
punch element calcium last "blr_f92.ca"
punch dr "blr_f92.dr"
punch transmitted continuum "blr_f92.trn" last
c
c blr_f92.ini
c class blr
c =======================================
```

This is similar to one of the BLR models presented in Ferland et al. (1992) for the well-studied Seyfert galaxy NGC 5548. It has a very large column density and is marginally optically thick to electron scattering. The spectrum is given relative to Lya, and the intensity of this line is reset to produce a spectrum that is on the same intensity scale as that paper.

`blr_fp89` *final F+P 1989 BLR model table 3*

```
title final F+P 1989 BLR model table 3
c
c commands controlling continuum =========
table agn
```

```
ionization parameter -0.5
c
c commands for density & abundances =========
hden 9.5
init file="c84.ini"
abundances old solar 84
c
c commands controlling geometry =========
stop column density 25.5
c
c other commands for details =========
iterate to convergence
failures 2
c
c commands controlling output =========
c set a relatively high faint level since lines relative to Lya=100
print line faint -1
print h-like departure coefficients
plot continuum
normalize to "H 1" 1216 = 85.60
punch heating "blr_fp89.het"
punch convergence reason "blr_fp89.cvr"
punch cooling "blr_fp89.col"
punch overview last "blr_fp89.ovr"
punch element nitrogen last "blr_fp89.nit"
punch results last "blr_fp89.rlt"
punch dr "blr_fp89.dr"
c
c blr_fp89.in
c class blr
c =========================================
```

Ferland and Persson (1989) presented this calculation of a BLR cloud. The differences between the present predictions and those given by FP are largely due to improved treatment of Balmer line escape and destruction. The spectrum is given relative to a Lya intensity of 100. The column density is VERY large, to reproduce intensities of low-ionization lines, especially the Ca II lines.

---

blr_hizqso *high Z quasar cloud*

```
title high Z quasar cloud
c
c commands controlling continuum =========
table agn
ionization parameter 1
c
c commands for density & abundances =========
c actual elec den will be about 100x larger
hden 8
abundances starburst 35
c
c commands controlling geometry =========
c this is more gramage than it appears because of high Z
stop column density 21
c
c other commands for details =========
c once elec den fail occurs due to tripping over He+ - He I-front
c code does recover so not a problem. The default for this is 10,
c and is set low because (hopefully) this would never occur
c hydrogen is a minor constituent in this high metallicity gas
c the electron density is well above the hydrogen density
```

```
failures 3
iterate 3
c
c commands controlling output =========
normalize "totl" 1909
punch dr "blr_hizqso.dr"
punch convergence reason "blr_hizqso.cvr"
punch overview last "blr_hizqso.ovr"
punch element nitrogen last "blr_hizqso.nit"
punch element chlorine last "blr_hizqso.cl"
punch element iron last "blr_hizqso.fe"
punch element neon last "blr_hizqso.ne"
punch element calcium last "blr_hizqso.ca"
punch element zinc last "blr_hizqso.zn"
punch heating last "blr_hizqso.het"
punch cooling last "blr_hizqso.col"
c
c
c blr_hizqso.in
c class blr
c =========================================
```

This is a model of a very high metallicty BLR cloud. It checks the intensities of some of the brigher lines, and is a check that the code can converge a cloud with this high Z.

Secondary ionization is very important when H is highly ionized, due to very high He abundance. Sec ionization becomes important at the He+ - He ionization front, where H+/H is 1e-5.

---

blr_kk81 *old blr*

```
title old blr
c
c commands controlling continuum =========
interpolate (0 -5) (.05 -5) (.1 0) (1 -0.5) (7.353 -2.233)
continue (735 -3.233) (800 -15) (8.e6 -15)
f(nu) -7.32148
constant gas pressure
c
c commands for density & abundances =========
hden 9.60206
init file="c84.ini"
abundances he-1 c-3.699 n-4 o-3.1549 ne-4 na=-8 mg-4.5229
continue al-10 si-4.4229 s-10 ar-10 ca-10 fe-4.5229 ni=-8
c
c commands controlling geometry =========
stop column density 23
c
c other commands for details =========
failures 2
iterate to convergence
c
c commands controlling output =========
print line faint -1
print ages
normalise "H 1" 1216 100
plot continuum range -3
punch overview last "blr_kk81.ovr"
punch dr last "blr_kk81.dr"
punch continuum last "blr_kk81.con"
punch ionizing continuum "blr_kk81.ion"
```

```
c
c
c blr_kk81.in
c class blr
c =======================================
c
```

This is the "standard" BLR model presented by Kwan and Krolik (1981).

>>refer blr cloud Kwan, J., & Krolik, J. 1981, ApJ, 250, 478

Compare line intensities to previous versions of CLOUDY by entering into table on page Error! Bookmark not defined..

The code caution that the resulting total pressure was not constant is to be expected. The KK calculation assumed constant gas pressure, but internally generature line radiation pressure is significant. Because of this the sum of gas plus radiation pressure was not constant although the gas pressure was.

`blr_level2` *test dominant level2 lines*

```
title test dominant level2 lines
c
c commands controlling continuum =========
table agn
ionization -2
c
c commands controlling geometry =========
stop column density 19
c
c commands for density & abundances =========
hden 10
element phosphorus abundance 0
c
c other commands for details =========
iterate
c
c commands controlling output =========
punch overview "blr_level2.ovr"
punch dr "blr_level2.dr"
c
c blr_level2.in
c class blr
c =======================================
```

This model checks predictions for the "level2" lines. These are lines that are normally very weak, have Opacity Project wavelengths, and g-bar collision strengths. Phosphorus is given a large abundance so that its level2 lines are significant.

`blr_n09_p18` *BLR model, density 1e09 cm-3, flux of H-ion phots 1e18 cm2 s-1*

```
title BLR model, density 1e09 cm-3, flux of H-ion phots 1e18 cm2 s-1
c
```

```
c commands controlling continuum =========
table agn
phi(h) 18
c
c commands for density & abundances =========
hden 9
c
c commands controlling geometry =========
stop column density 23
c
c other commands for details =========
init "c84.ini"
iterate convergence
c these are to try to speed things up
no level2 lines
c
c commands controlling output =========
normalize to "totl" 1216 100
punch overview "blr_n09_p18.ovr" last
punch dr "blr_n09_p18.dr" last
punch convergence reason "blr_n09_p18.cvr"
c
c blr_n09_p18.in
c class blr
c =========================================
```

This is one of the 5 models that sample the LOC plane.

---

blr_n09_p18_Z20 *BLR model, density 1e09 cm-3, flux of H-ion phots 1e18 cm2 s-1, Z=20*

```
title BLR model, density 1e09 cm-3, flux of H-ion phots 1e18 cm2 s-1, Z=20
c
c commands controlling continuum =========
table agn
phi(h) 18
c
c commands for density & abundances =========
hden 9
abundances starburst 20
c
c commands controlling geometry =========
stop column density 23
c
c other commands for details =========
iterate to convergence
c these are to try to speed things up
init "c84.ini"
no level2 lines
c
c commands controlling output =========
normalize to "totl" 1216 100
print h-like hydrogen departure coefficients
punch overview "blr_n09_p18_Z20.ovr"
punch dr "blr_n09_p18_Z20.dr"
punch convergence reason "blr_n09_p18_Z20.cvr"
punch convergence error "blr_n09_p18_Z20.cve"
c
c blr_n09_p18_Z20.in
c class blr
```

```
c =========================================
```

This is one of the 5 models that sample the LOC plane.

`blr_n09_p20` *BLR model, density 1e09 cm-3, flux of H-ion phots 1e20 cm2 s-1*

```
title BLR model, density 1e09 cm-3, flux of H-ion phots 1e20 cm2 s-1
c
c commands controlling continuum =========
table agn
phi(h) 20
c
c commands for density & abundances =========
hden 9
c
c commands controlling geometry =========
stop column density 23
c
c other commands for details =========
init "c84.ini"
iterate convergence
c these are to try to speed things up
no level2 lines
c
c commands controlling output =========
normalize to "totl" 1216 100
punch overview "blr_n09_p18.ovr" last
punch dr "blr_n09_p18.dr" last
punch convergence reason "blr_n09_p18.cvr"
c
c blr_n09_p18.in
c class blr
c =========================================
```

This is one of the 5 models that sample the LOC plane.

`blr_n09_p20_z20` *BLR model, density 1e09 cm-3, flux of H-ion phots 1e20 cm2 s-1, Z=20*

```
title BLR model, density 1e09 cm-3, flux of H-ion phots 1e20 cm2 s-1, Z=20
c
c commands controlling continuum =========
table agn
phi(h) 20
c
c commands for density & abundances =========
abundances starburst 20
hden 9
c
c commands controlling geometry =========
stop column density 23
c
c other commands for details =========
init "c84.ini"
```

```
iterate convergence
c these are to try to speed things up
no level2 lines
c
c commands controlling output =========
normalize to "totl" 1216 100
punch overview "blr_n09_p20_Z20.ovr" last
punch dr "blr_n09_p20_Z20.dr" last
punch convergence reason "blr_n09_p20_Z20.cvr"
c
c blr_n09_p20_Z20.in
c class blr
c ========================================
```

This is one of the 5 models that sample the LOC plane.

This simulation is optically thin in the Lyman continuum - no H ionization front is present. As a result it can be difficult to converge.

`blr_n09_p22` *BLR model, density 1e09 cm-3, flux of H-ion phots 1e20 cm2 s-1*

```
title BLR model, density 1e09 cm-3, flux of H-ion phots 1e20 cm2 s-1
c
c commands controlling continuum =========
table agn
phi(h) 22
c
c commands for density & abundances =========
hden 9
c
c commands controlling geometry =========
stop column density 23
c
c other commands for details =========
init "c84.ini"
iterate convergence
c these are to try to speed things up
no level2 lines
c
c commands controlling output =========
normalize to "totl" 1216 100
punch overview "blr_n09_p22.ovr" last
punch dr "blr_n09_p22.dr" last
punch convergence reason "blr_n09_p22.cvr"
c
c blr_n09_p22.in
c class blr
c ========================================
```

This is one of the models that sample the LOC plane.

`blr_n09_p22_Z20` *BLR model, density 1e09 cm-3, flux of H-ion phots 1e22 cm2 s-1, Z=20*

```
title BLR model, density 1e09 cm-3, flux of H-ion phots 1e22 cm2 s-1, Z=20
c
c commands controlling continuum =========
table agn
phi(h) 22
c
c commands for density & abundances =========
abundances starburst 20
hden 9
c
c commands controlling geometry =========
stop column density 23
c
c other commands for details =========
init "c84.ini"
iterate convergence
c these are to try to speed things up
no level2 lines
c
c commands controlling output =========
normalize to "totl" 1216 100
punch overview "blr_n09_p22_Z20.ovr" last
punch dr "blr_n09_p22_Z20.dr" last
punch convergence reason "blr_n09_p22_Z20.cvr"
c
c blr_n09_p22_Z20.in
c class blr
c =======================================
```

This is one of the models that sample the LOC plane.

`blr_n11_p20` *BLR model, density 1e11 cm-3, flux of H-ion phots 1e20 cm2 s-1*

```
title BLR model, density 1e11 cm-3, flux of H-ion phots 1e20 cm2 s-1
c
c commands controlling continuum =========
table agn
phi(h) 20
c
c commands for density & abundances =========
hden 11
c
c commands controlling geometry =========
stop column density 23
c
c other commands for details =========
iterate convergence
c these are to try to speed things up
init "c84.ini"
no level2 lines
c
c commands controlling output =========
normalize to "totl" 1216 100
punch convergence reason "blr_n11_p20.cvr"
punch overview "blr_n11_p20.ovr" last
punch dr "blr_n11_p20.dr" last
c
c blr_n11_p20.in
```

```
c class blr
c =======================================
c
```

This is one of the 5 models that sample the LOC plane.

---

blr_n11_p20_Z20 *BLR model, density 1e11 cm-3, flux of H-ion phots 1e20 cm2 s-1, Z=20*

```
title BLR model, density 1e11 cm-3, flux of H-ion phots 1e20 cm2 s-1, Z=20
c
c commands controlling continuum =========
table agn
phi(h) 20
c
c commands for density & abundances =========
hden 11
abundances starburst 20
c
c commands controlling geometry =========
stop column density 23
c
c other commands for details =========
iterate convergence
c these are to try to speed things up
init "c84.ini"
no level2 lines
c
c commands controlling output =========
normalize to "totl" 1216 100
punch cooling "blr_n11_p20_Z20.col"
punch heating "blr_n11_p20_Z20.het"
punch element calcium "blr_n11_p20_Z20.cal"
punch overview "blr_n11_p20_Z20.ovr"
punch dr "blr_n11_p20_Z20.dr"
punch temperature history "blr_n11_p20_Z20.tem"
c
c blr_n11_p20_Z20.in
c class blr
c =======================================
```

This is one of the 5 models that sample the LOC plane.

---

blr_n12_p19 *BLR model, density 1e12 cm-3, flux of H-ion phots 1e19 cm2 s-1*

```
title BLR model, density 1e12 cm-3, flux of H-ion phots 1e19 cm2 s-1
c
c commands for density & abundances =========
c
c commands controlling continuum =========
table agn
phi(h) 19
c
c commands controlling geometry =========
```

```
hden 12
stop column density 23
c
c other commands for details =========
iterate convergence
c these are to try to speed things up
init "c84.ini"
no level2 lines
c
c commands controlling output =========
normalize to "totl" 1216 100
punch overview "blr_n12_p19.ovr" last
punch dr "blr_n12_p19.dr" last
c
c
c blr_n12_p19.in
c class blr
c =======================================
```

This is one of the 5 models that sample the LOC plane.

`blr_n12_p19_Z20` *BLR model, density 1e12 cm-3, flux of H-ion phots 1e19 cm2 s-1, Z=20*

```
title BLR model, density 1e12 cm-3, flux of H-ion phots 1e19 cm2 s-1, Z=20
c
c commands controlling continuum =========
table agn
phi(h) 19
c
c commands for density & abundances =========
hden 12
abundances starburst 20
c
c commands controlling geometry =========
stop column density 23
c
c other commands for details =========
failures 5
iterate convergence
c these are to try to speed things up
init "c84.ini"
no level2 lines
c
c commands controlling output =========
normalize to "totl" 1216 100
punch convergence reason "blr_n12_p19_Z20.cvr"
punch overview "blr_n12_p19_Z20.ovr" last
punch dr "blr_n12_p19_Z20.dr" last
c
c blr_n12_p19_Z20.in
c class blr
c =======================================
```

This is one of the 5 models that sample the LOC plane.

`blr_n13_p18` *BLR model, density 1e13 cm-3, flux of H-ion phots 1e18*

cm2 s-1

```
title BLR model, density 1e13 cm-3, flux of H-ion phots 1e18 cm2 s-1
c
c this is a very low ionization cloud
c the conditions, and some lines, are surprisingly sensitive
c to the treatment of hydrogen molecules
c
c commands controlling continuum =========
table agn
phi(h) 18
c
c commands for density & abundances =========
hden 13
c
c commands controlling geometry =========
stop column density 23
c
c other commands for details =========
iterate to convergence
c
c commands controlling output =========
normalize to "totl" 1216 100
init "c84.ini"
no level2 lines
punch overview "blr_n13_p18.ovr" last
punch dr "blr_n13_p18.dr" last
punch convergence reason "blr_n13_p18.cvr"
c
c blr_n13_p18.in
c class blr
c ========================================
```

This is one of the 5 models that sample the LOC plane.

`blr_n13_p18_z20` *BLR model, density 1e13 cm-3, flux of H-ion phots 1e18 cm2 s-1, Z=20*

```
title BLR model, density 1e13 cm-3, flux of H-ion phots 1e18 cm2 s-1, Z=20
c
c this is a very low ionization cloud
c the conditions, and some lines, are surprisingly sensitive
c to the treatment of hydrogen molecules
c
c commands controlling continuum =========
phi(h) 18
table agn
c
c commands for density & abundances =========
hden 13
abundances starburst 20
c
c commands controlling geometry =========
stop column density 23
c
c other commands for details =========
c this sim needs roughly 10 iterations to converge
```

```
iterate convergence
c these are to try to speed things up
init "c84.ini"
no level2 lines
c abort if any failures occur - we want to pass this test
failures 1
c
c commands controlling output =========
normalize to "totl" 1216 100
punch overview "blr_n13_p18_Z20.ovr" last
punch overview "blr_n13_p18_Z20.ovr1"
punch dr "blr_n13_p18_Z20.dr" last
punch convergence reason "blr_n13_p18_Z20.cvr"
c
c
c blr_n13_p18_Z20.in
c class blr
c ========================================
```

This is one of the 5 models that sample the LOC plane.

`blr_n13_p22` *BLR model, density 1e13 cm-3, flux of H-ion phots 1e22 cm2 s-1*

```
title BLR model, density 1e13 cm-3, flux of H-ion phots 1e22 cm2 s-1
c
c commands controlling continuum =========
table agn
phi(h) 22
c
c commands for density & abundances =========
hden 13
c
c commands controlling geometry =========
stop column density 23
c
c other commands for details =========
iterate convergence
c these are to try to speed things up
init "c84.ini"
no level2 lines
c
c commands controlling output =========
print h-like departure hydrogen
print h-like departure helium
normalize to "totl" 1216 100
punch overview "blr_n13_p22.ovr" last
punch dr "blr_n13_p22.dr" last
c
c blr_n13_p22.in
c class blr
c ========================================
```

This is one of the 5 models that sample the LOC plane.

`blr_n13_p22_Z20` *BLR model, density 1e13 cm-3, flux of H-ion phots*

1e18 cm2 s-1, Z=20

```
title BLR model, density 1e13 cm-3, flux of H-ion phots 1e18 cm2 s-1, Z=20
c
c commands controlling continuum =========
table agn
phi(h) 22
c
c commands for density & abundances =========
hden 13
abundances starburst 20
c
c commands controlling geometry =========
stop column density 23
c
c other commands for details =========
iterate convergence
c these are to try to speed things up
init "c84.ini"
no level2 lines
c
c commands controlling output =========
normalize to "totl" 1216 100
punch overview "blr_n13_p22_Z20.ovr"
punch dr "blr_n13_p22_Z20.dr"
punch heating "blr_n13_p22_Z20.het"
punch cooling "blr_n13_p22_Z20.col"
c
c blr_n13_p22_Z20.in
c class blr
c =======================================
```

This is one of the 5 models that sample the LOC plane.

`blr_nf84` *early model of blr*

```
title early model of blr
c
c commands controlling continuum =========
ionization parameter -1.92
power law -1 100 0.01
c
c commands for density & abundances =========
hden 9.5
init, file="ism.ini"
abundances cameron
stop lyman optical 6
c
c commands controlling geometry =========
c
c other commands for details =========
constant gas pressure
iterate to convergence
c
c commands controlling output =========
print diffuse continuum
plot continuum
punch overview last "blr_nf84.ovr"
```

```
punch dr last "blr_nf84.dr"
punch results last "blr_nf84.rlt"
c
table lines "LineList_strong.dat"
c blr_nf84.in
c class blr
c =======================================
c
```

This is an example of a "conventional" BLR calculation. The parameters are similar to those of Table 1 of
Netzer and Ferland (1984). Notice that the ratio of Lyalpha to Hbeta ratio is much larger than observed.

>>refer blr model Netzer, H., & Ferland, G. J. 1984, PASP, 96, 593

`blr_nf84_45deg` *early model of BLR, with illumination at 45 degree angle*

```
title early model of BLR, with illumination at 45 degree angle
c
c commands controlling continuum =========
ionization parameter -1.92
power law -1 100 0.01
c this continuum is a beam coming in at 45 degrees
illuminate 45 degrees
c
c commands for density & abundances =========
hden 9.5
abundances cameron
stop lyman optical 6
c
c commands controlling geometry =========
c
c other commands for details =========
init file "ism.ini"
constant gas pressure
iterate to convergence
c
c commands controlling output =========
print diffuse continuum
plot continuum
punch overview "blr_nf84_45deg.ovr"
punch dr "blr_nf84_45deg.dr"
punch results last "blr_nf84_45deg.rlt"
c
c blr_nf84_45deg.in
c class blr
c =======================================
c
```

This is an example of a "conventional" BLR calculation. The parameters are similar to those of Table 1 of
Netzer and Ferland (1984). Notice that the ratio of Lyalpha to Hbeta ratio is much larger than observed.

>>refer blr model Netzer, H., & Ferland, G. J. 1984, PASP, 96, 593

`blr_rnfa` *table 1 of Rees et al. ApJ 347, 648*

```
title table 1 of Rees et al. ApJ 347, 648
c
c commands controlling continuum =========
table agn
ionization parameter -2
c
c commands for density & abundances =========
hden 10
init file="c84.ini"
abundances he -1 c -3.328 n -4.0088 o -3.0809 ne -4 na -20
continue mg -4.3768 al -5.5686 si -4.36653 s -4.76955
continue a -5.4202 ca -5.6383 fe -4.4815 ni -20
c
c commands controlling geometry =========
stop column density 23
c
c other commands for details =========
iterate to convergence
c
c commands controlling output =========
normalise "H 1" 1216 = 100
print line faint -1
punch overview last "blr_rnfa.ovr"
punch continuum last "blr_rnfa.con" units microns
punch transmitted continuum last "blr_rnfa.trn" units rydbergs
punch element nitrogen last "blr_rnfa.nit"
punch dr last "blr_rnfa.dr"
punch results last "blr_rnfa.rlt"
c
c blr_rnfa.in
c class blr
c =========================================
c
```

This is the lower density cloud computed in Rees et al. (1989). Table 1 of that paper lists the predictions, which were a mean of those of Hagai Netzer's ION and roughly version 76 of CLOUDY. The lines are generally still in good agreement with the predictions of that paper. In particular the changes in the line fluxes shown in Figure 1 of that paper are reproduced quite well.

---

blr_rnfb *table 1 of Rees et al. ApJ 347, 648*

```
title table 1 of Rees et al. ApJ 347, 648
c
c blr model from
c
c commands controlling continuum =========
table agn
ionization parameter -2
c
c commands for density & abundances =========
hden 12
init file="c84.ini"
abundances he -1 c -3.328 n -4.0088 o -3.0809 ne -4 na -20
continue mg -4.3768 al -5.5686 si -4.36653 s -4.76955
continue a -5.4202 ca -5.6383 fe -4.4815 ni -20
c
c commands controlling geometry =========
stop column density 23
c
```

```
c other commands for details =========
iterate to convergence
c
c commands controlling output =========
print line faint -1
punch overview last "blr_rnfb.ovr"
punch element nitrogen last "blr_rnfb.nit"
punch dr last "blr_rnfb.dr"
punch results last "blr_rnfb.rlt"
normalise "H 1" 1216 = 91.8
c
c blr_rnfb.in
c class blr
c ========================================
c
```

This is a very dense cloud, and was computed in Rees et al. (1989). Table 1 of that paper lists the predictions, which were a mean of those of Hagai Netzer's ION and roughly version 76 of CLOUDY. The lines are generally still in good agreement with the predictions of that paper. In particular the changes in the line fluxes shown in Figure 1 of that paper are reproduced quite well. The fluxes of Lya and Hb are not reproduced with great precision by this model because of changes in collision rates for hydrogen and especially the form of the escape probability function for subordinate lines. As Figure 1 of RNF showed the line intensities are very sensitive to density for these parameters.

`coll_coronal` *model of active region of solar corona*

```
title model of active region of solar corona
c
c commands controlling continuum =========
c temperature of corona
coronal equilibrium, 2.5e6 K
c assume the sun is a blackbody at 5770K
blackbody 5770 K
luminosity solar linear 1
c
c commands for density & abundances =========
hden log=10
c
c commands controlling geometry =========
c its height above photosphere and thickness
radius 11 thickness = 10
c the gas fully covers the center of symmetry
sphere
c
c other commands for details =========
iterate
c
c commands controlling output =========
normalize to "o 8" 18.97A
punch dr "coll_coronal.dr" last
punch continuum "coll_coronal.con" last units angstroms
c
c
c coll_coronal.in
c class coronal
c ========================================
c
```

This is a rough model of the solar corona. The test checks that the coronal equilibrium commands work. The

gas is predominantly collisionally ionized.

`coll_heat_only` *test code in limit where ONLY mechanical heating is present*

```
title test code in limit where ONLY mechanical heating is present
c
c commands controlling continuum =========
no photoionization
blackbody 5000 STE
c
c commands controlling geometry =========
stop column density 15
c
c commands for density & abundances =========
hden 10
c
c other commands for details =========
hextra -1.5
iterate
c
c commands controlling output =========
c this command must be tested somewhere
print cooling, zone 5
c output files
punch overview "coll_heat_only.ovr"
punch heating "coll_heat_only.het"
punch cooling "coll_heat_only.col"
punch convergence reason "coll_heat_only.cvr"
punch dr "coll_heat_only.dr"
punch continuum "coll_heat_only.con" units angstroms
c
c
c coll_heat_only.in
c class coronal
c =======================================
```

This test is an optically thin collisionally ionized gas with no photoionization at all.

`coll_t3` *coronal equilibrium at 10^4 K*

```
title coronal equilibrium at 10^4 K
c
c commands controlling continuum =========
coronal 3
c add component of cosmic rays to drive chemistry
cosmic ray background
c
c other commands for details =========
iterate
c
c commands controlling geometry =========
stop column density 15
c
c commands for density & abundances =========
hden 10
```

```
c
c commands controlling output =========
punch overview "coll_t3.ovr"
punch continuum "coll_t3.con" units angstroms
c
c
c coll_t3.in
c class coronal
c ========================================
```

This tests conditions of collisional equilibrium at low densities. This is one of a series of sims coll_t?.in which test ionization over a range of temperatures. This one, unlike the others, includes cosmic rays. Chemistry is important at this low temperature and the chemical network will collapse without a source of ionization. The cosmic rays provide this source of ionization.

---

coll_t4 *coronal equilibrium at 10^4 K*

```
title coronal equilibrium at 10^4 K
c
c commands controlling continuum =========
coronal 4
c
c other commands for details =========
iterate
c
c commands controlling geometry =========
stop column density 15
c
c commands for density & abundances =========
hden 10
c
c commands controlling output =========
punch overview "coll_t4.ovr"
punch continuum "coll_t4.con" units angstroms
c
c
c coll_t4.in
c class coronal
c ========================================
```

this tests conditions of collisional equilibrium at low densities

---

coll_t5 *coronal equilibrium at 10^5 K*

```
title coronal equilibrium at 10^5 K
c
c commands controlling continuum =========
coronal 5
c
c commands controlling geometry =========
stop column density 15
c
c commands for density & abundances =========
hden 10
c
c other commands for details =========
```

```
iterate
c
c
c commands controlling output =========
c normalize to strong HeII line
normalize to "He 2" 303.8A
punch overview "coll_t5.ovr"
punch continuum "coll_t5.con" units angstroms
c
table lines "LineList_He_like.dat"
c coll_t5.in
c class coronal
c =======================================
```

This is a test collisional ionization equilibrium at 1e5 K.

`coll_t6` *coronal equilibrium at 10^6 K*

```
title coronal equilibrium at 10^6 K
c
c commands controlling continuum =========
coronal 6
c
c commands controlling geometry =========
stop column density 15
c
c commands for density & abundances =========
hden 10
c
c other commands for details =========
iterate
c
c commands controlling output =========
normalize to "c 5" 40.27A
c output files
punch overview "coll_t6.ovr"
punch continuum "coll_t6.con" units angstroms
c
c coll_t6.in
c class coronal
c =======================================
```

This test is an optically thin collisionally ionized gas.

`coll_t7` *coronal equilibrium at 10^7 K*

```
title coronal equilibrium at 10^7 K
c
c commands controlling continuum =========
coronal 7
c
c commands controlling geometry =========
stop column density 15
c
c commands for density & abundances =========
hden 10
```

```
c
c other commands for details =========
c
normalize to "o 8" 18.97A
c commands controlling output =========
punch overview "coll_t7.ovr"
punch continuum "coll_t7.con" units angstroms
punch cooling "coll_t7.col"
c
c coll_t7.in
c class coronal
c =========================================
```

Test with only collisional ionization at a high temperature.

___

dynamics_orion_flow *Orion nebula blister with wind*

```
title Orion nebula blister with wind
c
c commands controlling continuum =========
blackbody 40000
phi(h) 13.0
brems 6
phi(h) 10
c
c commands for density & abundances =========
hden 4
abundances hii region no grains
grains orion no qheat single
c
c commands controlling geometry =========
wind -7 km/s advection
c this accounts for matter beyond end of the model
double optical depths
sphere
stop AV 5
stop temperature linear 5
c
c other commands for details =========
iterate 4
magnetic field -4
cosmic rays background
c this has no effect on dynamics, but is to desaturate the lines as per flow
turbulence 8 km/sec no pressure
c try to speed up simulation without hurting dynamics
init "fast.ini"
c use Bakes & Tielens heating
set PAH Bakes
* init "c84.ini"
* failures 5
c
c commands controlling output =========
c want to print line intensities are surface brightness, per arcsec^2
print lines surface brightness arcsec
print diffuse continuum
print line faint -1.5
punch overview "dynamics_orion_flow.ovr" no hash
punch hydrogen 21 cm "dynamics_orion_flow.21cm" no hash
punch dr "dynamics_orion_flow.dr" no hash
punch continuum last "dynamics_orion_flow.con" units microns no hash
punch pressure "dynamics_orion_flow.pre" no hash
```

```
punch heating "dynamics_orion_flow.het" no hash
punch cooling "dynamics_orion_flow.col" no hash
punch molecules "dynamics_orion_flow.mol" no hash
punch wind "dynamics_orion_flow.wnd" last
c
c dynamics_orion_flow.in
c class dynamics hii pdr
c ========================================
c
```

This is a model similar in spirit to the blister geometry H+ region model computed by Baldwin et al. (1991), but with a D-critical flow. Many physical processes have been disabled to make this simulationn faster. Grain physics is not done so the gas temperature is incorrect. The main purpose is to do a quick test of the dynamical flow with grain opacities included. The slow directory contains a full simulation of a flow like Orion.

`dynamics_veryfast` *very fast wind model*

```
title very fast wind model
c
c commands controlling continuum =========
table agn
ionization parameter -2
c
c commands for density & abundances =========
hden 9
c this is a set of commands to speed up calc
init file "fast.ini"
c
c commands controlling geometry =========
stop thickness 8.4
c this is a subsonic wind with advection
wind velo -5 advection
c
c other commands for details =========
c absolutely no convergence problems should occur
failures 1
c this degrades the continuum resolution by factor of 3, to
c speed up calc
set continuum resolution 3
iterate 4
c
c commands controlling output =========
punch pressure last "dynamics_veryfast.pre"
punch total opacity last "dynamics_veryfast.opc"
punch ionizing continuum last "dynamics_veryfast.ion"
punch continuum last "dynamics_veryfast.con"
punch dr "dynamics_veryfast.dr"
c
c
c dynamics_veryfast.in
c class dynamics
c ========================================
c
```

This is meant to be a very fast calculation to use when running extensive debug-enabled runtimes.

dynamics_veryfast_rec *very fast wind model*

```
title very fast wind model
c
c commands controlling continuum =========
table agn
ionization parameter -2
c
c commands for density & abundances =========
hden 9
c this is a set of commands to speed up calc
init file "fast.ini"
c
c commands controlling geometry =========
stop thickness 8.4
c this is a subsonic wind with advection
wind velo 5 advection
c
c other commands for details =========
c absolutely no convergence problems should occur
failures 1
c this degrades the continuum resolution by factor of 3, to
c speed up calc
set continuum resolution 3
iterate 4
c
c commands controlling output =========
punch pressure last "dynamics_veryfast_rec.pre"
punch total opacity last "dynamics_veryfast_rec.opc"
punch ionizing continuum last "dynamics_veryfast_rec.ion"
punch continuum last "dynamics_veryfast_rec.con"
punch dr "dynamics_veryfast_rec.dr"
c
c
c dynamics_veryfast_rec.in
c class dynamics
c =========================================
c
```

This is meant to be a very fast calculation to use when running extensive debug-enabled runtimes.

---

dynamics_wind *test of equations of motion in a very highly ionized wind*

```
title test of equations of motion in a very highly ionized wind
c
c test motion in electron scattering limit
c radiative acceleration (e- only) is 9.54E-7 cm s^-2
c terminal velocity (e- only) is 7.6 km s^-1
c
c commands controlling continuum =========
table agn
luminosiy (total) 45
c
c commands for density & abundances =========
hden 4
c this only includes H and He
```

```
init file="hheonly.ini"
c
c commands controlling geometry =========
radius (parsecs) 1
stop thickness (parsecs) -1
c a slow wind velocity, will assert velocity at outer radius
wind velo 0.1 ball
c
c other commands for details =========
c this will speed things up a bit
no level2
no radiation pressure
constant temperature 8
c
c commands controlling output =========
punch pressure "dynamics_wind.pre"
punch wind "dynamics_wind.wnd"
punch dr "dynamics_wind.dr"
c
c dynamics_wind.in
c class limit dynamics
c =======================================
c
```

This tests the radiative acceleration and terminal velocity of a wind in which only electron scattering is important. The parameters were chosen so that electron scattering is the dominant opacity source, so that the equations can be solved both numerically (in the example) and analytically (the expected solution given above). In a realistic wind the gas would be more neutral and line driving would dominate. The force multiplier, given in the punch wind output, is nearly unity as a result.

Checks:
- The radiative acceleration is correct (e- 9.543910-7 cm s-2).
- The terminal velocity should be 7.57 km s-1.
- Force multiplier near unity (no line driving since so highly ionized).
- Thickness of cloud correct (R-Ro + dr/2 should be 3.086391017 cm).

`feii_hin` *test feii in high density limit*

```
title test feii in high density limit
c
c commands controlling continuum =========
black 25000 K
ionization -10
c
c commands for density & abundances =========
hden 18
c force a high electron density to insure collisions dominate
eden 18
c
c commands controlling geometry =========
constant tempera 4
stop zone 1
c
c other commands for details =========
atom feii
iterate
c want to test collision only limit
```

```
no induced processes
c
c commands controlling output =========
c
c
c feii_hin.in
c class limit
c =========================================
c
```

This checks that, at high particle densitites, in which the gas should be in collisional equilibrium, the level populations of the large model Fe+ ion go to the proper values, where the departure coefficients are all equal to unity.

`feii_hirad` *feii in case of high radiation density limit*

```
title feii in case of high radiation density limit
c
c commands controlling continuum =========
black 8000 K lte
c
c commands for density & abundances =========
hden 9
element iron abundance 2
c
c commands controlling geometry =========
set dr 0
stop zone 1
c
c other commands for details =========
atom feii
constant tempera 8000
iterate
c
c commands controlling output =========
punch feii relative populations range 0 200 "feii_hirad.lv1" last
punch feii relative populations range 201 371 "feii_hirad.lv2" last
c
c
c feii_hirad.in
c class limit
c =========================================
c
```

This checks that, at high radiation densitites, in which the gas is irradiated by a blackbody in strict thermodynamic equilibrium, the level populations of the large model Fe+ ion go to the proper values, where the departure coefficients are all equal to unity.

`feii_pump` *test feii in continuum pumped limit*

```
title test feii in continuum pumped limit
c
c commands controlling continuum =========
black 25000 K
ionization -5
```

```
c
c commands for density & abundances =========
hden 10
abundances old solar 84
c commands controlling geometry =========
set dr 0
stop zone 1
c
c other commands for details =========
Case B
iterate
constant tempera 4
atom feii
c
c commands controlling output =========
punch feii continuum "feii_pump.con"
c
c feii_pump.in
```

This is a constant temperature low ionization cloud, with BLR-like densities, which includes the large FeII atom. The tests check on the emission predicted in the Fe II bands.

This model tests the large FeII model in the optically thin, continuum pumped limit. The zone thickness is set to a small value (1 cm) so that full continuum hits atom.

`feii_ste` *thermal equilibrium of FeII in STE limit*

```
title thermal equilibrium of FeII in STE limit
c
c commands controlling continuum =========
black 8000 K lte
hden 9
c
c commands for density & abundances =========
element iron abundance 2
c
c commands controlling geometry =========
set dr 0
stop zone 1
c
c other commands for details =========
atom feii
iterate
c
c commands controlling output =========
punch feii continuum last "feii_ste.con"
punch feii column density last "feii_ste.col"
punch feii populations all last "feii_ste.pop"
c
c feii_ste.in
c class function
c =======================================
c
```

This model has a very high iron abundance, 100x H, and most Fe is in the form of Fe+. It is irradiated by a blackbody in strict thermodynamic equilibrium. We check that the temperature of the gas is equal to the radiation temperature, to confirm that the thermal properties of the model FeII atom obey thermodynamics.

`feii_t4` *FeII emission in typical intermediate density photoionzied cloud*

```
title FeII emission in typical intermediate density photoionzied cloud
c
c commands controlling continuum =========
black 5e5K
ionization parameter -2
c
c commands for density & abundances =========
hden 8
element iron abundance 2
element iron ionization -2 0 -2
c
c commands controlling geometry =========
stop thickness 5
c
c other commands for details =========
constant temperature 4
atom feii
iterate 3
c
c commands controlling output =========
set line precision 6
print lines column
print line inward
print line sort wavelength
print line faint 0
punch feii continuum last "feii_t4.con" units angstroms
punch feii inward continuum last "feii_t4.cin" units angstroms
punch feii outward continuum last "feii_t4.cout" units angstroms
punch feii column density last "feii_t4.col"
punch feii populations all last "feii_t4.pop"
punch line emissivity "feii_t4.ems"
Fe2b 2797
Inwd 2797
end of lines
c
c feii_t4.in
c class function
c =======================================
c
```

This model has a very high iron abundance, 100x H, and most Fe is in the form of Fe+. it has intermediate density and should produce and FeII spectrum something like an AGN. The set line precision 6 increases the number of significant figures in the wavelengths for each line. This is needed to get the right FeII inward band.

`func_abund_fluc` *check fluctuating heavy-element abundances*

```
title check fluctuating heavy-element abundances
c commands controlling continuum =========
blackbody 5
ionization -2
c
c commands for density & abundances =========
hden 3
```

```
c this varies the abundnaces between default and 1e-4 times default,
c over a 1e10 cm period - we will compute half a cycle
fluctuatsion abundances 10.3 0 -4
c
c commands controlling geometry =========
c stop at half a cycle so pick off lowest abundances
stop thickness 10
c
c other commands for details =========
c speed things up a bit
init "c84.ini"
c
c commands controlling output =========
print column density
punch overview "func_abund_fluc.ovr"
punch abundances "func_abund_fluc.abn"
c
c commands giving asserts
c
c func_abund_fluc.in
c class function
c =======================================
c
```

This checks that the variable abundances option works

---

func_distance *check that distance and print flux earth'commands work*

```
title check that distance and "print flux earth" commands work
c
c commands controlling continuum =========
c luminosity in ionizing radiation, this and distance
c were chosen so that flux at earth assocated with total
c luminoisity of object is 1 erg cm-2 s-1
luminosity 40
laser 2
c
c commands for density & abundances =========
hden 4
init file "honly.ini"
c
c commands controlling geometry =========
stop zone 1
radius 17
c
c other commands for details =========
distance 19.450395
constant temperature 4
c
c commands controlling output =========
c give flux at Earth
print lines, flux at Earth
c
c func_distance.in
c class function
c =======================================
c
```

Normally the code predicts the intensity or luminosity of the emission lines. This test confirms that it can predict the flux recieved at the Earth instead. The model is the simplest and fastest that can be computed - a H-only constant temperature single zone. The total luminosity is set to 1e40 erg/s, and the ionization source is a laser at 2 ryd. With these set, the total luminosity in ionizing radiation, the total luminoisity in the incident continuum (the emission line labeled "Inci 0") will be 1e40.

The code will predict the flux at the Earth if both the distance to the oject is specified with the distance command, and this is requested with the print flux earth command. The distance was chosen so that the total flux at the Earth will be 1 erg/s. This is asserted at the end of the calculation.

---

func_dlaw *test model with dlaw table*

```
title test model with dlaw table
c
c commands controlling continuum =========
phi(H) 15
table agn
c
c commands for density & abundances =========
dlaw table radius
continue 16 9
continue 17 7
continue 18 5
continue 19 3
continue 20 1
continue 21 -1
end of dlaw
init "honly.ini"
c
c commands controlling geometry =========
sphere
filling factor -5
radius 17 20
c
c other commands for details =========
constant temperature 4
c
c commands controlling output =========
punch dr "func_dlaw.dr"
punch overveiw "func_dlaw.ovr"
c
c
c func_dlaw.in
c class function
c =======================================
c
```

this model tests the dlaw density table command

---

func_fulltrace *test full trace output*

```
title test full trace output
c
c commands controlling continuum =========
trace h-like hydrogen full no print
```

```
stop zone 1
table agn
ionization parameter -0.5
c
c commands for density & abundances =========
hden 9.5
init file="honly.ini"
c
c other commands for details =========
atom h-like element hydrogen collapsed levels 1
c
c commands controlling output =========
c func_fulltrace.in
c class function
c =======================================
```

this checks that trace output functions correctly

`func_globule` *test of globule command*

```
title test of globule command
c
c commands controlling continuum =========
black 40000 K
ionization parameter -1.5
c
c commands for density & abundances =========
globule initial density=5 scale depth=14
init file "ism.ini"
abundances hii region no qheat
c
c commands controlling geometry =========
c
c other commands for details =========
failures 2
no diffuse line pumping
c
c commands controlling output =========
print line faint -1.5
punch overview last "func_globule.ovr"
punch results last "func_globule.rlt"
punch dr "func_globule.dr" last
punch heating "func_globule.het" last
punch cooling "func_globule.col" last
c
c func_globule.in
c class function
c =======================================
c
```

This model uses the globule command, tests that the zoning logic works for this extreme case, and that the code is able to converge the globule model.

`func_grid_line_ratios` *test generating line ratios in a grid run*

```
title test generating line ratios in a grid run
```

```
c
c commands controlling continuum =====
blackbody 40,000K
ionization parameter -2
c
c commands for density & abundances =====
c these are to speed up the calculation, only do H, O, and Ne
init "honly.ini"
element oxygen on
element neon on
element sulphur on
element oxygen ionization 1 1 1 0.01
element neon ionization 1 1 1 0.01
element sulphur ionization 1 1 1 0.01
c vary the hydrogen density
hden 4 vary
grid 2 6.1 1
c
c other commands for details =======
c these are constant temperature models, vary T
constant temperature 4 vary
grid 4000 17000 3000
stop zone 1
c
c commands controlling output =====
punch line list "func_grid_line_ratios.pun" "func_grid_line_ratios.dat" ratio no hash
punch grid "func_grid_line_ratios.grd"
c
c commands giving (lack of) assert ==========
c
c func_grid_line_line_ratios.in
c class function
c =====================================================
```

This uses the grid command to compute line ratios for a wide range of density and temperature. The ionization is set to a uniform value and only a few elements are included. this makes the calculation faster and prevents recombination [O III] 4363 from becoming important (there is no O+3).

These are the line ratios mentioned as limits in the Johnstone et al. Spitzer cooling flow filament paper (2007).

`func_hotgas_coolstar` *test very soft continuum, very hot gas*

```
title test very soft continuum, very hot gas
c
c commands controlling continuum =========
c put in the cosmic background as the only continuum source
CMB
c gas has constant temperature of 1e6 K
constant temperature, t=6
c
c commands for density & abundances =========
hden 10.
c
c commands controlling geometry =========
stop zone 1
c specify a thin cell of gas - 1 cm thick
set dr 0
c
c other commands for details =========
c iterate since gas is optically thin
```

```
iterate
c
c commands controlling output =========
c
c func_hotgas_coolstar.in
c class limit
c ========================================
c
```

This is a test where the CMB is the only continuum source. It does not extend to energies where the code
needs to work. There are special cases used in this situation, for continuum addressing, so this checks whether
those still function.

`func_ion_increase` *test model where ionization increases with depth*

```
title test model where ionization increases with depth
c
c commands controlling continuum =========
ionization parameter -4
table agn
c
c commands for density & abundances =========
hden 7 -4
init "hheonly.ini"
element oxygen on
element iron on
c
c commands controlling geometry =========
sphere
filling factor -5
radius 17 20
c
c other commands for details =========
constant temperature 4
c
c commands controlling output =========
punch dr "func_ion_increase.dr"
punch overveiw "func_ion_increase.ovr"
punch element oxygen "func_ion_increase.oxy"
punch element iron "func_ion_increase.fe"
c
c
c func_ion_increase.in
c class function
c ========================================
c
```

This density falls off faster than $1/r^2$ so the ionization increases with depth. Most sims have decreasing rather
than increasing ionization.

`func_lines` *create output file with list of func_lines*

```
title create output file with list of func_lines
atom h2 levels large
atom feii
test
```

```
c
c commands controlling continuum =========
c commands for density & abundances =========
c commands controlling geometry =========
c other commands for details =========
c
c commands controlling output =========
c this creates the file "func_lines.lab that contains a list
c of all emission labels
punch line labels "func_lines.lab"
c this creates the file "func_lines.lis" that contains a list
c of the emission func_lines that are transported.
punch line data "func_lines.lis"
c
c There is nothing extra to assert, since test does include many asserts
c
c func_lines.in
c class function
c =======================================
c
```

This runs the standard "test" case, and then creates the line data and labels files. Test by itself includes many asserts, so no further asserts are needed here.

The file func_lines.lab is a useful list of all lines predicted by the code. Cut and paste this into other places when you need to find a particular emission line.

The func_lines lines.dat gives atomic data for all the lines, and their critical density at 10000 K. The large H2 and Fe II model atoms are turned on to include their lines.

`func_map` *map of heating vs cooling*

```
title map of heating vs cooling
c
c commands controlling continuum =========
table agn
ionization parameter -2.5
c
c commands for density & abundances =========
hden 0
abundances old solar 84
c
c commands controlling geometry =========
set dr 0
stop lyman optical depth -7
stop zone 1
c
c other commands for details =========
c
c commands controlling output =========
c save map information to generate plot for hazy
punch map file="func_map.dta"
set nmaps 100
c
c map.in
c class function
c =======================================
c
```

This is a test of the continuity of the code over a very large range of temperature. It was used to produce one of the thermal maps shown in Hazy.

Checks:
- No breaks in the heating and cooling curves where various approximations change.

`func_sdrmin` *test set drmin command*

```
title test set drmin command
c
c commands controlling continuum =========
blackbody 8e4
luminosity 37.5
c
c commands for density & abundances =========
hden 4
c
c commands controlling geometry =========
radius 17
sphere
stop thickness 17.03
stop temperature off
c
c other commands for details =========
set drmin 15.5
cosmic rays background
c
c commands controlling output =========
punch dr "func_sdrmin.dr" last
c
```

This simulation tests the SET DRMIN command. It is a toy model of a planetary nebula that is designed to extend into the PDR. The minimum stepsize is deliberately set much too large so that we would immediately notice if the command was broken. It also tests if we still hit the correct outer radius when SET DRMIN is used. Without the SET DRMIN command this sim would need 410 zones (trunk@2760).

`func_set_ion` *test impact of setting ionization*

```
title test impact of setting ionization
c
c commands for density & abundances =========
init "ism.ini"
hden 0
element hydrogen ionization 0 1
element helium ionization 0 1 0
element carbon ionization 0 1
element oxygen ionization 0 0.1 0.2 0.3 0.2 0.1 0.1
c
c commands controlling continuum =========
blackbody 5
ionization parameter -2
c
c commands controlling geometry =========
c force model to be 1 cm thick
```

```
set dr 0
c
c other commands for details =========
constant temperature 4
stop zone 1
c
c commands controlling output =========
normalize to "O 3" 5007
punch dr "func_set_ion.dr"
c
c
c func_set_ion.in
```

c class function c ====================================== c this script exercies the option to specify the ionization of a species

---

func_stopline1 *test stop line command*

```
title test stop line command
c
c commands controlling continuum =========
ionization parameter -2
blackbody 50000
c
c commands for density & abundances =========
hden 2
abundances old solar 84
init file "ism.ini"
c test option to turn off elements with small abundances
element limit off -5
c
c commands controlling geometry =========
c stop calc at line ratio, the point of this test
stop line "C 2" 157.6m reaches 0.0358
c
c other commands for details =========
constant temperature 10000 K
c
c commands controlling output =========
punch dr "func_stopline1.dr" last
c
c func_stopline1.in
c class function
c ======================================
c
```

This is an example of a simple calculation that stops when a line reaches a specified intensity. The option to turn off elements with trivial abundances is used.

---

func_stopline2 *test stop line command*

```
title test stop line command
c
c commands controlling continuum =========
ionization parameter -2
blackbody 50000
```

```
c
c commands for density & abundances =========
hden 2
abundances old solar 84
init file "ism.ini"
c
c commands controlling geometry =========
c stop calc at line ratio, the point of this test
stop line "C 2" 157.6m reaches 0.002 relative to "o 3" 5007 A
c
c other commands for details =========
constant temperature 10000 K
c
c commands controlling output =========
normalise to "o 3" 5007
punch dr "func_stopline2.dr" last
c
c func_stopline2.in
c class function
c ========================================
c
```

this is an example of a simple calculation that stops when a certain emission line ratio is reached

---

func_t10 *test very soft continuum, very hot gas*

```
title test very soft continuum, very hot gas
c
c commands controlling continuum =========
table agn
ionization parameters -2
c
c commands for density & abundances =========
hden 1.
c
c commands controlling geometry =========
stop zone 1
c specify a thin cell of gas - 1 cm thick
set dr 0
c
c other commands for details =========
c set constant temperature to high-T limit
constant temperature, t=10 log
c iterate since gas is optically thin
iterate
c
c commands controlling output =========
c
c
c func_t10.in
c class limit
c ========================================
c
```

This is a test of the highest temperature the code can do.

---

func_t3 *test low temperature limit of code, 3K*

```
title test low temperature limit of code, 3K
c
c commands controlling continuum =========
table agn
ionization parameters -2
c
c commands for density & abundances =========
hden -1
c
c commands controlling geometry =========
stop zone 1
c specify a thin cell of gas 1 cm thick
set dr 0
c
c other commands for details =========
c set constant temperature to low-T limit
constant temperature, t=3.0 linear
c iterate since gas is optically thin
iterate
c
c commands controlling output =========
c
c
c
c func_t3.in
c class limit
c =======================================
c
```

This is a test of the lowest temperature the code can do. It runs a constant temperature of 3K

---

func_test *run smoke test*

```
title run smoke test
c commands controlling continuum =========
test
c commands for density & abundances =========
c commands controlling geometry =========
c other commands for details =========
c include this in one test to make sure it works
set continuum shield ferland
c
c commands controlling output =========
c increase precision of all printed wavelengths
set line precision 6
c following will print physical constants used by the code
print constants
c print the past to the data directory
print path
c the citation for the current version of the code
print citation
c following prints column densities
print column density
punch overview "func_test.ovr"
c func_test.in
c class function
c =======================================
c
```

This runs the smoke test command, which include several asserts. The tests the behavior of increasing the number of significant figures in printed wavelengths.

`func_testmole` *this runs the standard, one command, test, which contains many asserts*

```
title this runs the standard, one command, test, which contains many asserts
c commands controlling continuum =========
test mole
c
c commands for density & abundances =========
c commands controlling geometry =========
c other commands for details =========
c
c commands controlling output =========
punch overview "func_testmole.ovr"
c func_testmole.in
c class function
c ======================================
c
```

This runs the "test mole" command, which include several asserts.

`func_trans_punch` *first of func_trans_punch/transread pair, punch continuum*

```
title first of func_trans_punch/transread pair, punch continuum
c
c commands controlling continuum =========
ionization parameter -1
table agn
c
c commands for density & abundances =========
hden 10
c
c commands controlling geometry =========
set dr 0
stop zone 1
c
c other commands for details =========
iterate
c
c commands controlling output =========
print incident continuum
punch ionization last "func_trans_punch.ion"
punch continuum last "func_trans_punch.con"
c this continuum will be used by transread.in
punch last transmitted continuum "func_trans_punch.trn"
c
c this contains a large number of asserts that both tests use
init file "trans.dat"
c
c func_trans_punch.in
c class function
```

```
c ========================================
c
```

func_trans_punch.in and transread.in are a pair of tests that check that the code can punch a transmitted continuum then read it.

This sim must come before func_trans_read since it generates the punch file needed by func_trans_read. Alphabetical order insures this.

`func_trans_read` *second of transpunch/transread pair, used transmitted continuum*

```
title second of transpunch/transread pair, used transmitted continuum
c the transmitted continuum produced by func_trans_punch
c
c commands controlling continuum =========
ionization parameter -1
table read "func_trans_punch.trn"
brems 6
ionization parameter -6
c
c commands for density & abundances =========
hden 10
c
c commands controlling geometry =========
set dr 0
stop zone 1
c
c other commands for details =========
iterate
c
c commands controlling output =========
c this contains a large number of asserts used by both commands
init file "trans.dat"
print incident continuum
punch ionization last "func_trans_read.ion"
c
c
c func_trans_read.in
c class function
c ========================================
c
```

func_trans_punch.in and func_trans_read.in are a pair of tests that check that the code can punch a transmitted continuum then read it.

`grains_conserve_pp` *test energy conservation with vastly optically thick dust*

```
title test energy conservation with vastly optically thick dust
c
c commands controlling continuum =========
black 500 STE
```

```
c
c commands for density & abundances =========
init "honly.ini"
hden 0
grains ism single graphite
c
c commands controlling geometry =========
stop Av 1000
sphere
stop temperature 100K
c
c other commands for details =========
iterate
constant temper 500 K
no grain gas collisional energy exchange
no molecules
cosmic rays background
c
c commands controlling output =========
punch grain temperature "grains_conserve_pp.grntem"
punch continuum units microns "grains_conserve_pp.con"
c
c
c grains_conserve_pp.in
c class limit
c ========================================
c
```

This tests that multiple absorption / reemission by dust conserves energy Energy density and grain temperatures should be exactly 500 K

`grains_conserve_sp` *test energy conservation with vastly optically thick dust, sphere geometry*

```
title test energy conservation with vastly optically thick dust, sphere geometry
c
c commands controlling continuum =========
black 500 STE
c
c commands for density & abundances =========
init "honly.ini"
hden 3.25 -2
grains ism single graphite
c
c commands controlling geometry =========
radius from 21 to 26
* stop Av 1000
sphere
stop temperature 100K
c
c other commands for details =========
iterate
constant temper 500 K
no grain gas collisional energy exchange
no grain electrons
no molecules
cosmic rays background -10
c
c commands controlling output =========
```

```
punch grain temperature "grains_conserve_sp.grntem"
punch continuum units microns "grains_conserve_sp.con"
c
c
c grains_conserve_sp.in
c class limit
c =======================================
c
```

This tests that multiple absorption / reemission by dust conserves energy in a spherical geometry. Radiation density temperature in last zone should be 500 * sqrt(1e21/1e26) = 1.5811 K

`grains_hot` *test temperature of gas and dust in high energy density environment*

```
title test temperature of gas and dust in high energy density environment
c
c commands controlling continuum =========
table power law
c intensity of incident radiation field set by its energy density
energy density temp 550 K
c
c commands for density & abundances =========
hden 5.0
init "ism.ini"
c Orion grains and abundances
abundances orion
c
c commands controlling geometry =========
stop zone 1
set drmax 10
c
c other commands for details =========
c
c commands controlling output =========
normalize to "O 8" 18.97A
punch overview "grains_hot.ovr" last
c
c
c grains_hot.in
c class limit
c =======================================
c
```

This tests the grains in an extreme condition - irradiation by an AGN near the illuminated face of the molecular torus. The gas is predominantly heated by the grain electron photo-ejection.

`grains_hot_wd01` *test temperature of gas and dust in high energy density environment*

```
title test temperature of gas and dust in high energy density environment
c
c commands controlling continuum =========
table power law
```

```
energy density temp 550
c
c commands for density & abundances =========
hden 5.0
init "ism.ini"
abundances orion no grains
grains orion no qheat
c
c revert to WD01 treatment
no grain x-ray treatment
c
c commands controlling geometry =========
stop zone 1
set drmax 10
c
c other commands for details =========
c
c commands controlling output =========
normalize to "O 8" 18.97A
punch overview "grains_hot_wd01.ovr" last
c
c
c grains_hot_wd01.in
c class limit
c ========================================
c
```

This tests the grains in an extreme condition - irradiation by an AGN near the illuminated face of the molecular torus. The gas is predominantly heated by the grains. The grain treatment has been reverted to Weingartner & Draine, 2001, which is NOT appropriate for these conditions. It is however a good test whether the old treatment is not broken....

---

grains_lte *check that grains equilibriate at correct temp in ste limit*

```
title check that grains equilibriate at correct temp in ste limit
c this also tests that the code works when H He ionization is ZERO
c
c commands controlling continuum =========
black 3 lte
c
c commands for density & abundances =========
hden 5
eden 5
init file "hheonly.ini"
c need to normalize to some line, oxygen will provide some
element oxygen on
c ism grains with high dust to gas ratio
grain abund 1
c
c commands controlling geometry =========
stop zone 1
set dr 0
c
c other commands for details =========
iterate
c the equilibrium temp would not be correct since small H_2 molecule
c does not go to ste, being an equivalent two-level system
constant temperature 3
c
c commands controlling output =========
```

```
normalize to "O 1" 63.17m
punch heating last "grains_lte.het"
punch cooling last "grains_lte.col"
c
c grains_lte.in
c class limit
c ========================================
c
```

This test irradiates a set of grains with a true blackbody in strict thermodynamic equilibrium. We expect the grains (and everything else) to equilibriate at the blackbody temperature. The gas temperature is forced to the radiation temperature because the current molecule network (based on ISM approximations) does not go to LTE in the high-radiation density limit. The calculation asserts that all grain temperatures are very close to the radiation temperature.

`grains_qheat` *cool atomic ISM with Si grain quantum heating*

```
title cool atomic ISM with Si grain quantum heating
c
c commands controlling continuum =========
table draine
extinguish 20
c
c commands for density & abundances =========
hden 0
c turns on grains with default abundance (log abundance ratio to ism = 0)
c and with quantum heating enabled by default
c (disabled with "no grains qheat" option)
grains 0. "silicate_ism_10.opc"
c
c commands controlling geometry =========
stop zone 1
set dr 0
c
c other commands for details =========
cosmic rays background
iterate
c
c commands controlling output =========
normalize to "C 2" 157.6m
print diffuse continuum
print line faint -2
print line column
punch grains temperature last "grains_qheat.qht"
punch continuum "grains_qheat.con" units microns last
punch two photon continuum "grains_qheat.2nu" units microns
c
c
c grains_qheat.in
c class limit
c ========================================
c
```

This sim produces dust emission with a Wein trail that is dominated by quantum heating emission.

`grains_temp` *test all grain species temperature*

```
title test all grain species temperature
c
c commands controlling continuum =========
black 50000
ionization parameter -2
c
c commands for density & abundances =========
hden 3
grains_temp "graphite_ism_01.opc"
grains_temp "silicate_ism_01.opc"
grains_temp "graphite_orion_01.opc"
grains_temp "silicate_orion_01.opc"
grains_temp "silicate_0m010.opc"
grains_temp "silicate_0m100.opc"
grains_temp "grey_ism_01.opc"
grains_temp "silicate_1m000.opc"
c
c commands controlling geometry =========
stop zone 1
c
c other commands for details =========
iterate
constant temper 4
c
c commands controlling output =========
c
c
c grains_temp.in
c class limit
c =========================================
c
```

This turns on all the grain species that are included in the distribution. A model of an ionized layer is done and the monitors confirm the resulting grain temperatures.

`grains_temp_all` *test all grain species temperature*

```
title test all grain species temperature
c
c commands controlling continuum =========
black 50000
ionization parameter -2
c
c commands for density & abundances =========
hden 3
c this turns on orion-style distributed grains
grains orion
c this turns on ism distributed grains
grains ism
c
c commands controlling geometry =========
stop zone 1
c
c other commands for details =========
constant temper 4
iterate
c
c commands controlling output =========
punch dr "grains_temp_all.dr" last
```

```
punch grain charge "grains_temp_all.grnchr" last
punch grain temperature "grains_temp_all.grntem" last
punch grain drift velocity "grains_temp_all.grndft" last
c
```

c grains_temp_all.in c class limit c == c

This is a relatively quick test of grains. The Orion and ISM silicate and graphitic grains are turned on and their equilibrium temperature checked.

`h2_cr` *H2 with background cosmic ray ionization*

```
title H2 with background cosmic ray ionization
c
c commands controlling continuum =========
c this is hot star SED
black 30000
intensity -5. range 0.4412 to 1 Ryd
c this will remove all ionizing radiation
extinguish 24 0
c background CR ionization rate of 2.5e-17 s^-1,
cosmic rays, background
c
c commands for density & abundances =========
hden 5
c
c commands controlling geometry =========
set dr 0
stop zone 5
stop temperature 400
c
c other commands for details =========
c failures should not occur
failures 1
atom h2
iterate
c want clean test of CR physics so set constant kinetic temperature
constant temperature 500K
c must turn off line photoexcitation and Solomon process
c since we want to test cr heating and ionization
no induced processes
c
c commands controlling output =========
print column density
print ages
punch overview "h2_cr.ovr" last
punch h2 lines "h2_cr.lin" last all
punch h2 rates "h2_cr.rat" last
punch h2 column density "h2_cr.col" last
c
c
c h2_cr.in
c class limit
c ========================================
c
```

This test conditions of cosmic ray ionization. Solar abundances with no dust are assumed so this involves gas-phase chemistry alone. The Solomon process is disabled with the "no induced processes' command so H2 is mainly dissociated by cosmic rays. This forms a pair with h2_cr_grains, which does include grains.

`h2_cr_grains` *background cosmic ray ionization by suprathermal electrons only*

```
title background cosmic ray ionization by suprathermal electrons only
c
c commands controlling continuum =========
c this is hot star continuum
black 30000
intensity -5. range 0.4412 to 1 Ryd
c this will remove all ionizing radiation
extinguish 24 0
c background CR ionization rate should be 2.5e-17 s^-1,
c Tielens & Hollenbach 1985; McKee AstroPH 9901370
cosmic rays, background
c
c commands for density & abundances =========
abundances ism
hden 5
c
c commands controlling geometry =========
set dr 0
stop zone 5
stop temperature 400
c
c other commands for details =========
set h2 grain formation thermal pump
atom h2
iterate
constant temperature 500K
c must turn off line photoexcitation
c since we want to test cr heating and ionization
no induced processes
c
c commands controlling output =========
normalize to "H2 " 2.121m
print column density
print ages
punch overview "h2_cr_grains.ovr" last
punch h2 lines "h2_cr_grains.lin" last all
punch h2 rates "h2_cr_grains.rat" last
punch h2 column density "h2_cr_grains.col" last
punch grains H2rates "h2_cr_grains.h2r" last
c
c h2_cr_grains.in
c class limit
c =======================================
c
```

This tests conditions of cosmic ray ionization and grain formation pumping. Solomon process is turned off with the "no induced processes" command so cosmic rays are the main dissociation process. This forms a pair with h2_cr which does not include grains, so relies only on gas-phase chemistry.

`h2_hminus` *H2 populations in H- dominated limit*

```
title H2 populations in H- dominated limit
```

```
c this is one of a pair with h2_solomon.in
c
c commands controlling continuum =========
c this is hot star continuum
black 30000
intensity -5. range 0.4412 to 1 Ryd
c this will remove all ionizing radiation
extinguish 24 0
c background CR ionization rate should be 2.5e-17 s^-1,
c Tielens & Hollenbach 1985; McKee AstroPH 9901370
cosmic rays, background
c
c commands for density & abundances =========
c no grains in this sim so H2 forms by H- route
abundances ism no grains
hden 5
c
c commands controlling geometry =========
set dr 0
stop zone 5
stop temperature 400
c
c other commands for details =========
c default is Takahashi 2001 - test thermal distribution here
set h2 grain formation thermal pump
atom h2
iterate
constant temperature 500K
c
c commands controlling output =========
print column density
print ages
punch overview "h2_hminus.ovr" last
punch h2 lines "h2_hminus.lin" last all
punch h2 rates "h2_hminus.rat" last
punch h2 column density "h2_hminus.col" last
c
c h2_hminus.in
c class limit
c ==========================================
c
```

This tests large H2 model in limit of H- formation and Solomon destruction.

`h2_pdr_leiden_f1` *low density and flux model 1*

```
title low density and flux model 1
atom H2 He collisions new
c
c commands controlling continuum =========
c Use the Draine 1978 field, for a semi-infinite slab we have to use half
c the requested value, so the actual value
c they want for the model is actually twice the value below
table draine 5
c this is to insure that no H-ionizing radiation strikes the cloud
extinguish 24
c
c commands for density & abundances =========
c hydrogen density
hden 3.
grains ism 1.16 no qheat
```

```
c
c commands controlling geometry =========
c
c other commands for details =========
c turn on the large H2 model
atom h2
c turn down the number of convergence failures - there should be none
failures 3
c use leiden initialization file
init file="pdr_leiden.ini"
c This command defines the grain temperature to be a constant 20 Kelvin
constant grain temperature 20
c This sets the gas kinetic temperature to a constant 50 Kelvin
constant temperature 50 linear
c
c commands controlling output =========
c default normalization line is Hbeta, which is not produced by this gas
normalize to "C 2" 157.6m
print line faint -4
punch overview "h2_pdr_leiden_f1.ovr"
punch line emissivity "h2_pdr_leiden_f1.ems"
H2 2.121m
12CO 647.2m
C 2 157.6m
O 1 63.17m
Si 2 34.81m
end of lines
punch leiden lines "h2_pdr_leiden_f1.lin"
punch leiden "h2_pdr_leiden_f1.lei"
punch dr "h2_pdr_leiden_f1.dr"
punch molecules "h2_pdr_leiden_f1.mol"
punch grain physical conditions "h2_pdr_leiden_f1.grn"
punch overview "h2_pdr_leiden_f1.ovr"
punch H2 lines "h2_pdr_leiden_f1.h2lin" all
punch H2 column density "h2_pdr_leiden_f1.h2col"
punch H2 populations matrix zone "h2_pdr_leiden_f1.pop"
punch H2 rates "h2_pdr_leiden_f1.rat"
c
c
c h2_pdr_leiden_f1.in
c class pdr
c =========================================
c
```

This is one of the tests in Rollig et al. 2007, A&A, 467, 187

---

h2_solomon *H2 populations in solomon dominated limit*

```
title H2 populations in solomon dominated limit
c this is a pair with h2_hminus
c
c commands controlling continuum =========
c this is hot star continuum
black 30000
intensity -5. range 0.4412 to 1 Ryd
c this will remove all ionizing radiation
extinguish 24 0
c background CR ionization rate should be 2.5e-17 s^-1,
c Tielens & Hollenbach 1985; McKee AstroPH 9901370
cosmic rays, background
c
```

```
c commands for density & abundances =========
abundances ism
hden 5
c
c commands controlling geometry =========
set dr 0
stop zone 5
stop temperature 400
c
c other commands for details =========
c default is Takahashi 2001 - test thermal distribution here
set h2 grain formation thermal pump
atom h2
iterate
constant temperature 500K
c
c commands controlling output =========
c don't print so many lines
print line faint 10
print line column
print line sort wavelength
print column density
print ages
punch overview "h2_solomon.ovr" last
punch h2 lines "h2_solomon.lin" last all
punch h2 rates "h2_solomon.rat" last
punch h2 column density "h2_solomon.col" last
c
c h2_solomon.in
c class limit
c =========================================
c
```

This test H2 in case of grain formation and solomon destruction

---

h2_t2000 *test large H2 molecule in shock-like conditions*

```
title test large H2 molecule in shock-like conditions
c
c commands controlling continuum =========
c there is a continuum but it is unimportant
c first continuum is FIR hot grain continuum produced in
c unmodeled HII Region
black 30000
intensity -5 range 0.4412 to 1 Ryd
c this will remove all ionizing radiation
extinguish 24 0
c must provide source of ionization for chemistry
cosmic rays, background
c
c commands for density & abundances =========
hden 6
grains ism, abundance log 0.16 single
init file="ism.ini"
abundances he -1.01 c -3.52 n-8 o-3.30 ne-8 mg-5.89
continue si -6.10 s -5.10 cl=-7 ar-8 fe -6.60
c
c commands controlling geometry =========
stop zone 10
c force zone thickess to 1 cm
set dr 0
```

```
c need to lower stop temperature since gas is cooler than 4000K
stop temperature 3
c
c other commands for details =========
failures 2
atom h2
constant temperature 2000
iterate
c this stop Lymnan line excitation of H, which is then photoionized
c by Balmer continuum
case b
c
c commands controlling output =========
normalize to "H2 " 2.121m
c
punch pdr "h2_t2000.pdr"
punch h2 rates "h2_t2000.rat"
punch h2 column density "h2_t2000.col"
punch h2 lines "h2_t2000.lin" all last
punch h2 populations "h2_t2000.pop" all last
punch raw continuum "h2_t2000.raw"
punch continuum "h2_t2000.con"
punch overview "h2_t2000.ovr"
c
c h2_t2000.in
c class limit
c =========================================
```

This is a collisionally dominated H2 simulation. The temperature has been fixed at 2000K and the large molecule turned on. The calculation checks the returned value of the ortho to para densities. Cosmic rays and the incident continuum have little effect, the density is high, so the populations should be close to LTE.

---

h2_t500 *test large H2 molecule in PDR-like conditions*

```
title test large H2 molecule in PDR-like conditions
c
c commands controlling continuum =========
c first continuum is FIR hot grain continuum produced in
c unmodeled HII Region
blackbody, t = 75 K
intensity -2.7 (total)
c this is hot star continuum
black 30000
intensity -5 range 0.4412 to 1 Ryd
c this will remove all ionizing radiation
extinguish 24 0
c
c commands for density & abundances =========
hden 6
grains ism, abundance log 0.16 single, no qheat
init file="ism.ini"
abundances he -1.01 c -3.52 n-8 o-3.30 ne-8 mg-5.89
continue si -6.10 s -5.10 cl=-7 ar-8 fe -6.60
turbulence 2.7 km/sec
c
c commands controlling geometry =========
stop zone 10
c force zone thickess to 1 cm
sphere
set dr 0
```

```
c stop when gas is fully neutral
stop efrac -10
c stop when gas is cold
stop temperature 10 linear
c
c other commands for details =========
failures 2
atom h2
constant temperature 500
iterate
case b
cosmic rays background
c
c commands controlling output =========
normalize to "12CO" 863.0m
c uv lines are strongly pumped by stellar continuum, break out contribution
print line pump
print line optical depths
print line faint -2
print ages
c
punch pdr "h2_t500.pdr"
punch H2 rates "h2_t500.rat"
punch H2 lines "h2_t500.lin" all
punch H2 column density "h2_t500.col"
punch H2 populations "h2_t500.pop"
punch H2 levels "h2_t500.lev"
punch raw continuum "h2_t500.raw"
punch continuum "h2_t500.con"
punch overview "h2_t500.ovr"
c
table lines "LineList_PDR_H2.dat"
c h2_t500.in
c class limit
c =========================================
```

This is a dense molecular gas with background cosmic rays and the incident radiation field set to a small value. The lower levels are in LTE.

---

h_otsopen *test ots, inward fractions for pure hydrogen, open geo, filling factor*

```
title test ots, inward fractions for pure hydrogen, open geo, filling factor
c
c commands controlling continuum =========
ionization -2
blackbody 40000
c
c commands for density & abundances =========
hden 5
init file "honly.ini"
c
c commands controlling geometry =========
stop efrac 0.05 %stop when 5 percent ionized
constant temper 7500
filling factor 0.01
c
c other commands for details =========
c must iterate to get inward line fractions, that are asserted below
```

```
iterate
diffuse ots
c not iterating does not affect the solution
no level2
no induced processes (OK)
c
c commands controlling output =========
print diffuse continuum
print line inward
punch overview "h_otsopen.ovr" last
punch dr "h_otsopen.dr" last
punch continuum "h_otsopen.con" last units angstrom
normalize to "Ca B" 4861 = 1 % normalize to exact result
c
c this is an open geometry, so do not expect H atom to exactly be case B
c due to Lyman line leakage. we iterated, so have a good estimate of
c inward and outward fractions, which we test
c h_otsopen.in
c class limit
c =========================================
c
```

This tests the total emission from a hydrogen Stromgren sphere using the OTS approximation. The conservation of the total number of ionizing photons, and the emitted spectrum, are all checked.

---

h_otspp *plane parallel conservation and hydrogenic emission for pure hydrogen*

```
title plane parallel conservation and hydrogenic emission for pure hydrogen
c
c commands controlling continuum =========
ionization -2
blackbody 40000
c
c commands for density & abundances =========
hden 5
init file "honly.ini"
c
c commands controlling geometry =========
sphere static (OK)
stop efrac 0.05 %stop when 5 percent ionized
c
c other commands for details =========
constant temper 7500
no level2
no induced processes (OK)
c needed to prevent 2s / Bal from becoming optically thick
turbulence 20 km/s
diffuse ots
atom h-like levels resolved 10
c
c commands controlling output =========
print diffuse continuum
normalize to "Ca B" 4861 = 1 % normalize to exact result for comp ion struc
punch overview "h_otspp.ovr"
punch dr "h_otspp.dr"
c
c h_otspp.in
c class limit
```

```
c ========================================
c
```

This tests the total emission from a plane parallel pure hydrogen Stromgren sphere using the OTS approximation. The conservation of the total number of ionizing photons, and the emitted spectrum, are all checked.

h_otssp *spherical conservation and hydrogenic emission for pure hydrogen*

```
title spherical conservation and hydrogenic emission for pure hydrogen
c
c commands controlling continuum =========
ionization -2
blackbody 40000
c
c commands for density & abundances =========
hden 5
init file "honly.ini"
c
c commands controlling geometry =========
radius 13
c not iterating does not affect the solution
sphere static (OK)
stop efrac 0.05 %stop when 5 percent ionized
c
c other commands for details =========
constant temper 7500
diffuse ots
no level2
no induced processes (OK)
c
c commands controlling output =========
print diffuse continuum
normalize to "Ca B" 4861 = 1 % normalize to exact result
punch overview "h_otssp.ovr"
punch dr "h_otssp.dr"
c
c h_otssp.in
c class limit
c ========================================
c
```

This tests the total emission from a spherical pure hydrogen Stromgren sphere using the OTS approximation. The conservation of the total number of ionizing photons, and the emitted spectrum, are all checked.

h_outopen *test open geometry*

```
title test open geometry
c
c commands controlling continuum =========
ionization -2
blackbody 40000
c
c commands for density & abundances =========
```

```
hden 0
init file "honly.ini"
c
c commands controlling geometry =========
stop efrac 0.05 %stop when 5 percent ionized
filling 0.01
c
c other commands for details =========
diffuse outward
iterate
atom h-like levels resolved 15
no level2
no induced processes (OK)
constant temper 4
c
c commands controlling output =========
print diffuse continuum
c normally will not print out inward parts of lines
print line inward
normalize to "Ca B" 4861 = 1 % normalize to exact result
punch overview "h_outopen.ovr" last
punch dr "h_outopen.dr" last
c
c this is an open geometry, so do not expect H atom to exactly be case B
c due to Lyman line leakage. we iterated, so have a good estimate of
c inward and outward fractions, which we test
c h_outopen.in
c class limit
c =========================================
c
```

This tests the total emission from an open geometry, hydrogen Stromgren sphere, using the outward only approximation. The conservation of the total number of ionizing photons, and the emitted spectrum, are all checked.

`h_outpp` *plane parallel H-only, close, test hydrogenic emission*

```
title plane parallel H-only, close, test hydrogenic emission
c
c commands controlling continuum =========
ionization -2
blackbody 40000
c
c commands for density & abundances =========
c the hydrogen density
hden 5
c an ini file that will include H-only
init file "honly.ini"
c
c commands controlling geometry =========
stop efrac 0.05 %stop when 5 percent ionized
c need this to prevent lyman lines from escaping
sphere static (OK)
c
c other commands for details =========
c prevent continuum pumping of lyman lines
no induced processes (OK)
c must include this to keep Ha optically thin
turbulence 20 km/s
c this is the approximation we are testing
```

```
diffuse outward
c force a constant temperature, then give stopping criteria
c if stop were not present model would go to limiting number of zones
constant temper 7500
c increase number of levels to get full emission
atom h-like levels resolved 20
c
c commands controlling output =========
c need to turn this on so that continua at various energies will
c be included in the emission line list
print diffuse continuum
c normalize results to perfect case b - our predictions should
c be close to this
normalize to line "Ca B" 4861
c the output and asserts
punch overview "h_outpp.ovr"
punch dr "h_outpp.dr"
c
c lots of continuum emission points turned on with print diffuse continuum
c energies dominated by two photon emission not included since very
c sensitive to Lya transport
c h_outpp.in
c class limit
c =======================================
c
```

This tests the total emission from a plane parallel pure hydrogen Stromgren sphere using the outward only approximation. The conservation of the total number of ionizing photons, and the emitted spectrum, are all checked.

---

h_outsp *spherical conservation and hydrogenic emission for pure hydrogen*

```
title spherical conservation and hydrogenic emission for pure hydrogen
c
c commands controlling continuum =========
ionization -2
blackbody 40000
c
c commands for density & abundances =========
hden 5
init file "honly.ini"
c
c commands controlling geometry =========
radius 13
c not iterating does not affect the solution
sphere static (OK)
stop efrac 0.05 %stop when 5 percent ionized
c
c other commands for details =========
constant temper 7500
no level2
atom h-like levels resolved 20
no induced processes (OK)
diffuse outward
c
c commands controlling output =========
print diffuse continuum
normalize to "Ca B" 4861 = 1 % normalize to exact result
```

```
punch overview "h_outsp.ovr"
punch dr "h_outsp.dr"
c
c lots of continuum emission points turned on with print diffuse continuum
c energies dominated by two photon emission not included since very
c sensitive to Lya transport
c h_outsp.in
c class limit
c =========================================
c
```

This tests the total emission from a spherical pure hydrogen Stromgren sphere using the outward only approximation. The conservation of the total number of ionizing photons, and the emitted spectrum, are all checked.

`h_t4_conemis` *continuous emission from H atom*

```
title continuous emission from H atom
c
c commands controlling continuum =========
c use laser so that stellar continuum is not in predicted continuum
laser 2
ionization -1
c
c commands for density & abundances =========
c high density to suppress two-photon emission
hden 8
init "honly.ini"
c
c commands controlling geometry =========
c this thisckness is the inverse square of the density, so the
c total emission from the computed slab will in effect have the
c square of the density removed, so predictions will be j pi j / n_e n_p
stop thickness -16
c
c other commands for details =========
c include a large number of levels to capture IR emission
c increase default resolution stored in continuum_mesh.ini
set continuum resolution 0.1
iterate
constant temperature 4
c
c commands controlling output =========
print diffuse continuum
```
print lines column *punch continuum h_t4_conemis.con"last no units microns*
```
punch continuum "h_t4_conemis.con" last no title units microns
```
punch spectrum h_t4_conemis.ncon"last no units microns
```
punch spectrum "h_t4_conemis.ncon" last no title units microns
c
c h_t4_conemis.in
c class limit
c =========================================
c
```

This tests the continuous emission from the model H atom. The gas temperature is 10,000 K and the continuous emissivity is asserted for a range of wavelengths.

This was used to generate the plot in Hazy 2 comparing the emission from a pure hydrogen plasma with those

of Ferland 1980.

//>>refer HI emission Ferland, G. J. 1980, PASP, 92, 596

`h_t4_conemis_lon` *low-den continuous HI emission with 2-nu important*

```
title low-den continuous HI emission with 2-nu important
c
c commands controlling continuum =========
laser 2
ionization -1
c
c commands for density & abundances =========
hden 0
init "honly.ini"
c
c commands controlling geometry =========
stop thickness 0
c
c other commands for details =========
c increase default resolution stored in continuum_mesh.ini
set continuum resolution 0.1
case b
iterate
constant temperature 4
c
c commands controlling output =========
print diffuse continuum
print lines column
punch continuum "h_t4_conemis_lon.con" last units microns
punch two photon continuum "h_t4_conemis_lon.2nu" last rydbergs
punch spectrum "h_t4_conemis_lon.ncon" last units microns
c
c h_t4_conemis_lon.in
c class limit
c =======================================
c
```

This is a mate to hatomt10.in except that the density is low enough for two-photon emission to be very imporant in the optical and uv.

`h_t4_conemis_thick` *H I continuous emissivity, used for plot in hazy*

```
title H I continuous emissivity, used for plot in hazy
c
c commands controlling continuum =========
black 5
ionization parameter -2
c
c commands for density & abundances =========
c high density to suppress two-photon emission
hden 7
c turn off all elements, except for hydrogen, so that
c we really test hydrogen emission
```

```
init file= "honly.ini"
c
c commands controlling geometry =========
c this is necessary to stop model at H ionization front
c constant temperature models will go on to infinity without this
stop efrac -0.8
c
c other commands for details =========
turbulence 200 km/s ; needed to keep balmer lines thin
c increase continuum resolution by factor of ten
set continuum resolution 0.1
c line fluorescense would be very important with above turbulence,
c turn it off
no induced processes
c need large atom to get the many Rydgerg levels in the infrared
c atom h-like levels resolved 10
c what full outward flux so use spherical geometry
sphere
constant temperature 4
c
c commands controlling output =========
c results will be relative to interplated Hummer&Storey Hbeta
c this tests hydrogen atom too
normalize "Ca B" 4861
c by default, the continuum flux at many wavelengths is not printed,
c this turns that information on
print diffuse continuum
c output the predicted continuum
punch emitted continuum last iteration "h_t4_conemis_thick.con" units micron
punch dr last "h_t4_conemis_thick.dr"
c
c h_t4_conemis_thick.in
c class limit
c =======================================
c
```

This checks that the predicted hydrogen continuum is in good agreement with exact results in the optically thin nebular limit.

Checks:
- This output was used to generate figure h_t4_conemis_thick in Part I of HAZY.
- Continuum relative to Hbeta should agree with Ferland (1980) filter averaged results.
- Hbeta should agree with Case B predictions, and Q(H) 4861.

---

heatomt10 *continuous emission from HeI*

```
title continuous emission from HeI
c
c commands controlling continuum =========
laser 1.9 ryd
ionization -1
c
c commands for density & abundances =========
c need high density to supress two-photon emission
hden 10
c force high density for electrons, He+ density will be the same
set eden 12
c set He/H ratio to 100, He and elec will have same density
```

```
element abundance helium 2
init "hheonly.ini"
c
c commands controlling geometry =========
c this thickness will get unit emission when combined with above densities
stop thickness -24
c
c other commands for details =========
iterate
constant temperature 4
c
c commands controlling output =========
print continuum
c
set continuum resolution 0.1
set nFnu diffuse outward, diffuse inward
print continuum
print line faint 1
print line column
```
punch spectrum "heatomt10.spc" last no units microns

punch spectrum "heatomt10.spc" last no title units microns *punch continuum "heatomt10.con" last no units microns*

punch continuum "heatomt10.con" last no title units microns *punch diffuse continuum "heatomt10.dif" last no units microns*
```
punch diffuse continuum "heatomt10.dif" last no title units microns
c
c heatomt10.in
c class limit
c ==========================================
c
```

This tests continuous emission from the He I atom. The laser is used so that the incident continuum is not included in the total emission.

`heatomt10lon` *test low-den continuous emission from H atom, 2-nu is important*

```
title test low-den continuous emission from H atom, 2-nu is important
c
c commands controlling continuum =========
laser 2
ionization -1
c
c commands for density & abundances =========
hden -8
element helium abundance log 3
set eden -5
init "hheonly.ini"
c
c commands controlling geometry =========
stop thickness log 10
c
c other commands for details =========
no scattering opacity
c force all helium to be in He+
element helium ionization -4 0 -4
case b
c increase default resolution stored in continuum_mesh.ini
set continuum resolution 0.1
```

```
iterate
constant temperature 4
c
c commands controlling output =========
normalize to "he 1" 4471
print lines column
print diffuse continuum
print lines column
punch continuum "heatomt10lon.con" last units microns
punch diffuse continuum "heatomt10lon.dif" last units microns
punch two photon continuum "heatomt10lon.2nu" last
punch spectrum "heatomt10lon.ncon" last units microns
c
c heatomt10lon.in
c class limit
c =======================================
c
```

This is a mate to hatomt10.in except that everything is He at low density

---

heiont10 *continuous emission from HeII*

```
title continuous emission from HeII
c
c commands controlling continuum =========
laser 5 ryd
ionization 1
c
c commands for density & abundances =========
hden 8
set eden 10
element abundance helium 2
init "hheonly.ini"
c
c commands controlling geometry =========
stop thickness -20
c
c other commands for details =========
constant temperature 4
iterate
c increase continuum resolution by factor of ten
set continuum resolution 0.1
set nFnu diffuse outward, diffuse inward
c
c commands controlling output =========
print continuum
print line faint 1
```
print line column *punch continuum heiont10.con last no units microns*

punch continuum "heiont10.con" last no title units microns *punch spectrum heiont10.spc last no units microns*

punch spectrum "heiont10.spc" last no title units microns
```
c
c heiont10.in
c class limit
c =======================================
c
```

This tests the He II continuous emission. The helium abundance is very large so that He II overwhelms other emission sources. The resolution of the continuum mesh is increased so that we get a better representation of

the continuous emission.

---

helike_ar *He-like argon emission*

```
title He-like argon emission
c
c commands controlling continuum =========
laser 600
ionization -1
c
c commands for density & abundances =========
hden 7
c include lots of the element we are testing and set the ionization
c to be dominated by the correct ion stage
element argon abundance -2
element argon ionization -5 -5 -5 -5 -5 -5 -5 -5 -5 -5 -5 -5 -5 -5 -5 -5 -5 0 -5
init file "ism.ini"
element helium off
element nitrogen off
element carbon off
element neon off
element magnesium off
element silicon off
element sulphur off
element oxygen off
element iron off
c
c commands controlling geometry =========
set dr -12
stop zone 1
c
c other commands for details =========
atom he-like levels 10
atom he-like collapsed levels 20
constant temper 5
iterate
c
c commands controlling output =========
print he-like departure argon
print line faint -2
c This is Lyman alpha
normalize to "Ar17" 3.949A
c
c 3.994A is 2^3S to ground
c Totl is 2^3P to ground
c 3.365A is 3^1P to ground
c 21.54A is 3^3P to 2^3S
c 22.24A is 3^3D to 2^3P
c
c helike_ar.in
c class limit
c =======================================
c
```

test He-like emission for argon

---

helike_c *he-like carbon emission*

```
title he-like carbon emission
c
c commands controlling continuum =========
laser 50
ionization -1
c
c commands for density & abundances =========
hden 7
c include lots of the element we are testing and set the ionization
c to be dominated by the correct ion stage
element carbon abundance -2
element carbon ionization -5 -5 -5 -5 -5 0 -5
init file "ism.ini"
element helium off
element nitrogen off
element oxygen off
element neon off
element magnesium off
element silicon off
element sulphur off
element argon off
element iron off
c
c commands controlling geometry =========
set dr -12
stop zone 1
c
c other commands for details =========
atom he-like levels 10
atom he-like collapsed levels 20
constant temper 5
iterate
c
c commands controlling output =========
print he-like departure carbon
print line faint -2
c This is Lyman alpha
normalize to "C 5" 40.27A
c
c
c helike_c.in
c class limit
c =========================================
c
```

test he-like carbon emission

`helike_co` *He-like cobalt emission*

```
title He-like cobalt emission
atom he-like levels 10
atom he-like collapsed levels 20
print he-like departure cobalt
constant temper 5
iterate
hden 7
set dr -12
init file "ism.ini"
c must turn on since ism.ini turned it off
element cobalt on
```

```
c then set abundance and ionization after turning it on
element cobalt abundance -2
element cobalt ionization -5 -5 -5 -5 -5 -5 -5 -5 -5 -5 -5 -5 -5 -5 -5 -5 -5 -5 -5 -5 -5
-5 -5 -5 -5 -5 0 -5
c now turn lots of elements off to save time
element helium off
element nitrogen off
element carbon off
element neon off
element magnesium off
element silicon off
element sulphur off
element oxygen off
element argon off
element iron off
laser 1300
ionization -1
stop zone 1
print line faint -2
c
c helike_co.in
c class limit
c ========================================
c
```

test emission of He-like Co

`helike_cu` *He-like copper emission*

```
title He-like copper emission
atom he-like levels 10
atom he-like collapsed levels 20
print he-like departure copper
constant temper 5
iterate
hden 7
set dr -12
init file "ism.ini"
c must turn on since ism.ini turned it off
element copper on
c then set abundance and ionization after turning it on
element copper abundance -2
element copper ionization -5 -5 -5 -5 -5 -5 -5 -5 -5 -5 -5 -5 -5 -5 -5 -5 -5 -5 -5 -5 -5
-5 -5 -5 -5 -5 -5 -5 0 -5
c now turn lots of elements off to save time
element helium off
element nitrogen off
element carbon off
element neon off
element magnesium off
element silicon off
element sulphur off
element oxygen off
element argon off
element iron off
laser 1300
ionization -1
stop zone 1
print line faint -2
c
c helike_cu.in
```

```
c class limit
c ========================================
c
```

test emission of He-like Cu

`helike_fe` *he-like iron emission*

```
title he-like iron emission
c
c commands controlling continuum =========
laser 1300
ionization -1
c
c commands for density & abundances =========
hden 7
set dr -12
c include lots of the element we are testing and set the ionization
c to be dominated by the correct ion stage
element iron abundance -2
element iron ionization -5 -5 -5 -5 -5 -5 -5 -5 -5 -5 -5 -5 -5 -5 -5 -5 -5 -5 -5 -5 -5
-5 -5 -5 -5 0 -5
init file "ism.ini"
element helium off
element nitrogen off
element carbon off
element neon off
element magnesium off
element silicon off
element sulphur off
element oxygen off
element argon off
c
c commands controlling geometry =========
stop zone 1
c
c other commands for details =========
atom he-like levels 10
atom he-like collapsed levels 20
constant temper 5
iterate
c
c commands controlling output =========
print line faint -2
print he-like departure iron
c This is Lyman alpha
normalize to "Fe25" 1.850A
c
c
c helike_fe.in
c class limit
c ========================================
c
```

check He-like emission for iron

`helike_mg` *he-like magnesium emission*

```
title he-like magnesium emission
c
c commands controlling continuum =========
laser 250
ionization -1
c
c commands for density & abundances =========
hden 7
set dr -12
element magnesium abundance -2
element magnesium ionization -5 -5 -5 -5 -5 -5 -5 -5 -5 -5 -5 0 -5
init file "ism.ini"
element helium off
element nitrogen off
element carbon off
element neon off
element silicon off
element oxygen off
element sulphur off
element argon off
element iron off
c
c commands controlling geometry =========
stop zone 1
c
c other commands for details =========
atom he-like levels 10
atom he-like collapsed levels 20
constant temper 5
iterate
c
c commands controlling output =========
print he-like departure magnesium
print line faint -2
c This is Lyman alpha
normalize to "Mg11" 9.169A
c
c
c helike_mg.in
c class limit
c =========================================
c
```

test He-like Mg emission

`helike_n` *He-like nitrogen emission*

```
title He-like nitrogen emission
c
c commands controlling continuum =========
laser 75
ionization -1
c
c commands for density & abundances =========
hden 7
element nitrogen abundance -2
element nitrogen ionization -5 -5 -5 -5 -5 -5 0 -5
init file "ism.ini"
element helium off
element carbon off
```

```
element oxygen off
element neon off
element magnesium off
element silicon off
element sulphur off
element argon off
element iron off
c
c commands controlling geometry =========
stop zone 1
set dr -12
c
c other commands for details =========
atom he-like levels 10
atom he-like collapsed levels 20
constant temper 5
iterate
c
c commands controlling output =========
print he-like departure nitrogen
print line faint -2
c This is Lyman alpha
normalize to "N 6" 28.79A
c
c helike_n.in
c class limit
c ========================================
c
```

test He-like emission for N

`helike_ne` *he-like neon emission*

```
title he-like neon emission
c
c commands controlling continuum =========
laser 170
ionization -1
c
c commands for density & abundances =========
hden 7
c include lots of the element we are testing and set the ionization
c to be dominated by the correct ion stage
element neon abundance -2
element neon ionization -5 -5 -5 -5 -5 -5 -5 -5 -5 0 -5
init file "ism.ini"
element helium off
element nitrogen off
element carbon off
element oxygen off
element magnesium off
element silicon off
element sulphur off
element argon off
element iron off
c
c commands controlling geometry =========
stop zone 1
set dr -12
c
c other commands for details =========
```

```
atom he-like levels 10
atom he-like collapsed levels 20
constant temper 5
iterate
c
c commands controlling output =========
print he-like departure neon
print line faint -2
c This is Lyman alpha
normalize to "Ne 9" 13.45A
c
c
c helike_ne.in
c class limit
c =======================================
c
```

test He-like emission for oxygen

`helike_ni` *he-like nickel emission*

```
title he-like nickel emission
atom he-like levels 10
atom he-like collapsed levels 20
print he-like departure nickel
constant temper 5
iterate
hden 7
set dr -12
init file "ism.ini"
c must turn on since ism.ini turned it off
element nickel on
c then set abundance and ionization after turning it on
element nickel abundance -2
element nickel ionization -5 -5 -5 -5 -5 -5 -5 -5 -5 -5 -5 -5 -5 -5 -5 -5 -5 -5 -5 -5
-5 -5 -5 -5 -5 -5 0 -5
c now turn lots of elements off to save time
element helium off
element nitrogen off
element carbon off
element neon off
element magnesium off
element silicon off
element sulphur off
element oxygen off
element argon off
element iron off
laser 1300
ionization -1
stop zone 1
print line faint -2
c
c helike_ni.in
c class limit
c =======================================
c
```

Test He-like Ni emission.

`helike_o` *he-like oxygen ion vs. Bautista & Kallman 2000 Table 1, column 3*

```
title he-like oxygen ion vs. Bautista & Kallman 2000 Table 1, column 3
c
c commands controlling continuum =========
laser 100
ionization -1
c
c commands for density & abundances =========
hden 7
c include lots of the element we are testing and set the ionization
c to be dominated by the correct ion stage
element oxygen abundance -2
element oxygen ionization -5 -5 -5 -5 -5 -5 -5 0 -5
init file "ism.ini"
element helium off
element nitrogen off
element carbon off
element neon off
element magnesium off
element silicon off
element sulphur off
element argon off
element iron off
c
c commands controlling geometry =========
stop zone 1
set dr -12
c
c other commands for details =========
atom he-like levels 10
atom he-like collapsed levels 20
constant temper 5
iterate
c
c commands controlling output =========
print he-like departure oxygen
print line sort wavelength
print line faint -5
normalize to "o 7" 21.60A
c
c helike_o.in
c class limit
c =======================================
c
```

test He-like emission for oxygen

`helike_si` *He-like silicon emission*

```
title He-like silicon emission
c
c commands controlling continuum =========
laser 350
ionization -1
c
```

```
c commands for density & abundances =========
hden 7
c include lots of the element we are testing and set the ionization
c to be dominated by the correct ion stage
element silicon abundance -2
element silicon ionization -5 -5 -5 -5 -5 -5 -5 -5 -5 -5 -5 -5 -5 0 -5
init file "ism.ini"
element helium off
element nitrogen off
element carbon off
element neon off
element magnesium off
element oxygen off
element sulphur off
element argon off
element iron off
c
c commands controlling geometry =========
stop zone 1
set dr -12
c
c other commands for details =========
atom he-like levels 10
atom he-like collapsed levels 20
constant temper 5
iterate
c
c commands controlling output =========
print he-like departure silicon
print line faint -2
c This is Lyman alpha
normalize to "Si13" 6.648A
c
c
c helike_si.in
c class limit
c
```

test He-like emission for silicon

---

helike_zn *He-like zinc emission*

```
title He-like zinc emission
atom he-like levels 10
atom he-like collapsed levels 20
print he-like departure zinc
constant temper 5
iterate
hden 7
set dr -12
init file "ism.ini"
c must turn on since ism.ini turned it off
element zinc on
c then set abundance and ionization after turning it on
element zinc abundance -2
element zinc ionization -5 -5 -5 -5 -5 -5 -5 -5 -5 -5 -5 -5 -5 -5 -5 -5 -5 -5 -5 -5
-5 -5 -5 -5 -5 -5 -5 -5 0 -5
c now turn lots of elements off to save time
element helium off
element nitrogen off
element carbon off
```

```
element neon off
element magnesium off
element silicon off
element sulphur off
element oxygen off
element argon off
element iron off
laser 1300
ionization -1
stop zone 1
print line faint -2
c
c helike_zn.in
c class limit
C =======================================
c
```

test emission for He-like Zn

`hhe_otspp` *plane parallel conservation and H-like emission for H, He*

```
title plane parallel conservation and H-like emission for H, He
c
c commands controlling continuum =========
ionization -2
blackbody 40000
c
c commands for density & abundances =========
hden 5
init file "hheonly.ini"
c
c commands controlling geometry =========
c expanding sphere to stop Lya from destroying HeI 23S
sphere
stop efrac 0.05 %stop when 5 percent ionized, so just beyond H I-front
c
c other commands for details =========
diffuse ots
atom h-like element hydrogen levels resolved 10
c need to iterate since we will assert helium triplet lines
c that depend on having stable solution
iterate
constant temper 7500
no induced processes (OK)
c
c commands controlling output =========
normalize to "Ca B" 4861 = 1 % normalize to exact result
punch overview "hhe_otspp.ovr" last
punch dr "hhe_otspp.dr" last
c
c helium lines
c hhe_otspp.in
c class limit
C =======================================
c
```

This tests the total emission from a spherical pure H + He Stromgren sphere using the outward only approximation. The conservation of the total number of ionizing photons, and the emitted spectrum, are all checked.

`hhe_otssp` *spherical conservation and H-like emission for H and He*

```
title spherical conservation and H-like emission for H and He
c
c commands controlling continuum =========
blackbody 40000
ionization -2
c
c commands for density & abundances =========
hden 5
init file "hheonly.ini"
c
c commands controlling geometry =========
diffuse ots
atom h-like levels resolved 10
radius 13
stop efrac 0.05 %stop when 5 percent ionized
sphere static
c
c other commands for details =========
constant temper 7500
iterate
no level2
no induced processes (OK)
c
c commands controlling output =========
normalize to "Ca B" 4861 = 1 % normalize to exact result
punch overview "hhe_otssp.ovr" last
punch dr "hhe_otssp.dr" last
c
c helium lines
c hhe_otssp.in
c class limit
c =======================================
c
```

This tests the total emission from a spherical pure H + He-like Stromgren sphere using the outward only approximation. The conservation of the total number of ionizing photons, and the emitted spectrum, are all checked. The geometry is plane paralel.

`hhe_outpp` *plane parallel conservation emission for H, He gas*

```
title plane parallel conservation emission for H, He gas
c
c commands controlling continuum =========
blackbody 40000
ionization -2
c
c commands for density & abundances =========
hden 5
init file "hheonly.ini"
c
c commands controlling geometry =========
c open sphere to stop Lya from destroying HeI 23S
sphere
stop efrac 0.05 %stop when 5 percent ionized
c
c other commands for details =========
```

```
c need this since will assert helium triplet lines that depend on stable soln
iterate
constant temper 7500
atom h-like element hydrogen levels resolved 10
no level2
no induced processes (OK)
c
c commands controlling output =========
normalize to "Ca B" 4861 = 1 % normalize to exact result
punch overview "hhe_outpp.ovr" last
punch dr "hhe_outpp.dr" last
c
c hhe_outpp.in
c class limit
c =======================================
c
```

This tests the total emission from a spherical pure hydrogen Stromgren sphere using the outward only approximation. The conservation of the total number of ionizing photons, and the emitted spectrum, are all checked.

`hhe_outppff` *plane parallel filling factor pure H, He gas*

```
title plane parallel filling factor pure H, He gas
c
c commands controlling continuum =========
ionization -2
blackbody 40000
c
c commands for density & abundances =========
hden 5
init file "hheonly.ini"
c
c commands controlling geometry =========
filling factor -1
stop efrac 0.05 %stop when 5 percent ionized
sphere
c
c other commands for details =========
c need this since will assert helium triplet lines that depend on stable soln
iterate
constant temper 7500
no level2
no induced processes (OK)
atom h-like element hydrogen levels resolved 10
c
c commands controlling output =========
normalize to "Ca B" 4861 = 1 % normalize to exact result
punch overview "hhe_outppff.ovr" last
punch dr "hhe_outppff.dr" last
c
c helium lines
c hhe_outppff.in
c class limit
c =======================================
c
```

This is a plane-parallel constant temperature cloud with only hydrogen and helium. The gas has a filling factor of 0.1. Induced processes are turned off and a large H atom is used so that the hydrogen recombination

spectrum will be close to Case B. The calculation stops beyond the hydrogen ionization front, because of the stop efrac command (it needs this since this is a constant temperature calculation, so the usual lower-temperature stopping criterion does not apply). The asserts confirm that energy is conserved and that the hydrogen spectrum is correct.

`hhe_outsp` *spherical conservation and H-like emission for H, He*

```
title spherical conservation and H-like emission for H, He
c
c commands controlling continuum =========
ionization -2
blackbody 40000
c
c commands for density & abundances =========
hden 5
init file "hheonly.ini"
c
c commands controlling geometry =========
radius 13
stop efrac 0.05 %stop when 5 percent ionized
sphere static
c
c other commands for details =========
iterate
diffuse outward only
constant temper 7500
no level2
no induced processes (OK)
atom h-like element hydrogen levels resolved 10
c
c commands controlling output =========
normalize to "Ca B" 4861 = 1 % normalize to exact result
punch overview "hhe_outsp.ovr" last
punch dr "hhe_outsp.dr" last
c
c helium lines
c
c hhe_outsp.in
c class limit
c ========================================
c
```

This tests a spherical cloud with only hydrogen and helium. Diffuse fields are transferred with the outward only approximation. The asserts check that the ionizing radiation is conserved.

`hii_blister` *Lexington 1995 dust-free hii blister region*

```
title Lexington 1995 dust-free hii blister region
c
c commands controlling continuum =========
c a simple blackbody
blackbody 40000
c the flux of H-ionizing photons
phi(h) 13.0
c add the CMB after hot star shape & luminosity specified
```

```
CMB
c
c commands for density & abundances =========
hden 4
init file="ism.ini"
c HII region abundance but no dust
abundances hii region no grains
abundances he1 c1 n1 o1 ne1 mg=.0001 si.01 s1 cl1 ar1
continue fe.001
c
c commands controlling geometry =========
sphere
c
c other commands for details =========
c this is to desaturate Lya to prevent excited state photoionization
c from being important - the other codes did not include this important
c physical process
turbulence 10 km/s
c must iterate since optically thick
iterate
c
c commands controlling output =========
normalize to "Ca B" 4861
print line faint .01
print line sum
He 1 5876
totl 2326
C 2 1335
TOTL 1909
N 2 6584
N 2 6548
N 3 57.21m
TOTL 7325
TOTL 3727
O 3 51.80m
O 3 5007
O 3 4959
Ne 2 12.81m
Ne 3 15.55m
Ne 3 3869
Ne 3 3968
S 3 18.67m
S 3 9532
S 3 9069
S 4 10.51m
end of lines
punch overview last "hii_blister.ovr"
punch dr last "hii_blister.dr"
punch results last "hii_blister.rlt"
print line optical depths
c
c
c hii_blister.in
c class hii
c =======================================
```

This is one of the test cases from the Lexington Meeting suite of nebulae (Ferland et al. 1995). It is a grain-free hii_blister HII region, similar to inner regions of the Orion Nebula, except for the absence of grains. The set of lines entered with the print line sum command lists the most powerful coolants in this model. This is one of the tabulated quantities in the Lexington Meeting, and is a fundamental test of energy conservation in the code. The ratio of the sum of these lines to Hb is equivalent to the Stoy ratio, used for determining stellar temperatures.

The "dielec kludge 0" command is to turn off my estimates of the DR rates for those elements that had none. This was only to allow comparison with other calculations that did not make similar estimates. For an actual calculation I would not include this command, since the guesses are better than nothing.

the turbulence is to stop the balmer lines from becoming optically thick since few other codes include an actual H atom, but use case b instead. The Orion HII region does have an observed turbulence of about 8 km/s.

This calculation stops near the H+ - H0 ionization front, where the temperature falls below the default lowest temperature of 4000 K. This model would have continued into the PDR had a lower temperature been specified with the STOP LOWEST TEMP command.

`hii_coolstar` *dust free cool HII region model, Lexington 1995*

```
title dust free cool HII region model, Lexington 1995
c
c commands controlling continuum =========
black body, T = 20000 K
q(h) 49
c add the CMB after hot star shape & luminosity specified
CMB
c
c commands for density & abundances =========
hden = 2
init file="ism.ini"
abund He-1 C-3.6576 N-4.39794 O-3.481146 ne-4.30103 mg-8
continue si-8 s-5.04576 cl-7 ar-8 fe-8
c
c commands controlling geometry =========
radius = 18.477121
sphere
c this is to go deep to pick up all H recombs
stop efract -2
stop temperature 1,000
c
c other commands for details =========
c must iterate since fine structure lines are optically thick
iterate
c assume there is neutral gas beyond what we compute
double
c
c commands controlling output =========
plot continuum range .1
print line faint .01
c this is the sum of lines in table 2 of the Lexington meeting
print line sum
N 2 6584
N 2 6548
TOTL 3727
Ne 2 12.81m
S 2 6720
S 3 18.67m
S 3 33.47m
S 3 9532
S 3 9069
end of lines
punch overview last "hii_coolstar.ovr"
punch dr last "hii_coolstar.dr"
```

```
punch results last "hii_coolstar.rlt"
punch continuum last units microns "hii_coolstar.con"
punch lines, cumulative, "hii_coolstar.cum"
totl 4861
o 3 5007
totl 3727
o 1 6300
end of lines
c
c Hbeta 4.93E36, L(total)4.30xHbeta
c hii_coolstar.in
c class hii
c =========================================
c
```

This is one of the test cases from the Lexington Meeting suite of nebulae. It is a grain-free HII region ionized by a very cool star. Hydrogen is ionized but not helium so this tests the transport of the H Lyman continuum. The set of lines is entered with the print line sum command to test energy conservation.

`hii_icf` *HII region with negative He/H ICF*

```
title HII region with negative He/H ICF
c
c commands controlling continuum =========
c this is the result of this command
c table star mihalas 46000
c and makes it possible to run these orion sims without
c installing the stellar atmosphere files
table read "star_mihalas_46000.dat"
ionization parameter -2.
c add the CMB after hot star shape & luminosity specified
CMB
c
c commands for density & abundances =========
hden 3
init file "ism.ini"
abundances ism no grains
grains no qheat single
metals and grains 0.320
c
c commands controlling geometry =========
stop efrac -3
stop temperature 15
sphere
c
c other commands for details =========
failures 1
c
c commands controlling output =========
punch overview "hii_icf.ovr" last
punch dr "icr.dr" last
c
c hii_icf.in
c class hii
c =========================================
```

This is an example of an H II region irradiated by a hard stellar continuum - one of the Mihalas NLTE stars. The hard continuum produces a negative He/H ionization correction factor, as discussed in Ballantyne, Ferland & Martin (2000). >>refer HeI icf by Ballantyne,D.R., Ferland, G.J., & Martin, P.G., 2000, ApJ 536,

773-777

`hii_paris` *New' Paris meeting HII region*

```
title "New" Paris meeting HII region
c "standard" HII region model of the Pequignot Meudon Conferance
c
c commands controlling continuum =========
black body, T = 40000 K radius = 12.113943
c add the CMB after hot star shape & luminosity specified
CMB
c
c commands for density & abundances =========
hden = 2
init file="ism.ini"
abund he -1 C-3.6576 N-4.39794 O-3.481146 ne-4.30103 mg-8
continue si-8 s-5.04576 cl=-7 ar-8 fe-8
c
c commands controlling geometry =========
radius = 18.477121
sphere
c next two to make sure we pick up all possible H recombination
* stop temperature 600
set temperature floor 1000
c make sure sim goes very deep into cold gas
stop temperature 100K
stop efrac -2
c
c other commands for details =========
c must iterate since fine structure lines are opticall thick
atom h-like element hydrogen levels resolved 10
iterate
c
c commands controlling output =========
normalize to "Ca B" 4861
print column densities
print line optical depths
plot continuum range .1
print line faint 0.01
print line sum
He 1 5876
totl 2326
TOTL 1909
N 2 121.7m
N 2 6584
N 2 6548
N 3 57.21m
TOTL 3727
O 3 5007
O 3 4959
O 3 51.80m
O 3 88.33m
Ne 2 12.81m
Ne 3 15.55m
Ne 3 3869
Ne 3 3968
S 2 6720
S 3 18.67m
S 3 33.47m
S 3 9532
S 3 9069
```

```
S 4 10.51m
end of lines
punch overview last "hii_paris.ovr"
punch hydrogen 21 cm last "hii_paris.21cm"
punch results last "hii_paris.rlt"
punch dr last "hii_paris.dr"
c this is to confirm that this commnand works
punch lines emissivity "hii_paris.str" last
TOTL 4861
end lines
c
table lines "LineList_HII.dat"
c hii_paris.in
c class hii
C ========================================
c
```

This is one of the "standard" models computed at the Paris and Lexington meetings on photoionization and shock calculations. a bable in hazy compares the predictions of the current version of CLOUDY with predictions of a few of the other codes. It is necessary to iterate since some fine structure lines are optically thick. The set of lines entered with the print line sum command is used to obtain the total luminosity in detected lines, a measure of the Stoy temperature.

Checks:
- Hb close to case B, Q(H) 4861, intensities.
- Enter answers in Table Error! Reference source not found..

---

hlike_c *H-like C VI case B*

```
title H-like C VI case B
c
c commands controlling continuum =========
table agn
ionization parameter 3
c
c commands for density & abundances =========
hden = 8
init file="ism.ini"
element carbon abundance 1
set eden 9
c
c commands controlling geometry =========
set dr -18
stop zone 1
c
c other commands for details =========
case b hummer no photoionization
no induced processes (OK)
constant temperature = 1.e5
c
c commands controlling output =========
punch results "hlike_c.rlt"
c
c hlike_c.in
c class limit
C ========================================
c
```

```
This test case compares the predictions of the multi-level H-like CVI atom
with the Storey and Hummer (1995) results. The set dr command sets the
zone thickness to 1 cm. The case b command sets Lyman line optical depths
to very large values.
Checks:
- Neutral fractions
- H* emissivity
- Relative line intensities
```

`hlike_o` *H-like O VIII case B*

```
title H-like O VIII case B
c
c commands controlling continuum =========
table agn
ionization parameter 4
c
c commands for density & abundances =========
hden = 8
init file="ism.ini"
element oxygen abundance 1
set eden 9
c
c commands controlling geometry =========
c set following so the n^2 dr is unity
set dr -18
stop zone 1
c
c other commands for details =========
case b hummer no photoionization
no induced processes (OK)
constant temperature = 1.e5
c
c commands controlling output =========
punch results "hlike_o.rlt"
c
c caseb_o.in
c class limit
c =======================================
c
```

```
This test case compares the predictions of the multi-level H-like O VIII atom
with the Storey and Hummer (1995) results. The set dr command sets the
zone thickness to 1 cm. The case b command sets Lyman line optical depths
to very large values.
Checks:
- Neutral fractions
- H* emissivity
- Relative line intensities
```

`igm_lalpha` *Ly alpha forest cloud*

```
title Ly alpha forest cloud
c
c commands controlling continuum =========
c ionized by cosmic background
```

```
c plus quasar/starburst continuum
cmb z=2
table hm05 z = 2
c
c commands for density & abundances =========
hden -2
metals -1.5 ;reduce all heavy elements by 1.5 dex
c
c commands controlling geometry =========
stop neutral column density 15
double ; mimic two-siced photoionization
c
c other commands for details =========
iterate to convergence; must iterate since optically thin
c
c commands controlling output =========
print line faint -1
punch overview last "igm_lalpha.ovr"
punch dr last "igm_lalpha.dr"
punch results last "igm_lalpha.rlt"
c
c on 05 aug 29 changed from old background command to
c cmb 2 plus hm05 z=2 background - all results changed substantially
c igm_lalpha.in
c class igm
c ========================================
c

This example demonstrates the behavior of the code in the low-density limit.
The ionizing source is the cosmic background at a redshift of z = 2.
The cloud is assumed to be optically thin to ionizing radiation,
in keeping with the Gunn-Peterson test, so the double command is included,
and an iteration is performed to converge the optical depth scale.
Continuum fluorescent excitation of lines is important because the cloud
is otically thin.
```

---

igm_primal *cloud with primordial abundances exposed to background at Z=10*

```
title cloud with primordial abundances exposed to background at Z=10
c
c commands controlling continuum =========
background, redshift= 10
c background due to agn from Haardt & Madau 1996 ApJ, 461, 20
c this sets both continuum shape and intensity, actually for z=2
table HM96 old
c
c other commands for details =========
c put in a velocity field to stop Lya from causing stability problems
turbulence 20 km/s
c do a second iteration to establish that it is optically thick downstream
iterate
c this sim should run cleanly - do not accept problems
failures 2
c
c commands for density & abundances =========
c this log of the hydrogen density
hden 5
c stored set of primordial abundances
abundances primordial
```

```
c
c commands controlling geometry =========
c we want to shielded face to not be exposed to empty space, but rather
c be just very deep in a dense cloud - this prevents lines from freely
c radiating from the shielded face
double optical depths
c set a lower temperature stopping criterion - default is 4000K
stop temperature 30
c
c commands controlling output =========
c don't want to print lots of faint hydrogen lines
print line faint -3
c some output files with info we want to save
punch temperature history "igm_primal_tem.his"
punch ionizing continuum "igm_primal.ion"
punch hydrogen conditions "igm_primal.mol"
punch heating "igm_primal.het"
punch dr "igm_primal.dr"
punch cooling "igm_primal.col"
punch overview "igm_primal.ovr"
punch results last "igm_primal.rlt"
c
c igm_primal.in
c class hii pdr igm
c ========================================

This is a high redshift cloud irradiated by the cosmic background and AGN light.
H- absorption of the z=10 cosmic background is a MAJOR heating agent
This model is very sensitive to treatment of Lya destruction, since resulting
J-bar is major destruction process for H-, which is the H2 formation route.
Lya ots oscillations could develop and are damped by not reevaluating H-
photo rate after first n times, as in hmole.c
```

igm_z3 *redshift 1000 recombination epoch*

```
title redshift 1000 recombination epoch
c
c commands controlling continuum =========
c a true blackbody radiation field, in strick thermo equilibrium
blackbody 3000 STE
c
c commands for density & abundances =========
hden 0
abundances primordial
c
c commands controlling geometry =========
stop zone 1
c
c other commands for details =========
case b
iterate
c
c commands controlling output =========
normalize lines to "Ca B" 4861
print ages
print departure coef
print line faint -5
c
c igm_z3.in
c class limit igm
c ========================================
```

c

This is a model of the universe near the recombination epoch, at a redshift of a thousand. The gas is exposed to a true blackbody at 3000 K, and the abundances are primordial.

---

ism *cloud irradiated by ism background*

```
title cloud irradiated by ism background
c
c commands controlling continuum =========
c this uses the Black continuum with no H-ionizing radiation,
c and with background cosmic rays
table ism
extinguish by a column of 22
c need cosmic rays to provide ionization to the chemistry
cosmic rays, background
c
c commands for density & abundances =========
hden 0
init file="ism.ini"
abundances ism
c
c commands controlling geometry =========
c set sphere since matter in all directions
sphere
stop temperature linear 10
stop thickness 0.1 linear parsecs
c
c other commands for details =========
c this prevents continuum pumping from exciting H0
case b
c this is to converge optical depths
iterate
c this should have no effect since T is larger than 100 K
set temperature floor 100K
c
c commands controlling output =========
c this model has no heavy element molecules, because H2 does not
c really fully form, since Lyman bands are not self-shielded
normalize to 157.6m "C 2"
print line pump
punch overview "ism.ovr" last
punch dr "ism.dr"
punch heating "ism.het"
punch coolign "ism.col"
c
c ism.in
c class ism
c =======================================
```

TODO - look at temperature struture - it has jitter at about the level of convergence, up and down. Temp jitter caused by eden jitter. This model is nearly isothermal, jitter measures noise in solver, and is great chance to pin this down.

This is a test of the behavior of the code in the extreme of photoionization by a relatively hard continuum, at low densities. The continuum is the galactic background, attenuated by a column density of 10^{22} cm-2. Ionization by galactic background cosmic rays is included. Case b appears since this region is deep in the ISM, and the Lyman lines are quite thick. This example checks whether the ionization balance, thermal

balance, and electron density sum, are performed correctly in this limit.

Checks:
- Numerical stability of solution
- Thickness exact

`ism_cosmicray` *background cosmic ray ionization by suprathermal electrons only*


```
title background cosmic ray ionization by suprathermal electrons only
c
c commands controlling continuum =========
c background CR H0 ionization rate should be 2.5e-17 s^-1,
c Williams et al. ApJ, 503, 689
cosmic rays, background
c this will be VERY faint black 50000
ionization parameter -25
black 50000
c
c commands for density & abundances =========
hden 5
c
c commands controlling geometry =========
set dr 0
stop zone 1
c
c other commands for details =========
iterate
set temperature floor 1000 K
c turn off molecules to test only ions
no molecules
c must turn off charge transfer and its heating, which would dominate,
c since we want to test cr heating and ionization
no charge transfer
no ctheat
c must turn of photoionization by recombining species,
c since we want to test cr heating and ionization
no photoionization
c any time photoionization is turned off, it is best to also
c turn off induced processes, which include pumping in the lyman lines
no induced processes
c
c commands controlling output =========
print ages
c
c ism_cosmicray.in
c class limit
c ========================================
c
```

This test conditions of cosmic ray ionization. Molecules and charge transfer are disabled so that analytical estimates can be made.

`ism_grid` *interstellar cloud irradiated by ism background*

```
title interstellar cloud irradiated by ism background
c
c commands controlling continuum =========
c this uses the Black continuum with no H-ionizing radiation,
c and with background cosmic rays
table ism
extinguish by a column of 22
cosmic rays, background
c
c commands for density & abundances =========
hden 0 vary
grid from -3 to 2 in 0.5 dex steps
init file="ism.ini"
c this turns on ism abundances and grains
abundances ism
c
c commands controlling geometry =========
c set sphere since matter in all directions
sphere
stop temperature linear 10
stop thickness 0.1 linear parsecs
stop zone 1
c
c other commands for details =========
c this prevents continuum pumping from exciting H0
case b
c
c commands controlling output =========
c this model has no heavy element molecules, because H2 does not
c really fully form, since Lyman bands are not self-shielded
normalize to 157.6m "C 2"
punch grid "ism_grid.grd" last
punch averages "ism_grid.avr" last
temperature hydrogen 1
end of averages
punch overview "ism_grid.ovr" last
punch dr "ism_grid.dr"
punch heating "ism_grid.het"
punch coolign "ism_grid.col"
punch xspec mtable "ism_grid.fit" range 0.1 3 keV
punch temperature "ism_grid.tem"
c
c this is a special version of the assert that does a series of models
c sert tempe hydro 1 9942 .05 grid 8575 6468 3606 776 304 169 103 66.2 44.1 31.3
c
c ism_grid.in
c class ism
c =========================================
```

this shows an S-curve calculation - make plot showing density as X-axis and gas pressure (nT) as y-axis

the gas is ionized by the galactic background. the density varies between 1e-3 and 100 cm-3. this is the full range found in the diffuse ism. The components that are produced are CNM - cold neutral medium, density ~ 40 cm-3 WNM - n ~ 0.5 cm-3, WIM - warm ionized medium, n ~ 0.25 cm-3 HIM - hot ionized medium, n ~ 1e-3 cm-3, calculation DOES NOT reproduce observed temperature of HIM - we get ~1e4K but observed is ~1e6 K. HIM is shock, not photo, ionized

---

`ism_hot_brems` *generate continuum due to hot ism in high Z,z starburst*

```
title generate continuum due to hot ism in high Z,z starburst
c
c commands controlling continuum =========
c this sets up coronal equilibrium for temperature and continuum
c cloud is predominantly collisionally ionized
coronal equilibrium, t = 1.e6 K
c
c commands for density & abundances =========
c log of hydrogen density (cm-3), by default a constant density model
hden 1
c chemical composition for a well-evolved star cluster with high Z
c the starburst command generates scale factors that will be used to
c multiply the old solar composition, used in ver 84. this is maintain
c record of physical changes
abundances old solar 84
abundances starburst Z=10 Zsun
c
c commands controlling geometry =========
c the log of the total hydrogen column density, cm-2
stop column density 21
c
c other commands for details =========
c continuum is plotted in Hzay, we me must iterate to predict this
iterate
c
c commands controlling output =========
c normalize to O 7 Lya
normalize to "O 7" 21.60
c this shortens the printout somewhat
print line faint 1
c output options punch continuum last 'ism_hot_brems.con'ho title, units keV
punch continuum last "ism_hot_brems.con" no title, units keV
punch sulphur ionization "ism_hot_brems.sul"
punch overview last "ism_hot_brems.ovr"
c this is only to document the problem described above
punch dr last "ism_hot_brems.dr"
c
c ism_hot_brems.in
c class ism
c ========================================
c
```

This model generates a large column constant density cloud similar to the hot phase of the interstellar medium. The continuum is punched to generate one of the figures in Part 2 of Hazy.

There is a strange feature between 1.7e-3A and 2.2e-3A that is the N emission (head starting at 1.7e-3A with O absorption at 2.2e-3A. This model is strongly enriched in heavies so many metal edges, esp O, are optically thick.

>>TODO 1 the guess of the thickness of the first zone is badly too small, because this model is collisionally ionized, and it used Stromgren length - better to use collisional balance and dr - as result of this the model takes far too many zones

---

`ism_jura` *check rate H2 forms on grain surfaces*

```
title check rate H2 forms on grain surfaces
c
```

```
c commands controlling continuum =========
c first continuum is FIR hot grain continuum produced in
c unmodeled HII Region
blackbody, t = 75 K
intensity 2.7 (total)
c this is hot star continuum
black 30000
intensity 1.90 range 0.4412 to 1 Ryd
c this will remove all ionizing radiation
extinguish 24 0
cosmic rays, background
c
c commands for density & abundances =========
hden 0
grains ism, abundance log 0.16 no qheat
init file="ism.ini"
abundances he -1.01 c -3.52 n-8 o-3.30 ne-8 mg-5.89
continue si -6.10 s -5.10 cl=-7 ar-8 fe -6.60
c
c commands controlling geometry =========
sphere
c stop when gas is fully neutral
stop efrac -10
c stop when gas is cold
stop temperature 10 linear
stop zone 1
c
c other commands for details =========
c set the gas temperature
constant temperature 100 K
case b
c
c commands controlling output =========
normalize to "C 2" 157.6m
c uv lines are strongly pumped by stellar continuum, break out contribution
print line pump
print line optical depths
print ages
punch hydrogen 21 cm last "ism_jura.21cm"
c
c ism_jura.in
c class limit
c =======================================
c
```

This model started out life as the Tielens & Hollenbach 1985 pdr. The density was set to unity and the incident radiation field adjusted so that the two default grains have temperatures near 100K. The model asserts that the H2 formation rate on grain surfaces is close to the //>>refer H2 grain physics Jura, M., 1975, ApJ, 197, 575 rate.

---

ism_opacity *generate standard ISM opacity curve*

```
title generate standard ISM opacity curve
c
c commands controlling continuum =========
table agn
ionization parameter -3
c
c commands for density & abundances =========
grains no qheat
```

```
c
c commands controlling geometry =========
set dr 0
stop zone 1
c
c other commands for details =========
iterate
c
c commands controlling output =========
punch total opacity last "ism_opacity.opc" units kev
c
c ism_opacity.in
c class limit
c =======================================
c
```

This example creates the file ism_opacity.opc which tabulates the total opacity of the gas as a function of energy. These plots are used in ISM studies to understand the transmission characteristics along a line of sight. The opacity depends on the dust to gas ratio, the gas phase abundances, and the level of ionization, all of which can be changed by altering parameters given above.

The model is of a 1 cm think parcel of gas which is optically thin in the Lyman continuum and Lyman lines. As a result the hydrogen emission line spectrum is close to case C. The model iterates so that the predicted ionization and emission know about this.

`ism_set_cr_rate` *background cosmic ray ionization by suprathermal electrons only*

```
title background cosmic ray ionization by suprathermal electrons only
c
c commands controlling continuum =========
cosmic ray rate -16
c this will be VERY faint black 50000
ionization parameter -25
black 50000
c
c commands for density & abundances =========
hden 5
element hydrogen ionization 1 0.00000001
c
c commands controlling geometry =========
set dr 0
stop zone 1
c
c other commands for details =========
constant temperature 1000 K
c turn off molecules to test only ions
no molecules
c must turn off charge transfer and its heating, which would dominate,
c since we want to test cr heating and ionization
no charge transfer
no ctheat
c must turn of photoionization by recombining species,
c since we want to test cr heating and ionization
no photoionization
c any time photoionization is turned off, it is best to also
c turn off induced processes, which include pumping in the lyman lines
no induced processes
```

```
c
c commands controlling output =========
print ages
c
c ism_set_cr_rate.in
c class limit
c =======================================
c
```

This test conditions of cosmic ray ionization. Molecules and charge transfer are disabled so that analytical estimates can be made.

`limit_casea_h_den13` *case A*

```
title case A
c
c Seaton, M.J, 1959 MN 119, 90,
c 4pi j(beta) 5.56E-26
c for this model total H-beta=4.745
c b(2)=3.73E-3 (3)=3.69E-2 (4)=0.091 (5)=0.145 (6)=0.193
c n.b. very different results if not l-mixed
c
c commands controlling continuum =========
black body, T = 50000 K
ionization parameter -2
c
c commands for density & abundances =========
hden = 13;
init file="honly.ini"
c
c commands controlling geometry =========
stop lyman optical depth -6
stop zone 1
set dr 0
c
c other commands for details =========
no induced processes (OK) ;N.B. case A is a fiction; no incuded transitions
atom h-like collisions off ; must turn off all collisions
constant temperature = 10000 K
iterate
c
c commands controlling output =========
c this should make the predicted totl Hbeta near unity for case a
normalize to "CaBo" 4861 = 2.23
print populations h-like
print departure coefficients
c
c limit_casea_h_den13.in
c class limit
c =======================================
c
```

Case A is a mathematical fiction; when the Lyman lines are optically thin continuum pumping must be important if the gas is ionized. Fluorescence is turned off with the no induced processes command. The density is set to a very high value (10^{15} cm-3) so that the 2s-2p states are well l-mixed, in keeping with standard case A assumptions. As a result, collisional excitation would dominate the level populations, and hydrogen collisions must be turned off with the hydrogen collisions off command. The Ly* optical depth is set to a small value. The set dr command sets the zone thickness to 1 cm. The abundances are set to a very small

value so that the electron density is equal to the hydrogen density.
Checks:
- Departure coefficients for H, He levels
- Neutral fractions
- H* emissivity

`limit_casea_h_den_temp` *Test model H in Case A limit*

```
title Test model H in Case A limit
c grid over nebular range of density and temperature to check Case A H
c
c commands controlling continuum =========
c create H+
laser 2 Ryd
ionization parameter -2
c
c commands for density & abundances =========
init "honly.ini"
hden 2. vary
grid 2 6 2
c
c commands controlling geometry =========
stop zone 1
c
c other commands for details =========
no scattering escape
case A hummer no Pdest no photoionization
no induced processes
constant temperature 4. vary
grid 5,000K to 20,000K in 5,000K steps
c
c commands controlling output =========
atom H-like print levels
c
c fixit the wavelength range is adjusted to avoid the Lyman lines
c the Lyman a / 2 photon ratio is wrong at high Tempetaure -
c is there a temperature dependent branching ratio to n=2 out
c of the collapsed level?
c
c limit_casea_h_den_temp.in
c class limit
c =========================================
c
```

this tests the predicted H I and He I spectra in the Case B limit. The grid is over both density and temperature.

`limit_caseb_h_hs87` *Case B from Hummer and Storey"*

```
title "Case B from Hummer and Storey"
init "honly.ini"
laser 1.1
ionization parameter -1
constant temperature 4
case b hummer no photoionization no pdest
atom h-like element hydrogen levels resolved 30
```

```
no induced processes
no level2
no scattering escape
set dr -20
hden 4
stop zone 1
iterate
print populaions h-like
print departure h-like
c
c
c limit_caseb_h_hs87.in
c class limit
```

`limit_caseb_h_lot` *log density case B, T=500 log n=2*

```
title log density case B, T=500 log n=2
c
c commands controlling continuum =========
black body, T = 2.e5 K
ionization parameter -1
c
c commands for density & abundances =========
hden = 2
init file="honly.ini"
c
c commands controlling geometry =========
stop zone 1
set dr 0
c
c other commands for details =========
constant temperature = 500
c want H and he fully stripped so we can check H He spectra
case b hummer no photoionization no pdest
no induced processes (OK)
iterate
c
c commands controlling output =========
print diffuse continua
normalize to "Ca B" 4861
c
c does not agree with Hummer & Storey, as explained by Ferguson & Ferland
c until 03 jun 02, used small atom, which had smaller differences
c turn on large atom to get best answer
atom h-like levels resolved 20
c
punch results "limit_caseb_h_lot.rlt" last
c
```

c limit_caseb_h_lot.in c class limit c === c

This tests the ionization and emission line spectrum for H case B at a low density and temperature.

`limit_caseb_h_n8` *h_caseb_n8 high density case B*

```
title h_caseb_n8 high density case B
```

```
c
c commands controlling continuum =========
black body, T = 2.e5 K
ionization parameter -1
c
c commands for density & abundances =========
hden = 8
init file="honly.ini"
c
c commands controlling geometry =========
stop zone 1
set dr 0
c
c other commands for details =========
c this will not affect the main model, but will test CR heating rate
cosmic ray background
case b hummer no photoionization no Pdest
no scattering escape
no induced processes (OK)
constant temperature = 20000
iterate
c
c commands controlling output =========
normalize to "Ca B" 4861
punch results "limit_caseb_h_n8.rlt"
punch ionizing continuum "limit_caseb_h_n8.cion"
c
c
c limit_caseb_h_n8.in
c class limit
c =========================================
c
```

This test case compares the predictions of the multi-level hydrogen atom with the Storey and Hummer (1995) results. The set dr command sets the zone thickness to 1 cm. The case b command sets Lyman line optical depths to very large values.
Checks:
- Neutral fractions
- H* emissivity
- Relative line intensities

high density causes disagreemeent with HS - collisions

`limit_caseb_he2_den8` *limit_caseb_he2_den8 He II case B*

```
title limit_caseb_he2_den8 He II case B
c
c commands controlling continuum =========
table agn
ionization parameter 0
c
c commands for density & abundances =========
hden = 8
element helium abundance 1
init file="ism.ini"
c
c commands controlling geometry =========
set dr 0
stop zone 1
```

```
c
c other commands for details =========
atom h-like element helium levels resolved 15
case b hummer no photoionization no Pdest
no induced processes (OK)
constant temperature = 50,000K
c
c commands controlling output =========
punch results "limit_caseb_he2_den8.rlt"
c
c
c limit_caseb_he2_den8.in
c class limit
c ========================================
c
```

This test case compares the predictions of the multi-level hydrogen atom with the Storey and Hummer (1995) results. The set dr command sets the zone thickness to 1 cm. The case b command sets Lyman line optical depths to very large values.
Checks:
- Neutral fractions
- H* emissivity
- Relative line intensities

`limit_caseb_he_den` *Test model H and He atoms in Case B limit*

```
title Test model H and He atoms in Case B limit
c grid over nebular range of density and temperature to check Case B H & He
c
c commands controlling continuum =========
laser 5 Ryd
ionization parameter -2
c
c commands for density & abundances =========
init "hheonly.ini"
hden 2. vary
grid 2 6 2
c
c commands controlling geometry =========
stop zone 1
c
c other commands for details =========
normalize to "He 2" 4686
case b hummer no Pdest no photoionization
no scattering escape
c prevent H I excited levels from being predicted
atom H-like element hydrogen levels 3
constant temperature 4.
c
c commands controlling output =========
atom H-like print levels
c
c
c limit_caseb_he_den.in
c class limit
c ========================================
c
```

this tests the predicted He II spectra in the Case B limit.

this effectively turns off hydrogen to avoid the problem with every other heII line lying beneath an HI line. this is done by reducing the number of levels for H I.

this asserts the values are within 9% for the standard T = 1e4K and a range of densities. Actually they are all nearly within a few percent except at the lowest temperature of 5,000K.

`limit_caseb_he_den4_temp4` *the best we can do to predict the HeI emission spectrum*

```
title the best we can do to predict the HeI emission spectrum
c
c commands controlling continuum =========
laser 2
ionization -1
c
c commands for density & abundances =========
hden 9000 linear
set eden 4
element helium abundance -1
init file "hheonly.ini"
c
c commands controlling geometry =========
stop zone 1
set dr -6.95424
c
c other commands for details =========
atom he-like levels 30
atom he-like collapsed levels 70
atom he-like gbar vriens
constant temper 4
case b no photoionzation no pdest
iterate
c
c commands controlling output =========
normalise to "He 1" 4471
print departure he-like helium
print line faint 0.01
c punch fits last "limit_caseb_he_den4_temp4.fit"
```
punch diffuse continuum l'imit_caseb_he_den4_temp4.dif'last no units microns
```
punch diffuse continuum "limit_caseb_he_den4_temp4.dif" last no title units microns
```
punch continuum l'imit_caseb_he_den4_temp4.con'last no units microns
```
punch continuum "limit_caseb_he_den4_temp4.con" last no title units microns
c
c
c limit_caseb_he_den4_temp4.in
c class limit
c =========================================
c
```

This is close to the best and most complete model of He I that the code can do. it is not hacked to try to reproduce HS?

`limit_caseb_he_den_temp` *Test model He atoms in Case B limit*

```
title Test model He atoms in Case B limit
c grid over nebular range of density and temperature to check Case B H & He
c
c commands controlling continuum =========
laser 5 Ryd
ionization parameter -2
c
c commands for density & abundances =========
init "hheonly.ini"
hden 2. vary
grid 2 6 2
c
c commands controlling geometry =========
stop zone 1
c
c other commands for details =========
normalize to "He 2" 4686
case b hummer no Pdest no photoionization
no scattering escape
c prevent H I excited levels from being predicted
atom H-like element hydrogen levels 3
constant temperature 4. vary
grid 5,000K to 20,000K in 5,000K steps
c
c commands controlling output =========
atom H-like print levels
c
c
c limit_caseb_he_den_temp.in
c class limit
c =========================================
c
```

this tests the predicted He II spectra in the Case B limit.

this effectively turns off hydrogen to avoid the problem with every other heII line lying beneath an HI line. this is done by reducing the number of levels for H I.

this asserts the values are within 15% for a range of density and temperature. Actually they are all nearly within a few percent except at the lowest temperature. The error greater than 10% occurs at the lowest temperature of 5,000K.

`limit_caseb_hhe_den` *Test model H and He atoms in Case B limit*

```
title Test model H and He atoms in Case B limit
c grid over nebular range of density and temperature to check Case B H & He
c
c commands controlling continuum =========
c create H+ and He+
laser 2.0 Ryd
ionization parameter -2
c
c commands for density & abundances =========
init "hheonly.ini"
hden 2. vary
grid 2 6 2
c
c commands controlling geometry ========
stop zone 1
```

```
c
c other commands for details =========
no scattering escape
case b hummer no Pdest no photoionization
constant temperature 4.
c
c commands controlling output =========
atom H-like print levels
c
c
c limit_caseb_hhe_den.in
c class limit
c =======================================
c
```

this tests the predicted H I and He I spectra in the Case B limit.

the grid is over density at the standard temperature of 1e4 K.

`limit_caseb_hhe_den_temp` *Test model H and He atoms in Case B limit*

```
title Test model H and He atoms in Case B limit
c grid over nebular range of density and temperature to check Case B H & He
c
c commands controlling continuum =========
c create H+ and He+
laser 2 Ryd
ionization parameter -2
c
c commands for density & abundances =========
init "hheonly.ini"
hden 2. vary
grid 2 6 2
c
c commands controlling geometry =========
stop zone 1
c
c other commands for details =========
no scattering escape
case b hummer no Pdest no photoionization
constant temperature 4. vary
grid 5,000K to 20,000K in 5,000K steps
c
c commands controlling output =========
atom H-like print levels
c
c
c limit_caseb_hhe_den_temp.in
c class limit
c =======================================
c
```

this tests the predicted H I and He I spectra in the Case B limit. The grid is over both density and temperature.

`limit_casec_h_den2` *H only optically thin in Lyman continuum*

```
title H only optically thin in Lyman continuum
c
c commands controlling continuum =========
blackbody 40000
ionization -2
c
c commands controlling geometry =========
stop thickness 16
c
c commands for density & abundances =========
hden 2
init file "honly.ini"
c
c other commands for details =========
iterate
constant temper 10000
no level2
c
c commands controlling output =========
print diffuse continuum
print line inward
normalize to "Ca B" 4861 = 1 % normalize to exact result
punch overview "limit_casec_h-den2.ovr" last
punch fine optical depths last "limit_casec_h-den2.fin" range 0.9 1.02 every 1
punch optical depths last "limit_casec_h-den2.opt"
punch continuum "limit_casec_h-den2.con" last units microns
punch dr "limit_casec_h-den2.dr" last
c
c limit_casec_h-den2.in
c class limit
c ========================================
c
```

This is a pure hydrogen cloud that is optically thin in the Lyman continuum. The asserts check the emission in several H I lines and continua. This should be close to what really happens in a low column density cloud exposed to a continuum source that does not have strong Lyman lines. (The continuum source used is a pure blackbody, and so has no lines). So this is an example of "Case C" emission >>refer H case C Ferland, G.J. 1999, PASP, 111, 1524

---

limit_casec_h_den5 *case C*

```
title case C
c
c commands controlling continuum =========
black body, T = 50000 K
ionization parameter -2
c
c commands for density & abundances =========
hden = 5
init file="hheonly.ini"
c
c commands controlling geometry =========
set dr -10
stop zone 1
c
c other commands for details =========
iterate
stop lyman optical depth -6
constant temperature = 10000 K
```

```
c
c commands controlling output =========
normalize to "Ca B" 4861
print h-like populations
print departure coefficients
c
c limit_casec_h_den5.in
c class limit
c =======================================
c
```

This is Case C, what really happens when optically thin gas is irradiated by a continuum with Lyman line continuum fluorescence allowed.
Checks:
- Departure coefficients for H, He levels
- Neutral fractions
- H* emissivity Case C is described in

>>refer H case C Ferland, G.J. 1999, PASP, 111, 1524

`limit_compton_hi_t` *test high-T Compton energy exchange*

```
title test high-T Compton energy exchange
c
c commands controlling continuum =========
c as hot as STE bb can be on IEEE 32-bit cpu
c equilibrium temperature should also be 2.51e7
c since gas radiated by true blackbody will equilibriate
c at its temperature
blackbody 7.4 STE
c
c commands for density & abundances =========
hden 10
c
c commands controlling geometry =========
stop zone 1
c
c other commands for details =========
atom h-like element oxygen levels 5
atom h-like element oxygen levels collapsed 30
set temperature convergence 0.0001
iterate
print h-like departure element oxygen
c
c commands controlling output =========
c
c limit_compton_hi_t.in
c class limit
c =======================================
c
```

This is the highest Compton temperature that can be computed in LTE on an IEEE 32-bit processor. This tests the code in the high-temperature Compton limit. Temperatures as high as 10^10 K can be computed successfully on CPUs with longer word lengths, such as a Cray or the new 64 bit processors.

Checks:

- The equilibrium temperature should be exactly 107.4 K (2.51239107 K).

---

limit_compton_lo_t *test low-T Compton energy exchange*

```
title test low-T Compton energy exchange
c
c commands controlling continuum =========
black linear 3 lte
brems 5
ionizat -5
c
c commands for density & abundances =========
hden -10
init file "hheonly.ini"
eden 0
c
c commands controlling geometry =========
stop zone 1
set dr 0
c
c other commands for details =========
set temperature convergence 0.0001
iterate 3
c
c commands controlling output =========
c
c limit_compton_lo_t.in
c class limit
c =========================================
c
```

This tests the code in the low temperature Compton limit. The gas is illuminated by a 3 K blackbody in thermodynamic equilibrium. The equilibrium temperature should be exactly 3 K. It is necessary to add an extra component of free electrons to test the code in this limit with the eden command.

---

limit_compton_mid_t *mid-T Compton energy exchange*

```
title mid-T Compton energy exchange
c test of thermal equil in limit_compton_mid_t limit; temp should EXACTLY equal 2E5K
c check continuum partition;
c energy range, photon densities, luminosities, follow
c 0.25-1. Q=26.6470 L=15.8190 c 1-1.807 Q=26.8867 L=16.3766
c 1.807-4 Q=27.3042 L=17.0948 c 4-20 Q=27.2041 L=17.3179
c 20 -- Q=22.9038 L=22.9038 c total lumin 17.5597
c nufnu(912A) = 1.8029E+16
c
c commands controlling continuum =========
black body t = 2.e5 K lte
c
c commands for density & abundances =========
hden = 6
init file "hheonly.ini"
c
c commands controlling geometry =========
stop zone 1
stop lyman continuum optical depth = -6
```

```
set dr 0
c
c other commands for details =========
set temperature convergence .0001 % decrease error on heat-cool match
c
c commands controlling output =========
print line faint .1
print departure coef
c
c limit_compton_mid_t.in
c class limit
c ========================================
c
```

This tests the behavior of the code in the Compton limit. The incident continuum is a blackbody in strict thermodynamic equilibrium. Strict thermodynamic equilibrium is expected for all constituents of the gas. The input stream also lists the expected photon fluxes for the incident continuum; this tests the normalization of the continuum, and its distribution. Grains are included to confirm their behavior in the LTE limit. The set dr command sets the zone thickness to 1 cm.

Checks:
- Luminosity, photon flux, over various energy intervals, 4*J at 912143.
- Electron temperature exactly 2*105 K.
- Grain temperature forced to 2*105 K by radiative processes.

---

limit_conserve *test that energy is limit_conserved*

```
title test that energy is limit_conserved
* trace temperature convergence 0 2
c
c commands controlling continuum =========
c primary continuum is a hot blackbody
black body, temp=5.5
intensity total -4
c add cosmic background
backbround
c
c commands controlling geometry =========
c don't stop the calculation until all energy used up
c in order to check energy conservation
stop temperature 3 linear
c cloud will go very deep with temperature near 3K without this
stop thickness 20
c
c commands for density & abundances =========
no molecules
metals -0.5
hden 1.0
constant pressure
c
c other commands for details =========
iterate
c want a clean calculation
failures 3
c this allows for more molecular gas beyond what we model
double optical depths
c must increase number of levels since so optically thick
```

```
atom co levels 30
c
c commands controlling output =========
punch pressure "limit_conserve.pre"
punch pressure history "limit_conserve.his"
punch cooling "limit_conserve.col"
punch heating "limit_conserve.het"
punch dr "limit_conserve.dr"
punch overview "limit_conserve.ovr1"
punch overview last "limit_conserve.ovr"
punch results "limit_conserve.rlt"
punch element carbon "limit_conserve.car"
punch element neon "limit_conserve.ne"
punch element magnesium "limit_conserve.mag"
punch element silicon "limit_conserve.sil"
punch molecules "limit_conserve.mol"
c
c limit_conserve.in
c class hii pdr
c ========================================
```

This checks that energy is limit_conserved. The code always checks that it did not radiate more energy than was absorbed. This calculation extends well past the photo-dissociation zone into fully molecular gas, so that all of the incident radiation is absorbed. Grains, CMB, & CRs are not present so that only the incident radiation field powers the gas.

Small changes can affect this model to surprising extents because of the presence of a major thermal front at the H0 - H+ transition region.

---

limit_eden *Martin Gaskell's funny model*

```
title Martin Gaskell's funny model
c used to test that electron density convergence is ok
c hydrogen line spectrum strongly pumped by continuum
c all elements > H have HIGH abundances
c ionization of elements Fe, Mg, Si strongly affected
c by charge transfer
c
c commands controlling continuum =========
black 4000
lumin 27.2
c
c commands for density & abundances =========
hden 5.138
abundances all 1000
c
c commands controlling geometry =========
radius 15
stop zone 1
set dr 0
c
c other commands for details =========
constant temper 5500
c
c commands controlling output =========
print line faint -3
c
c limit_eden.in
c class limit
```

```
c =======================================
c
```

This is mainly a test of the ability of the code to converge a model with a very strange electron density. The electrons are mainly contributed by heavy elements, and the gas is only slightly ionized.

Ionization of elements Fe, Mg, Si strongly affected by charge transfer with other heavy elements.

Checks:
- Electron density is correct.
- Hydrogen line spectrum strongly pumped by continuum.

`limit_h_induc` *constant temper black body limit from Ferland and Rees 1988*

```
title constant temper black body limit from Ferland and Rees 1988
c tests whether departure coef are forced to unity by induced processes
c
c commands controlling continuum =========
black body, t = 50000 lte
c
c commands for density & abundances =========
hden 10
init file="hheonly.ini"
c
c commands controlling geometry =========
stop lyman continuum optical depth -6
set dr -10
stop zone 1
c
c other commands for details =========
iterate
constant temper 50000
c
c commands controlling output =========
print populations h-like
print heating
print departure coef
print line faint -1
c
c limit_h_induc.in
c class limit
c =======================================
c
```

This example tests whether induced processes force level populations of hydrogen to LTE when they are irradiated by a blackbody in strict thermodynamic equilibrium. The density is low enough value for radiation to dominate the rate equations coupling levels with each other and the continuum. The expectation is for all departure coefficients to equal unity.

Checks:
- Departure coefficients exactly unity.
- Grain temperatures are exactly 5*104 K.

`limit_hi_ion` *very high ionization parameter limit*

```
title very high ionization parameter limit
c commands controlling continuum =========
table agn
ionization parameter 15
c commands controlling geometry =========
hden 0
stop zone 1
c other commands for details =========
iterate
c
c commands controlling output =========
punch heating "limit_hi_ion.het"
punch cooling "limit_hi_ion.col"
c
c
c func_hi_ion.in
c class function
c =======================================
c
```

This tests a limit of very high ionization

`limit_laser_1` *test of H ionization in optically thin limit*

```
title test of H ionization in optically thin limit
c
c commands controlling continuum =========
laser 1.5 Ryd
phi(h) 10
c
c commands for density & abundances =========
hden 1
init file="hheonly.ini"
c
c commands controlling geometry =========
set dr 0
stop zone 1
c
c other commands for details =========
constant temperature = 4
iterate to convergence
c
c commands controlling output =========
c
c limit_laser_1.in
c class limit
c =======================================
c
```

This checks the calculation of the hydrogen photoionization equilibrium. The continuum is a laser peaked at 1.5 Ryd, where the hydrogen photoionization cross section is 2.09*10-18 cm-2.

Checks:

- The hydrogen neutral fraction is nearly 2.00*10-4 (not exact since laser has finite width).
- Hb emissivity close to high density case A. The predicted TOTL 4861 intensity should be nearly 2.2 times the expected case B intensity.

H cross section is 2.09E-18 cm^2, rec coef is 4.18E-14 answer is neutral fraction 2.00E-4 also checks that only 3 iterations needed

---

`limit_laser_2` *test of H and HeI ionization in optically thin limit*

```
title test of H and HeI ionization in optically thin limit
c
c commands controlling continuum =========
laser 2.0 Ryd
intensity -0.3604
c
c commands for density & abundances =========
hden 1
init file="hheonly.ini"
abundances all -10
c
c commands controlling geometry =========
stop zone 1
set dr 0
c
c other commands for details =========
constant temperature = 4
iterate to convergence
c
c commands controlling output =========
c
c limit_laser_2.in
c class limit
c =======================================
c
```

This checks the calculation of the hydrogen and helium photoionization equilibrium. The continuum is a laser peaked at 2.0 Ryd, and so can only ionize hydrogen and atomic helium.

Checks:
- The hydrogen neutral fraction is nearly Ho/H+=4.51*10-4 (not exact since laser has finite width).
- Hb emissivity close to high density case A. The predicted TOTL 4861 intensity should be nearly 2.2 times the expected case B intensity.
- Helium ionization should be Heo/He+ = 6.61*10-4.

H cross section is 0.927E-18 cm^2, rec coef is 4.18E-13 answer is Ho/H+ = 4.51e-4 HeI cross section is 6.54E-18 cm^2, rec coef is 4.32E-13 answer is Heo/He+ = 6.61e-5

`limit_laser_200` *ionization in Auger-dominated limit*

```
title ionization in Auger-dominated limit
c
c commands controlling continuum =========
```

```
laser 200 Ryd
phi(h) 10
c
c commands for density & abundances =========
hden 1
c
c commands controlling geometry =========
stop zone 1
set dr 0
c
c other commands for details =========
iterate to convergence
constant temperature = 5
c
c commands controlling output =========
c
c limit_laser_200.in
c class limit
c ========================================
c
```

This checks the calculation of ionization equilibrium. The continuum is a laser peaked at 200 Ryd. It asserts ionization of C, O, and Fe. their ionization is dominated by the Auger effect.

Checks: Auger OK

`limit_laser_200_low` *test ionization in Auger-dominated limit*

```
title test ionization in Auger-dominated limit
c
c commands controlling continuum =========
laser 200 Ryd
phi(h) 5
c
c commands for density & abundances =========
hden 1
c
c commands controlling geometry =========
stop zone 1
set dr 0
c
c other commands for details =========
iterate to convergence
constant temperature = 4
c
c commands controlling output =========
c
c
c
c
c
c limit_laser_200_low.in
c class limit
c ========================================
c
```

This checks the calculation of ionization equilibrium. The continuum is a laser peaked at 200 Ryd. It asserts ionization of C, O, and Fe. their ionization is dominated by the Auger effect.

Checks: Auger OK

`limit_laser_3` *test H and He ionization in optically thin limit*

```
title test H and He ionization in optically thin limit
c
c commands controlling continuum =========
laser 4.3 Ryd
phi(h) 10
c
c commands for density & abundances =========
hden 1
abundances all -10
c
c commands controlling geometry =========
stop zone 1
set dr 0
c
c other commands for details =========
iterate to convergence
constant temperature = 4
c
c commands controlling output =========
c
c limit_laser_3.in
c class limit
c ========================================
c
```

This checks the calculation of the hydrogen and helium photoionization equilibrium. The continuum is a laser peaked at 4.3 Ryd, where it can fully ionize both hydrogen and helium.

Checks:
- The hydrogen neutral fraction is nearly 4.18*10-4 (not exact since laser has finite width).
- Helium ion: The ratio He+/He++ should be 1.69*10-3 and the ratio Heo/He+ should be 2.86*10-4.
- Hb emissivity should be close to high-density case A. The predicted TOTL 4861 intensity should be nearly 2.2 times the expected case B intensity.

H cross section is 1.0E-18 cm^2, rec coef is 4.18E-13 answer is n(Ho)/n(H+)=4.18e-3 HeI cross section is 1.51E-18 cm^2, rec coef is 4.32e-13 answer is n(Heo)/n(He+)=2.86e-4, so Heo/He = 4.83e-7 HeII cross section is 1.30E-18 cm^2, rec coef is 2.20e-12 answer is n(He+)/n(He2+)=1.69e-3

`limit_lowd0` *test low density limit*

```
title test low density limit
c this and limit_lowdm6 should get same results
c
c commands controlling continuum =========
table agn
ionization parameter -2
c
```

```
c commands for density & abundances =========
hden 0
abundances old solar 84
c
c commands controlling geometry =========
stop zone 1
set dr 0
c
c other commands for details =========
iterate
c
c commands controlling output =========
normalize to "H 1" 4861 0.86
print line sort wavelength range 100 1m
punch cooling last "limit_lowd0.col"
punch continuum last "limit_lowd0.con"
c
c all asserts for both sims are in this file
init file="lowd.dat"
c
c limit_lowd0.in
c class limit
c ========================================
c
```

this test case is paired with lowdm6.in both tests read in the same set of asserts, those contained in the file lowd.dat, and they should get exactly the same answer

this is also the test of the print lines intensity command

---

limit_lowden *test optically thin model that extends to very low densities*

```
title test optically thin model that extends to very low densities
c
c commands controlling continuum =========
table agn
ionization parameter 0
c
c commands for density & abundances =========
hden -2 to the -2 power
init file="ism.ini"
c
c commands controlling geometry =========
sphere
c this will take density down to 1e-8, below stated limit
radius 10 13
c
c other commands for details =========
c
c commands controlling output =========
punch overview "limit_lowden.ovr" last
c
c limit_lowden.in
c class limit
c ========================================
c
```

This model is optically thin, with density falling off as inverse square law, so ionization and temperature

should be nearly constant. if outer radius increased by 2 dex problem with level3 will appear, several li seq lines (OVI, NeVIII) will fluctuate when density about 1e-9

We do not assert H lines since the cloud is optically thin and takes at least three iterations to converge optical depth scale, That is not the purpose of this sim

---

`limit_lowdm6` *test low density limit*

```
title test low density limit
c this and limit_lowd0 should get same results
c
c commands controlling continuum =========
table agn
ionization parameter -2
c
c commands for density & abundances =========
hden -6
abundances old solar 84
c
c commands controlling geometry =========
stop zone 1
set dr 0
c
c other commands for details =========
iterate
c
c commands controlling output =========
normalize to "H 1" 4861 0.86
print line sort wavelength range 100 1m
punch cooling last "limit_lowdm6.col"
punch continuum last "limit_lowdm6.con"
c
c all asserts are in this file
init file="lowd.dat"
c
c limit_lowdm6.in
c class limit
c =======================================
c
```

this test case is paired with lowd0.in both tests read in the same set of asserts, those contained in the file lowd.dat, and they should get exactly the same answer

this also tests the print line sort range command

`limit_lowion_low` *test conditions of very low ionization matrix/simple solver*

```
title test conditions of very low ionization matrix/simple solver
c
c commands controlling continuum =========
blackbody 50000
ionization parameter -30
c
```

```
c commands for density & abundances =========
hden 2
init file "hheonly.ini"
c
c commands controlling geometry =========
stop zone 1
c
c other commands for details =========
atom h-like matrix lowt
atom he-like matrix lowt
no molecules
constant temperature 100
c
c commands controlling output =========
c
c limit_lowion_low.in
c class limit
c =======================================
c
```

The lowion_pops.in and limit_lowion_low.in models form a pair that have identical boundary conditions but use the two different hydrogenic level populations solvers. The results should agree. lowion_pops.in uses the full solution with the associate matrix inversion. This can fail under conditions of extreme low ionization due to numerical instabilities and roundoff. The solver used in limit_lowion_low.in is much simpler and will work for any conditions.

The model is almost totally molecular.

---

limit_lowion_pops *test conditions of very low ionization matrix/simple solver*

```
title test conditions of very low ionization matrix/simple solver
c
c commands controlling continuum =========
blackbody 50000
ionization parameter -30
c
c commands for density & abundances =========
hden 2
init file "hheonly.ini"
c
c commands controlling geometry =========
stop zone 1
c
c other commands for details =========
atom h-like matrix pops
atom he-like matrix pops
no molecules
constant temperature 100
c
c commands controlling output =========
c
c
c limit_lowion_pops.in
c class limit
c =======================================
c
```

The limit_lowion_pops.in and lowion_low.in models form a pair that have identical boundary conditions but use the two different hydrogenic level populations solvers. The results should agree. limit_lowion_pops.in uses the full solution with the associate matrix inversion. This can fail under conditions of extreme low ionization due to numerical instabilities and roundoff. The solver used in lowion_low.in is much simpler and will work for any conditions.

The model is almost totally molecular.

`limit_lte_h_t50_cion` *collisionally ionized H in LTE limit*

```
title collisionally ionized H in LTE limit
c
c commands controlling continuum =========
black body, T = 50000 K
ionization parameter -8
c
c commands for density & abundances =========
hden = 20
set eden 20
init file="hheonly.ini"
c
c commands controlling geometry =========
stop lyman optical depth -6
set dr -10
stop zone 1
c
c other commands for details =========
constant temperature = 50000 K
c want pure collisional model
no induced processes (OK)
atom h-like collisional excitation off
atom h-like collisions l-mixing off
atom he-like collisional excitation off
atom he-like collisions l-mixing off
atom he-like collapsed levels 20
iterate
c
c commands controlling output =========
print h-like departure coefficients
print he-like departure coefficients
print populations h-like
print departure coefficients
c
c limit_lte_h_t50_cion.in
c class limit
c =======================================
c
```

This is the limiting case pure hydrogen collisional ionization, There are no excitation or l-mixing collisions, so this tests whether collisional ionization - three body recombination works in detailed balance.

`limit_lte_h_t50_coll` *collisionally excited H in LTE limit*

```
title collisionally excited H in LTE limit
c
```

```
c commands controlling continuum =========
black body, T = 50000 K
ionization parameter -8
c
c commands for density & abundances =========
hden = 20
init file="hheonly.ini"
c
c commands controlling geometry =========
stop lyman optical depth -6
stop zone 1
set dr 0
set eden 20
c
c other commands for details =========
no induced processes (OK)
atom h-like collisions l-mixing off
atom he-like collapsed levels 20
atom he-like collisions l-mixing off
constant temperature = 50000 K
iterate
c
c commands controlling output =========
print h-like departure coefficients
print h-like populations
print he-like departure coefficients
print populations h-like
c
c limit_lte_h_t50_coll.in
c class limit
c =========================================
c
```

This checks that the model H atom goes to LTE at high densities.

chng 06 aug 24, had not included collisional ionization, and so he-like departure coefficients were very large, around 202. comments said there were problems. turned on collisional ionization, no problems noted

chng 06 jul 22 with RP changes in high-n n-changing collisions the rates are now much smaller - needed to change density to be far higher and several quantities changed. at lower density (1e18 cm-3) the populations are very unphysical and runaway maser now occurs. this is only a homework problem and intended to only test n-changing collisions. with higher density this test is done.

`limit_lte_he1_coll` *He atom at high densities*

```
title He atom at high densities
c
c commands controlling continuum =========
laser 2
ionization -3
c
c commands for density & abundances =========
hden 18
set eden 18
element helium abundance -1
init file "hheonly.ini"
c
c commands controlling geometry =========
stop zone 1
```

```
set dr -10
c
c other commands for details =========
atom he-like levels 10
atom he-like collapsed levels 20
constant temper 20000 K
iterate
c
c commands controlling output =========
print he-like departure helium
normalise to "He 1" 4471
print line faint -2
c
c limit_lte_he1_coll.in
c class limit
c =======================================
c
```

test whether he-like ion populations go to lte in high density limit. The level populations should be in LTE, and the departure coefficients should be unity.

`limit_lte_he1_ste` *He atom in LTE at high densities*

```
title He atom in LTE at high densities
c
c commands controlling continuum =========
blackbody 50000 STE
c
c commands for density & abundances =========
hden 18
* set eden 18
c a high he abundance so that it dominates
element helium abundance 1
init file "hheonly.ini"
c
c commands controlling geometry =========
stop zone 1
c
c other commands for details =========
atom he-like levels 7
atom he-like collapsed levels 30
iterate
c
c commands controlling output =========
print he-like departure helium
print h-like departure hydrogen
normalise to "He 1" 4471
print line faint -2
punch heating "limit_lte_he1_ste.het"
punch cooling "limit_lte_he1_ste.col"
punch dr "limit_lte_he1_ste.dr"
c
c limit_lte_he1_ste.in
c class limit
c =======================================
c
```

test whether a gas dominated by He goes to LTE in high-density limit. The level populations should be in LTE, the departure coefficients should be unity, and the temperature equal to the BB temp.

`limit_lte_hhe_coll_t50` *H, He in LTE at high densities*

```
title H, He in LTE at high densities
c test from Ferland and Rees 88, collisions drive H to LTE
c collisions should drive all departure coef to unity
c
c commands controlling continuum =========
black body t = 50000
ionization parameter -5
c
c commands for density & abundances =========
hden = 19
init file="hheonly.ini"
set eden 20
c
c commands controlling geometry =========
set dr -10
stop zone 1
stop lyman optical depth -6
c
c other commands for details =========
constant temperature = 50000
iterate
c
c commands controlling output =========
print departure coefficients h-like
print populations h-like
print departure coefficients h-like helium
print populations h-like helium
print departure coefficients he-like
print populations he-like
punch dr "limit_lte_hhe_coll_t50.dr" last
c
c limit_lte_hhe_coll_t50.in
c class limit
c =========================================
c
```

This model is a test of the behavior of hydrogen and helium in the high density, collision dominated, limit. The temperature is preset, the hydrogen density is set to a very high value, and the ionization parameter is very low. The resulting model is collision dominated, so this case checks that the collision physics occurs in detailed balance. The predicted departure coefficients should all equal unity. The set dr command sets the zone thickness to 1 cm.

Checks:
- Hydrogen departure coefficients exactly unity.
- Helium departure coefficients near unity. (Density not high enough to bring helium departure coefficients exactly to unity.)
- H-, H2, H2+ H3+, and HeH+ departure coefficients exactly unity. `

`limit_lte_hhe_induc` *radiation dominated H, He gas goes to STE*

```
title radiation dominated H, He gas goes to STE
c
c commands controlling continuum =========
```

```
black body, t = 50000 STE
c
c commands for density & abundances =========
hden 4
init file="hheonly.ini"
c
c commands controlling geometry =========
stop lyman continuum optical depth -6
set dr -10
stop zone 1
c
c other commands for details =========
iterate
set temperature convergence 0.002
c
c commands controlling output =========
print populations h-like
print heating
print departure coef
print line faint -1
c
c limit_lte_hhe_induc.in
c class limit
c =======================================
c
```

This is a H, He-only gas that is optically thin in the Lyman continuum. It is irradiated by a blackbody in strict thermodynamic equilibrium. The tests confirm that the gas temperature equilibriates close to the black body temperature.

`limit_lte_hhe_ste` *H, He in STE*

```
title H, He in STE
c from Ferland and Rees 1988
c this tests whether thermal processes go to STE
c
c commands controlling continuum =========
black body, t = 50000 STE
c
c commands for density & abundances =========
hden 10
init file="hheonly.ini"
c
c commands controlling geometry =========
stop lyman optical depth -6
set dr -10
stop zone 1
c
c other commands for details =========
iterate
c
c commands controlling output =========
print heating
print populations h-like
print departure coef h-like
print departure coef he-like
c
c limit_lte_hhe_ste.in
c class limit
c =======================================
```

```
c
```

This is the ultimate test of the behavior of the code in the strict thermodynamic equilibrium limit. The temperature is not held constant, so the resulting equilibrium temperature determines whether cooling processes are treated properly in the detailed balance limit. The equilibrium temperature should be exactly $5*10^4$ K, and all departure coefficients should equal unity. A small amount of grains are included to check that the grain thermal balance is handled properly in this limit.

Checks:
- Electron temperature exactly $5*10^4$ K.
- Departure coefficients unity.

`limit_lte_hminus` *H- goes to LTE*

```
title H- goes to LTE
c
c commands controlling continuum =========
c this is a second way to get STE and tests the energy density option
blackbody 5000 energy density 5000K
c
c commands for density & abundances =========
hden 10
c
c commands controlling geometry =========
stop zone 1
set dr 0
c
c other commands for details =========
iterate
constant temperature 5000
c
c commands controlling output =========
print populations h-like hydrogen
print departure coef
c
c limit_lte_hminusin
c
c class limit
c ========================================
c
```

This checks that the negative hydrogen ion goes to thermodynamic equilibrium when irradiated by a blackbody in thermodynamic equlibrium. It was originally presented in >>refer H- test Ferland, G. J., & Persson, S. E. 1989, ApJ, 347, 656

`limit_lte_metal` *STE with metals*

```
title STE with metals
c this tests whether thermal processes go to STE
c
c commands controlling continuum =========
c a dilution factor of 1 is one way to get STE
black body, t = 20000 dilution 1
```

```
c
c commands for density & abundances =========
hden 10
abundances starburst 5
c
c commands controlling geometry =========
stop zone 1
stop lyman optical depth -6
set dr -10
c
c other commands for details =========
iterate
c
c commands controlling output =========
print departure coef
print populations h-like
punch heating "limit_lte_metal.het" last
punch cooling "limit_lte_metal.col" last
c
c limit_lte_metal.in
c class limit
c ========================================
c
```

This checks that the code goes to strict thermodynamic equilibrium for the case of a metal rich gas exposed to a true black body. The many heavy element lines should dominate cooling, so this is a test that the multilevel atoms go to LTE in the radiation-dominated limit.

Checks:
- Temperature should equilibrate at 20000 K.
- Departure coefficients should equal unity.

limit_recoil_ion *test compton recoil ionization of hydrogen*

```
title test compton recoil ionization of hydrogen
c
c commands controlling continuum =========
laser 2000 ryd
ionization parameter -5
c
c commands for density & abundances =========
hden 0
init "honly.ini"
c
c commands controlling geometry =========
set dr 14
stop zone 10
c
c other commands for details =========
iterate
no secondary ionization
constant temperature 1000
c
c commands controlling output =========
c
c
c limit_recoil_ion.in
c class limit
```

```
c ======================================
c
```

H ionization is totally due to recoil ionization in this model. The assert checks the final hydrogen ionization.

`limit_strom` *pure-H Stromgren sphere*

```
title pure-H Stromgren sphere
c the answer is R(Stromgren) = 4.16E17 cm
c
c commands controlling continuum =========
blackbody 50000 K
q(h) 49
c
c commands for density & abundances =========
hden 4
init file "honly.ini"
abundances all -10
c
c commands controlling geometry =========
sphere static
radius 16
stop efrac -0.5
c
c other commands for details =========
no level2
turbulence 20 km/s
constant temper 7500
iterate
c
c commands controlling output =========
punch overview last "limit_strom.ovr"
punch results last "limit_strom.rlt"
punch dr last "limit_strom.dr"
c
c limit_strom.in
c class limit
c ======================================
c
```

This case checks that the code computes the geometry and emissivity correctly for a pure hydrogen spherical shell. The low temperature is chosen to avoid collisional ionization. The model stops at the Ho-H+ ionization front. The turbulence is to prevent the Balmer lines from becoming optically thick.
Checks
- Outer radius should be 4.16391017 cm.
- Predicted Hb, case B Hb, and Q(H) Hb, all agree.

`limit_supra` *secondary ionization dominated gas*

```
title secondary ionization dominated gas
c like SN envelope
c
c commands controlling continuum =========
c this continuum will be VERY faint
table agn
```

```
ionization parameter -25
c secondary ioinzations will dominate
set csupra -5
c
c commands for density & abundances =========
hden 5
abundances old solar 84
c
c commands controlling geometry =========
stop temperature 20
stop zone 1
set dr 0
c
c other commands for details =========
constant temperature 1000 K
iterate convergence
c
c commands controlling output =========
print ages
c check that no uninitialized DR rates are printed (r2757)
print arrays carbon only
punch heating "limit_supra.het"
punch ionizing continuum "limit_supra.ion" last
punch molecules "limit_supra.mol"
c
c
c limit_supra.in
c class limit
c ========================================
c
```

This model computes the ionization within cool gas that is totally ionized by suprathermal secondary electrons.

Charge transfer heating is VERY important in this simulation.

`limit_vbhum` *compare with Van Blerkom and Hummer exact RT results*

```
title compare with Van Blerkom and Hummer exact RT results
c
c commands controlling continuum =========
blackbody 50000 K
phi(h) 12.30103
c commands for density & abundances =========
hden 4
init "honly.ini"
abundances all -10
c
c commands controlling geometry =========
stop efrac -0.5
c
c other commands for details =========
diffuse ots
no level2 lines
constant temper 4
iterate
c
c commands controlling output =========
```

```
punch overview last "limit_vbhum.ovr"
punch dr last "limit_vbhum.dr"
c
c
c limit_vbhum.in
c
c class limit
c =======================================
c
```

This is a test of the treatment of the diffuse fields, their transfer, and their effects on the ionization structure of a nebula. The comparison is made against the exact calculation published by Van Blerkom and Hummer (1967). The geometry is open, that is, similar to that assumed in most BLR calculations.

>>refer H ionization Van Blerkom, D., & Hummer, D. G. 1967, MNRAS, 137, 353

The diffuse ots command is entered in order to reproduce the Van Blerkom and Hummer results. The default assumption, outward only, does not agree as well. I changed the default from OTS to outward only to be in better agreement with predictions by Harrington and Rubin at the Lexington meeting. They have not checked whether their codes are in agreement with the Van Blerkom and Hummer paper.

Checks:
- Neutral fraction at illuminated face 5.8*10-4.
- Location of ionization front at 7.8*1016 cm.
- 34TOTL 486134 and 34CA B 486134 agree; both slightly lower than 34Q(H) 486134.
- Answers with OTS agree with 1967 results.

test hydrogen ground state rec effic against vb+h exact results this is their case e) - "zero condition" their answer for H0/Htot at the illuminated edge is approx 5.8E-4, and a Stromgren radius of approximately 7.7E16 cm

---

limit_veryveryfast *very fast simulation for Purify/valgrind*

```
title very fast simulation for Purify/valgrind
c
c commands controlling continuum =========
table agn
ionization parameter -2
c
no file opacity
set dr 0
c
c commands for density & abundances =========
hden 9
init file "hheonly.ini"
element oxygen on
c
c commands controlling geometry =========
stop zone 1
c
c other commands for details =========
no level 2
constant temperature 4
atom h-like levels very small
atom he-like levels very small
```

```
c
c commands controlling output =========
punch dr last "limit_veryveryfast.dr"
c
c
c limit_veryveryfast
c class limit
c ========================================
c
```

This is meant to be a very fast calculation to use when running extensive debug-enabled runtimes.

---

nlr_lex00 *NLR model for Lexington 2000 Meeting*

```
title NLR model for Lexington 2000 Meeting
c
c commands controlling continuum =========
phi(h) 12.47712
c shape of continuum is interpolated table
interpolate (0 -10) (0.08 -10) (0.1 1) (3676 -4.935) (4700 -11)
continue (4750 -20) (7.4e6 -30)
c
c commands controlling geometry =========
stop column 22
c
c commands for density & abundances =========
hden 4
init file="c84.ini"
abundances -1 c-3.52 n-4 o-3.097 ne-4 na-9
continue mg-4.523 al-8 si-4.523 s-4.824 a-9 ca-8 fe-8 ni-9
c
c other commands for details =========
iterate
c
c commands controlling output =========
* normalize to "totl" 4861 of 0.923
print line faint .01
print diffuse continuum
c add sum of these lines to printout for Stoy ratio
print line sum
TOTL 1216
He 1 5876
TOTL 1909
TOTL 1549
N 2 6584
N 2 6548
N 3 57.21m
totl 1750
totl 1486
o 1 6300
o 1 63.17m
TOTL 3727
totl 1665
O 3 51.80m
O 3 5007
O 3 4959
totl 1402
Ne 3 15.55m
Ne 3 3869
Ne 3 3968
Ne 4 2424
```

```
Ne 5 3426
Ne 5 3346
totl 2798
si 2 34.81m
s 2 6720
S 3 18.67m
S 3 9532
S 3 9069
S 4 10.51m
end of lines
c
punch overview last "nlr_lex00.ovr"
punch dr last "nlr_lex00.dr"
punch results last "nlr_lex00.rlt"
c
table lines "LineList_NLR.dat"
c nlr_lex00.in
c class nlr
c ========================================
```

This is one of the test cases from the Lexington (1993) Meeting suite of nebulae. It is a grain-free NLR model.

---

nlr_liner *NLR liner model*

```
title NLR liner model
c a constant pressure (gas+radiation) model of a nlr_liner cloud
c in the spirit of Ferland and Netzer 1983
c
c commands controlling continuum =========
table agn
ionization parameter -3
c
c commands for density & abundances =========
hden 6
init file "ism.ini"
abundances old solar 84
metals 0.3
c
c commands controlling geometry =========
stop thickness 16.3
stop column density 23
constant pressure
c
c other commands for details =========
iterate to convergence ;must iterate to get radiation pressure correctly
c
c commands controlling output =========
punch overview last "nlr_liner.ovr"
punch element nitrogen last "nlr_liner.nit"
punch dr last "nlr_liner.dr"
punch results last "nlr_liner.rlt"
c
c nlr_liner.in
c class nlr
c ========================================
```

This is a model somewhat like the Liner parameters proposed by Ferland and Netzer (1983). A second iteration is performed to allow the calculation of the line radiation pressure.

>>refer nlr_liner model Ferland, G. J., & Netzer, H. 1983, ApJ, 264, 105

---

nlr_liner_grains *liner model with grains*

```
title liner model with grains
c
c a constant pressure (gas+radiation) model of a liner cloud
c in the spirit of Ferland and Netzer 1983
c
c commands controlling continuum =========
ionization parameter -3
table agn
c
c commands for density & abundances =========
hden 6
c this turns on both grains and depleted abundances
abundances ism no grains
grains ism no qheat single
init file "ism.ini"
c
c commands controlling geometry =========
stop column density 23
c
c other commands for details =========
constant pressure
iterate to convergence ;must iterate to get radiation pressure correctly
c
c commands controlling output =========
punch overview last "nlr_liner_grains.ovr"
punch dr last "nlr_liner_grains.dr"
punch continuum last "nlr_liner_grains.con" units microns
punch results last "nlr_liner_grains.rlt"
c
c nlr_liner_grains.in
c class nlr
c =======================================
c
```

This is a model somewhat like the Liner parameters proposed by Ferland and Netzer (1983). A second
iteration is performed to allow the calculation of the line radiation pressure. It includes grains and is so more
realistic.

>>refer liner model Ferland, G. J., & Netzer, H. 1983, ApJ, 264, 105

---

nlr_paris *Paris meeting NLR model*

```
title Paris meeting NLR model
c
c commands controlling continuum =========
interpolate (0 -10) (0.073 -10) (0.074 1) (7352 -4)
continue (7400 -15) (7.353e6 -20)
ionization parameter -2
c
c commands controlling geometry =========
stop lyman continuum optical depth 4
c
```

```
c commands for density & abundances =========
hden 3
init file="c84.ini"
abund -1 C-3.5229 N-4. O-3.22185 ne-3.82391 na=-8 mg-4.5229 al=-8
continue si-4.5229 s-4.82391 ar-8 ca=-8 fe-7 ni-8
c
c other commands for details =========
c this large group of lines are turned off so that the model runs
c 30 percent faster - lines are mainly fuv and x-ray, which we will
c not track in this model
no level2
iterate
atom h-like element helium levels resolved 10
c
c commands controlling output =========
punch overview last "nlr_paris.ovr"
punch dr last "nlr_paris.dr"
punch results last "nlr_paris.rlt"
c
c nlr_paris.in
c class nlr
c =======================================
```

This is the NLR model presented in the Meudon meeting on model nebulae. The init file is entered to make the code behave more like version 84.

Checks:
- init file works

---

nova_dqher *cold nova shell*


```
title cold nova shell
c Ferland et al. 1984 DQ Her
c model of room temperature ionized cloud arund old nova DQ Her
c roughly that of
c tests behaviour of code at very low temperatures
c
c commands controlling continuum =========
c flat continuum, followed by nu^-2 power law
interpolate (0 0) (0.3 0) (8.e6 -14.8)
luminosity total 34
c
c commands for density & abundances =========
hden = 2.
abundances nova
grains no qheat single
c
c commands controlling geometry =========
filling factor 0.667
covering factor 0.667
radius 16.5682 , thickness = 16.14613
c need this since gas is so cool
stop temp = 100
c
c other commands for details =========
# >>chng 06 aug 02, add cosmic rays, should always have been included
cosmic rays background
iterate
turbulence 300
```

```
age 70 years %it exploded in the 1930's
atom h-like element hydrogen levels resolved 15
atom h-like element helium levels resolved 15
c
c commands controlling output =========
normalize to 4861 label="CaBo" intensity 1.169
print line all
punch overview last "nova_dqher.ovr"
punch results last "nova_dqher.rlt"
punch dr last "nova_dqher.dr"
punch transmitted continuum "nova_dqher.trn" last
c
c nova_dqher.in
c class nova
c ========================================
c
```

This tests the code39s behavior in the limit posed by the metal rich low density nebula surrounding DQ Her (Ferland et al. 1984).
Checks:
- Thickness exact
- Thermal stability High-Z gas ionization at low temperature

`nova_photos` *dense nova photosphere*

```
title dense nova photosphere
c
c commands controlling continuum =========
c this is the result of this command
c table star kurucz 35000; use old atlas atmosphere
c and makes it possible to run sims without
c installing the stellar atmosphere files
c set SED of central star
table read "star_kurucz_35000.dat"
c set luminosity of central star
absolute bolometric magnitude -8.1
c
c commands for density & abundances =========
hden 10 ;log of total hydrogen density (cm-3) at illuminated face
abundances old solar 84 ; use old solar abundances for record keeping
c
c commands controlling geometry =========
sphere ; model is a full sphere covering the central star
radius 12 to 14; log of inner and outer radii (cm)
c
c other commands for details =========
iterate to convergence
c
c commands controlling output =========
print line faint -1
punch overview "nova_photos.ovr" last
punch results "nova_photos.rlt" last
punch dr "nova_photos.dr" last
punch heat "nova_photos.het" last
punch cool "nova_photos.col" last
c
c nova_photos.in
c class nova
c ========================================
c
```

this model is intensely affected by continuum pumping of atoms. The hydrogen ionizaiton is by lyman line pumping, followed by photoionization from excited states.

`optimize_phymir` *test phymir optimizers*

```
title test phymir optimizers
c spectrum computed with hden 5, temp 4
c subplex, phymir, powell
c
c commands controlling continuum =========
black 40000 K
ionization par -3
c
c commands for density & abundances =========
hden 4 vary
init file "hheonly.ini"
element oxygen on
c
c commands controlling geometry =========
Case B
stop zone 1
c
c other commands for details =========
optimize tolerance 0.03
constant temper 4.3 vary
optimize phymir
optimize iteration 100
c the following spectrum was derived by running this sim at hden 5 and t=4
c resulting eden was 5.039 and T=4, which we shall assert that we find
c >>chng 06 apr 03, backdate to old [OII] As and rederive spectrum
optimize lines
o ii 3729 0.1432
o ii 3726 0.4055
o ii 7323 0.5281
o ii 7332 0.4257
o 3 5007 3.6189
totl 4363 0.0342
end of lines
c
c commands controlling output =========
print line faint -2
c
c
c optimize_phymir.in
c class optimizer
c ========================================
c
```

This checks whether the optimizer can recover a known solution. The line spectrum was calculated at T = 1e4 K and and n_H=1e5 cm^-3, and resulted in the given electron density. The model optimize_subplex.in is a copy of this file.

`optimize_subplex` *test subplex optimizer*

```
title test subplex optimizer
c spectrum computed with hden 5, temp 4
```

```
c subplex, phymir, amoeba, powell
c
c commands controlling continuum =========
black 40000 K
ionization par -3
c
c commands for density & abundances =========
hden 4 vary
init file "hheonly.ini"
element oxygen on
c
c commands controlling geometry =========
Case B
stop zone 1
c
c other commands for details =========
constant temper 4.3 vary
optimize subplex
optimize iteration 100
optimize tolerance 0.01
c the following spectrum was derived by running this sim at hden 5 and t=4
c resulting eden was 5.039 and T=4, which we shall assert that we find
c >>chng 06 apr 03, backdate to old [OII] As and rederive spectrum
optimize lines
o ii 3729 0.1432
o ii 3726 0.4055
o ii 7323 0.5281
o ii 7332 0.4257
o 3 5007 3.6189
totl 4363 0.0342
end of lines
c
c commands controlling output =========
print line faint -2
c
c
c optimize_subplex.in
c class optimizer
c =======================================
c
```

This checks whether the optimizer can recover a known solution. The line spectrum was calculated at $T = 1e4$ K and and n_H=1e5 cm^-3, and resulted in the given electron density. The model optimize_phymir.in is a copy of this file.

`orion_hii_open` *conditions similar to Orion nebula blister*

```
title conditions similar to Orion nebula blister
c orion hii region with open geometry
c
c commands controlling continuum =========
c SED of central star
table read "star_kurucz_39600.dat"
c its flux of H-ionizing photons
phi(h) 13.0
c
c this is to include velocity field in radiative transfer
turbulence 8 km/sec no pressure
c
c commands for density & abundances =========
```

```
hden 4
abundances hii region no grains
grains orion
c
c commands controlling geometry =========
cosmic rays, background
constant pressure
c this is the account for continued matter beyond i front
double optical depths
c
c other commands for details =========
iterate 2
c
c commands controlling output =========
print line faint -5
c print lines as surface brightness
print lines surface brightness arcsec
print line optical depths
print diffuse continuum
c want to print line intensities are surface brightness, per arcsec^2
print lines surface brightness arcsec
print diffuse continuum
punch overview last "orion_hii_open.ovr"
punch dr last "orion_hii_open.dr"
punch continuum last "orion_hii_open.con" units microns
punch transmitted continuum last "orion_hii_open.trn"
punch overview last "orion_hii_open.ovr"
punch fine opacity last "orion_hii_open.opc" range 0.7 0.9 ryd
punch element nitrogen last no hash "orion_hii_open.nit"
punch line emissivity "orion_hii_open.ems" last no hash emergent
totl 5199
o 1 6300
end of lines
punch linelist "orion_hii_open.lin" "LineList_HII.dat" last no hash emergent absolute
c
c orion_hii_open.in
c class hii
c =========================================
```

This is a model similar in spirit to the blister geometry HII region model computed by Baldwin et al. (1991).
Size-resolved Orion grains are included. The constant pressure command does a hydrostatic equilibrium
structure. The predicted emission-line spectrum is affected by the reddening of the internal grains. The
resulting t2 analysis produces artificial results as a result. This has an open geometry, the original BFM paper
was a closed geometry. (This makes little difference). Background cosmic rays are also included although
these should have little effect on warm ionized gas. The emission line spectrum is given in surface brightness
units, as in the BFM paper.

`orion_hii_pdr` *constant pressure H+ region/pdr*

```
title constant pressure H+ region/pdr
c
c commands controlling continuum =========
c the incident continuum is two parts
c star and flux of photons striking it
c this is the photosphere of the OVI star, its temperature and Q(H)
c this is the result of this command
c table star kurucz 39600 K
c and makes it possible to run these orion sims without
c installing the stellar atmosphere files
```

```
table read "star_kurucz_39600.dat"
Q(H) 49
c this adds the observed hot brems, its temperature and the flux of
c photons striking the cloud
brems 6
phi(h) 10
c add cosmic rays, which are important in the pdr
cosmic rays, background
c
c commands controlling geometry =========
c this sets the separation from the star and the face of the cloud
radius 17.4507
c this turns off the stop temperature opption
c the sim will not stop due to temperature
stop temperature off
c this is typical of a gmc, larson 1981
stop total hydrogen column density 22
c this sets the thickness of the HII region & PDR
stop thickness 0.5 linear parsec
c assume constant pressure
c constant gas pressure
constant pressure
c this will result in a milli gauss B-field in molecular region
magnetic field -5 gauss
c the orion environment full covers the star, so turn on sphere
sphere
c we have a spherical geometry but want to simulate observing
c through a spectrometer's slit. use the aperture
c command for this
aperture beam
c
c other commands for details =========
c mimic existence of unmodeled molecular gas
double
c iterate since lines optically thick
iterate
c the observed microturbulence, partially a flow, so not included in pressure
turbulence 8 km/sec no pressure
c set the line width so lines appear on the punch continuum
set punchLwidth 10 km/s
c there should be no failures at all
failures 3
c
c this model goes quite deep, needs lots of zones
set nend 2200
c
c commands for density & abundances =========
c this is the log of the initial H density, cm-3
hden 4
c this will speed up the calculation a bit
init file="ism.ini"
c this uses HII region abundances, but no grains
abundances hii region no grains
c this turns on orion grains, but leaves quantum heating off to save time
grains orion no qheat
c turn on PAHs, with an abundance that depends on H0 fraction,
c as suggested by long-slit observations of Orion bar,
c with an abundance 3x larger than default built into the code
grains pah function 3 no qheat
c
c commands controlling output =========
normalize to "O 1" 63.17m
c print lots of faint CO lines
print line faint -4
punch overview last "orion_hii_pdr.ovr"
```

```
punch heating "orion_hii_pdr.het"
punch cooling "orion_hii_pdr.col"
punch dr last "orion_hii_pdr.dr"
punch results last "orion_hii_pdr.rlt"
punch continuum last "orion_hii_pdr.con" units microns
punch hydrogen 21 cm last "orion_hii_pdr.21cm"
punch hydrogen lya last "orion_hii_pdr.lya"
punch element silicon last "orion_hii_pdr.sil"
punch grain extinction last "orion_hii_pdr.grnext"
punch grain charge last "orion_hii_pdr.grnchr"
punch grain potential last "orion_hii_pdr.grnpot"
punch grain temperature last "orion_hii_pdr.grntem"
c
c
c orion_hii_pdr.in
c class hii pdr
c ========================================
```

This extends BFM from the H+ region into the PDR as in Abel et al 2005. This is the correct way to do a PDR calcualtion.

>>refer Orion model Baldwin, J., Ferland, G. J., >>refercon Martin, P. G., Corbin, M., Cota, S., Peterson, >>refercon B. M., & Slettebak, A. 1991, ApJ, 374, 580

>>refer physics HII/PDR Abel, N.P., Ferland, G.J., Shaw, G. & >>refercon van Hoof, P.A.M. 2005, ApJS, 161, 65

---

orion_hii_pdr_fast *constant gas pressure H+ region/PDR*

```
title constant gas pressure H+ region/PDR
c much faster due to physics disabled by fast.ini
c
c commands controlling continuum =========
c the incident continuum is two parts
c star and flux of photons striking it
c this is the result of this command
c table star kurucz 39600 K
c and makes it possible to run these orion sims without
c installing the stellar atmosphere files
table read "star_kurucz_39600.dat"
Q(H) 49
c add cosmic rays, which are important at depth
cosmic rays, background
c plus hot brems
brems 6
phi(h) 10
c
c commands for density & abundances =========
hden 4
init file="fast.ini"
abundances hii region no grains
c single sized grain with no quantum heating
grains orion no qheat single
c
c commands controlling geometry =========
radius 17.4507
sphere
c let it go into the molecular cloud
stop temperature 10 linear
```

```
c this is roughly the thickness of OMC1
stop thickness 1 linear parsec
c mimic existence of unmodeled molecular gas with double command
double
c
c other commands for details =========
failures 3
iterate
c make constant gas pressure since fast turned off line radiation pressure
constant gas pressure
c the observed microturbulence
turbulence 8 km/sec
c we have a spherical geometry but want to simulate observing
c through a spectrometer's slit. use the aperture
c command for this
aperture beam
c
c commands controlling output =========
normalize to "CaBo" 4861
print line faint -4
punch overview last "orion_hii_pdr_fast.ovr"
punch heating "orion_hii_pdr_fast.het"
punch cooling "orion_hii_pdr_fast.col"
punch dr last "orion_hii_pdr_fast.dr"
punch results last "orion_hii_pdr_fast.rlt"
punch continuum last "orion_hii_pdr_fast.con" units microns
punch ionizing continuum last "orion_hii_pdr_fast.ion"
punch hydrogen 21 cm last "orion_hii_pdr_fast.21cm"
punch hydrogen lya last "orion_hii_pdr_fast.lya"
punch element silicon last "orion_hii_pdr_fast.sil"
punch element sulphur last "orion_hii_pdr_fast.sul"
c
c
c orion_hii_pdr_fast.in
c class hii pdr
c ==========================================
c
```

Orion HII region and PDR, simialr to orion_hii_pdr but much faster because of fast.ini

---

orion_hii_pdr_pp *the Orion HII Region / PDR / Molecular cloud with an open geometry*

```
title the Orion HII Region / PDR / Molecular cloud with an open geometry
c
c commands controlling continuum =========
c the incident continuum has two parts
c kurucz continuum with flux of photons striking cloud
c this is the the OVI star, its temperature and phi(H)
c this is the result of this command
c table star kurucz 39600 K
c and makes it possible to run these orion sims without
c installing the stellar atmosphere files
table read "star_kurucz_39600.dat"
phi(H) 13
c
c this adds the observed hot brems
c its temperature (as log of T) and the flux of
c photons striking the cloud
brems 6
```

```
phi(h) 10
c
c cosmic rays are important for pdr chemistry
cosmic rays, background
c
c commands controlling geometry =========
c this turns off the stop temperature option
c so the sim will not stop due to temperature
stop temperature off
c this sets the thickness of the HII region & PDR
stop thickness 0.5 linear parsec
c this is typical of a gmc, larson 1981
stop total hydrogen column density 22
c this will result in a milli gauss B-field in molecular region
magnetic field -5 gauss
c assume constant pressure
constant pressure
set nend 2000
c
c other commands for details =========
failures 3
c mimic existence of unmodeled molecular gas
double
c iterate since lines optically thick
iterate
c set microturbulence in equipartition with B field
turbulence equipartition
c set the line width so lines appear on the punch continuum
set punchLwidth 10 km/s
c
c commands for density & abundances =========
c this is the log of the initial H density, cm-3
hden 4
c this will speed up the calculation a bit
init file="ism.ini"
c this uses HII region abundances, but no grains
abundances hii region no grains
c this uses orion grains
grains orion
c turn on PAHs, with an abundance that depends on H0 fraction,
c as suggested by long-slit observations of Orion bar,
c with an abundance 3x larger than default built into the code
grains pah function 3
c
c commands controlling output =========
c print lots of faint CO lines
print line faint -6
c normalize to Ha
normalize to "H 1" 6563
punch overview last "orion_hii_pdr_pp.ovr"
punch temperature last "orion_hii_pdr_pp.tem"
punch overview "orion_hii_pdr_pp.ovr1"
punch molecules last "orion_hii_pdr_pp.mol"
punch molecules "orion_hii_pdr_pp.mol1"
punch heating "orion_hii_pdr_pp.het"
punch cooling "orion_hii_pdr_pp.col"
punch dr last "orion_hii_pdr_pp.dr"
punch results last "orion_hii_pdr_pp.rlt"
punch continuum last "orion_hii_pdr_pp.con" units microns
punch hydrogen 21 cm last "orion_hii_pdr_pp.21cm"
punch hydrogen lya last "orion_hii_pdr_pp.lya"
punch element silicon last "orion_hii_pdr_pp.sil"
punch pressure last "orion_hii_pdr_pp.pre"
punch pressure "orion_hii_pdr_pp.pre1"
punch grain extinction last "orion_hii_pdr_pp.grnext"
```

```
punch grain charge last "orion_hii_pdr_pp.grnchr"
punch grain potential last "orion_hii_pdr_pp.grnpot"
punch grain temperature last "orion_hii_pdr_pp.grntem"
punch grain temperature "orion_hii_pdr_pp.grntem1"
c
c
c orion_hii_pdr_pp.in
c class hii pdr
c =======================================
```

similar to orion_hii_pdr except for plane parallel geometry

`pdr_co_fully` *H2 and CO in fully molecular limit*

```
title H2 and CO in fully molecular limit
c this is a pair with pdr_co_fully_noneq - that tests non equilibrium chem
c
c commands controlling continuum =========
blackbody 5000
luminosity total solar linear 2
brems 6
luminosity total solar log -2.7
c
c commands for density & abundances =========
c set the density and composition, ism with grains
hden 10
abundances ism no qheat
c
c other commands for details =========
c this is an important ionization source
cosmic ray background
c code will stop at 4000 K by default, lower this temp so we
c can do fully molecular gas
stop temperature 10K linear
c
c commands controlling geometry =========
c the radius and thickness
radius 15.8
stop thickness 6
turbulence 3 km / s
c
c commands controlling output =========
punch overview "pdr_co_fully.ovr"
punch molecules "pdr_co_fully.mol"
punch dr "pdr_co_fully.dr"
punch heating "pdr_co_fully.het"
punch cooling "pdr_co_fully.col"
punch grain charge "pdr_co_fully.grnchr"
punch grain drift "pdr_co_fully.grndft"
punch grain temperature "pdr_co_fully.grntem"
c
c
c gas has more he++ than He+ due to fast charge transfer with co
c neutralizing he+
c pdr_co_fully.in
c class pdr
c =======================================
```

test code in fully molecular limit this is a pair with pdr_co_fully_noneq - that tests non equilibrium chem

---

`pdr_co_fully_noneq` *H2 and CO are fully molecular, non-equilibrium case*

```
title H2 and CO are fully molecular, non-equilibrium case
c this is a pair with pdr_co_fully.in, which does not include non-equil chem
c
c commands controlling continuum =========
blackbody 5000
luminosity total solar linear 2
brems 6
luminosity total solar log -2.7
c
c commands for density & abundances =========
c set the density and composition, ism with grains
hden 10
abundances ism no qheat
c
c other commands for details =========
c this is an important ionization source
cosmic ray background
c code will stop at 4000 K by default, lower this temp so we
c can do fully molecular gas
stop temperature 10K linear
c
c commands controlling geometry =========
c the radius and thickness
radius 15.8
stop thickness 6
turbulence 3 km / s
c this turns on non-equilibrium chemistry as suggested by Federman et al. 1996
set chemistry non equilibrium
c
c commands controlling output =========
punch overview "pdr_co_fully_noneq.ovr"
punch molecules "pdr_co_fully_noneq.mol"
punch dr "pdr_co_fully_noneq.dr"
punch heating "pdr_co_fully_noneq.het"
punch cooling "pdr_co_fully_noneq.col"
punch grain charge "pdr_co_fully_noneq.grnchr"
punch grain drift "pdr_co_fully_noneq.grndft"
punch grain temperature "pdr_co_fully_noneq.grntem"
c
c
c gas has more he++ than He+ due to fast charge transfer with co
c neutralizing he+
c pdr_co_fully_noneq.in
c class pdr
c ========================================
```

test code in fully molecular limit with Federman non-equilibrium chem this is a pair with pdr_co_fully.in, which does not include non-equil chem

`pdr_coolbb` *illumination by cool STE blackbody*

```
title illumination by cool STE blackbody
c gasis fully molecular, grains should be in STE
c
```

```
c commands controlling continuum =========
c illuminate with 120 blackbody in thermodynamic equilibrium
blackbody 120 STE
CMB
c
c commands for density & abundances =========
c density
hden 7
c ism dust and grains
abundances ISM no grains
c dust to gas ratio is 10x standard
grains 1.0
c lowers metals and grains to 1/10 standard, so metals are
c 1/10 solar while dust to gas ratio is galactic
metals and grains 0.1
c
c other commands for details =========
c set background cosmic rays to allow chemistry
cosmic ray background
iterate
c
c commands controlling geometry =========
radius 17.5
sphere
stop temperature 20K linear
stop column density 25.5
c
c commands controlling output =========
punch overview "pdr_coolbb.ovr"
punch molecules "pdr_coolbb.mol"
punch heating "pdr_coolbb.het"
punch cooling "pdr_coolbb.col"
punch grain temperature "pdr_coolbb.grntem"
c
c
c pdr_coolbb.in
c class pdr
c ========================================
```

This is a dense ISM exposed to a cool blackbody. This tests the ability to converge onto a VERY molecular environment. Nearly all C and H are in CO and H2. Cosmic rays are the main source of heat and ionization.

`pdr_dense_persei` *dense phase zeta persei cloud*

```
title dense phase zeta persei cloud
c
c density and abundances ==========
c density of dense phase from Table 2 of Le Petit paper ========
hden 4.3
c
c abundandces from Table 1 of Le Petit paper ==========
element carbon abundance 0.000132 linear
element helium abundance 0.10 linear
element oxygen abundance 0.00032 linear
element nitrogen abundance 0.000075 linear
element sulphur abundance 0.0000186 linear
element silicon abundance 0.000029 linear
c set abundance of all other elements to zero ===================
element copper off
element magnesium off
```

```
element manganese off
element sodium off
element chlorine off
element vanadium off
element potassium off
element phosphorous off
element calcium off
element iron off
element zinc off
element neon off
element argon off
element fluorine off
element aluminum off
element boron off
element lithium off
element beryllium off
element scandium off
element nickel off
element titanium off
element chromium off
element cobalt off
c do not use Federman rates for this model =============
set federman chemistry off
c use standard ism grain size distribution ============
grains ism
c
c command controlling the continuum, for this model is Draine 1978 field ====
table draine 0.5 linear
c make sure no H-ioinizing radiation strikes the cloud
extinguish 24
c
c fix the temperature to 20 K =============
constant temperature 20
c stop at a radius of 4.3e-4 parsecs ============
stop thickness 0.00043 parsecs linear
c Le Petit model does not consider ices, so turn this off ============
no grain molecules
c turn on cosmic rays =========
cosmic rays background
c Set cosmic ray ionization rate to Table 2 of Le Petit paper ========
set csupra -15.6
c Allow calculation to go extend into cold environment ============
stop temperature linear 3
c
c commands controlling output ============
punch overview "pdr_dense_persei.ovr"
punch dr "pdr_dense_persei.dr"
punch molecules "pdr_dense_persei.mol"
punch heating "pdr_dense_persei.het"
punch cooling "pdr_dense_persei.col"
c
c pdr_dense_persei.in
c class pdr
c ====================================
```

This is the dense phase model presented by LePetit, Roueff, and Herbst in order to reproduce C2 and C3 column densities observed along the line of sight to zera persei. This is our attempt at reproducing their calculation. This is the dense molecular phase, not the phase that produces H3+

// >>refer model pdr Le Petit, F., Roueff, E., & Herbst, E. 2004, // >>refercon A&A, 417, 993

If you do a thermal equilibrium calculation by removing the constant temperature command the kinetic temperature will be about three times larger than assumed in their paper.

pdr_HTT91 *low-density PDR from HTT91*

```
title low-density PDR from HTT91
c >>refer test model Hollenbach, D., Takahashi, T., &
c >>refcon Tielens, A.G.G.M., 1991, ApJ,377, 192-209
c
c commands controlling continuum =========
c first continuum is FIR hot grain continuum produced in
c unmodeled HII Region
c this is hot star continuum
black 30000
intensity 0.2 range 0.4412 to 1 Ryd
c this will remove all ionizing radiation
extinguish 24 0
cosmic rays, background
c
c commands for density & abundances =========
hden 3
init file="ism.ini"
abundances he -1.01 c -3.52 n-8 o-3.30 ne-8 mg-5.89
continue si -6.10 s -5.10 cl-7 ar-8 fe -6.60
grains orion, abundance log 0.16 no qheat
c
c commands controlling geometry =========
sphere
c stop when gas is fully neutral
stop efrac -10
c this is to stop at an intended place, since results would be
c very dependent on details if we stop on temperature
stop thickness 18.954
c stop when gas is cold
stop temperature 10 linear
c add this to mimic unmodelled neutral gas
double optical depths
c
c other commands for details =========
iterate 2
case b
turbulence 1.5 km/sec
c
c commands controlling output =========
normalize to 157.6m "C 2"
c uv lines are strongly pumped by stellar continuum, break out contribution
print line pump
print line optical depths
print ages
punch overview last "pdr_HTT91.ovr"
punch dr "pdr_HTT91.dr"
punch molecules last "pdr_HTT91.mol"
punch results last "pdr_HTT91.rlt"
punch continuum units microns last "pdr_HTT91.con"
punch heating last "pdr_HTT91.het"
punch cooling last "pdr_HTT91.col"
c
c pdr_HTT91.in
c class pdr
c =======================================
c
```

This is the Hollenbach et al 1991 Low-density PDR The case b command appears because the Lyman lines are vastly optically thick in this environment. If the command is removed the Lyman lines will be optically

thin, which will result in fast fluorescent excitation of the (nearly totally neutral) hydrogen atom.

there is very little CO in this model since it is not thick enough for the UV pumping lines to become optically thick

---

`pdr_leiden_f1` *Leiden meeting model 1*

```
title Leiden meeting model 1
c This is one of the tests in Rollig et al. 2007, A&A, 467, 187
c
c commands controlling continuum =========
c Use the Draine 1978 field, for a semi-infinite slab we have to use half
c the requested value, so the actual value
c they want for the model is actually twice the value below
table draine 5
extinguish 24
c
c commands for density & abundances =========
c hydrogen density
hden 3.
grains ism 1.16 no qheat
c
c commands controlling geometry =========
c
c other commands for details =========
failures 3
c use leiden initialization file
init file="pdr_leiden.ini"
c This command defines the grain temperature to be a constant 20 Kelvin
constant grain temperature 20
c This sets the temperature to a constant 50 Kelvin
constant temperature 50 linear
c
c commands controlling output =========
print line optical depths
punch overview "pdr_leiden_f1.ovr"
punch continuum "pdr_leiden_f1.con"
punch leiden lines "pdr_leiden_f1.lin"
punch leiden "pdr_leiden_f1.lei"
punch dr "pdr_leiden_f1.dr"
punch molecules "pdr_leiden_f1.mol"
punch grain physical conditions "pdr_leiden_f1.grn"
punch chemistry rates CO "pdr_leiden_f1.rat"
punch H2 destruction "pdr_leiden_f1.H2d"
punch H2 creation "pdr_leiden_f1.H2c"
punch secondaries "pdr_leiden_f1.sec"
c
c
c pdr_leiden_f1.in
c class pdr
c =======================================
c
```

This sim has some interesting properties. The grain temp is so low that O freezes onto grains as H2O. This is so efficient that CO does not become well formed - the CO/C ratio only reaches 0.3. This is one of the tests in Rollig et al. 2007, A&A, 467, 187

`pdr_leiden_f2` *Leiden meeting model 2*

```
title Leiden meeting model 2
c This is one of the tests in Rollig et al. 2007, A&A, 467, 187
c
c commands controlling continuum =========
c Use the Draine 1978 field, for a semi-infinite slab we have to use
c half the requested value, so the actual value
c they want for the model is actually twice the value below
table draine 50000
extinguish 24
c
c commands for density & abundances =========
grains ism 1.16 no qheat
c hydrogen density
hden 3.
c
c commands controlling geometry =========
c
c other commands for details =========
failures 3
c use leiden initialization file
init file="pdr_leiden.ini"
c This command defines the grain temperature to be a constant 20 Kelvin
constant grain temperature 20
c This sets the temperature to a constant 50 Kelvin
constant temperature 50 linear
c
c commands controlling output =========
punch overview "pdr_leiden_f2.ovr"
punch continuum "pdr_leiden_f2.con"
punch molecules "pdr_leiden_f2.mol"
punch leiden lines "pdr_leiden_f2.lin"
punch leiden "pdr_leiden_f2.lei"
punch dr "pdr_leiden_f2.dr"
punch grain physical conditions "pdr_leiden_f2.grn"
c
c
c pdr_leiden_f2.in
c class pdr
c ========================================
c
```

c This is one of the tests in Rollig et al. 2007, A&A, 467, 187

`pdr_leiden_f3` *Leiden meeting model 3*

```
title Leiden meeting model 3
c This is one of the tests in Rollig et al. 2007, A&A, 467, 187
c
c commands controlling continuum =========
c Use the Draine 1978 field, for a semi-infinite slab we have to use half
c the requested value, so the actual value
c they want for the model is actually twice the value below
table draine 5
extinguish 24
c
c commands for density & abundances =========
```

```
grains ism 1.16 no qheat
c hydrogen density
hden 5.5
c
c commands controlling geometry =========
c
c other commands for details =========
failures 3
c use leiden initialization file
init file="pdr_leiden.ini"
c This command defines the grain temperature to be a constant 20 Kelvin
constant grain temperature 20
c This sets the temperature to a constant 50 Kelvin
constant temperature 50 linear
c
c commands controlling output =========
punch overview "pdr_leiden_f3.ovr"
punch continuum "pdr_leiden_f3.con"
punch leiden lines "pdr_leiden_f3.lin"
punch leiden "pdr_leiden_f3.lei"
punch dr "pdr_leiden_f3.dr"
punch grain physical conditions "pdr_leiden_f3.grn"
c
c
c pdr_leiden_f3.in
c class pdr
c =========================================
c
```

c This is one of the tests in Rollig et al. 2007, A&A, 467, 187

`pdr_leiden_f4` *Leiden meeting model 4*

```
title Leiden meeting model 4
c This is one of the tests in Rollig et al. 2007, A&A, 467, 187
c
c commands controlling continuum =========
c
c commands for density & abundances =========
grains ism 1.16 no qheat
c
c commands controlling geometry =========
c Use the Draine 1978 field, for a semi-infinite slab we have to use half
c the requested value, so the actual value
c they want for the model is actually twice the value below
table draine 50000
extinguish 24
c
c other commands for details =========
failures 3
c use leiden initialization file
init file="pdr_leiden.ini"
c This command defines the grain temperature to be a constant 20 Kelvin
constant grain temperature 20
c This sets the temperature to a constant 50 Kelvin
constant temperature 50 linear
c hydrogen density
hden 5.5
c
c commands controlling output =========
punch overview "pdr_leiden_f4.ovr"
```

```
punch continuum "pdr_leiden_f4.con"
punch leiden lines "pdr_leiden_f4.lin"
punch leiden "pdr_leiden_f4.lei"
punch dr "pdr_leiden_f4.dr"
punch grain physical conditions "pdr_leiden_f4.grn"
c
c
c pdr_leiden_f4.in
c class pdr
c ========================================
c
```

c This is one of the tests in Rollig et al. 2007, A&A, 467, 187

`pdr_leiden_hack_f1` *Leiden meeting model 1 with hacks*

```
title Leiden meeting model 1 with hacks
c This is one of the tests in Rollig et al. 2007, A&A, 467, 187
c
c commands controlling continuum =========
c Use the Draine 1978 field, for a semi-infinite slab we have to use half
c the requested value, so the actual value
c they want for the model is actually twice the value below
table draine 5
extinguish 24
c
c commands for density & abundances =========
c hydrogen density
hden 3.
grains ism 1.16 no qheat
c
c commands controlling geometry =========
c
c other commands for details =========
failures 3
c use leiden initialization file
init file="pdr_leiden_hack.ini"
c This command defines the grain temperature to be a constant 20 Kelvin
constant grain temperature 20
c This sets the temperature to a constant 50 Kelvin
constant temperature 50 linear
c
c commands controlling output =========
punch leiden lines "pdr_leiden_hack_f1.lin"
punch leiden "pdr_leiden_hack_f1.lei"
punch dr "pdr_leiden_hack_f1.dr"
punch grain physical conditions "pdr_leiden_hack_f1.grn"
punch overview "pdr_leiden_hack_f1.ovr"
c
c
c
c
c pdr_leiden_f1_hack.in
c class pdr
c ========================================
c
```

c This is one of the tests in Rollig et al. 2007, A&A, 467, 187 this includes hacks to get "standard answer"

`pdr_leiden_hack_f2` *Leiden meeting model 2 with hacks*

```
title Leiden meeting model 2 with hacks
c This is one of the tests in Rollig et al. 2007, A&A, 467, 187
c
c commands controlling continuum =========
c Use the Draine 1978 field, for a semi-infinite slab we have to use
c half the requested value, so the actual value
c they want for the model is actually twice the value below
table draine 50000
extinguish 24
c
c commands for density & abundances =========
grains ism 1.16 no qheat
c hydrogen density
hden 3.
c
c commands controlling geometry =========
c
c other commands for details =========
failures 3
c use leiden initialization file
init file="pdr_leiden_hack.ini"
c This command defines the grain temperature to be a constant 20 Kelvin
constant grain temperature 20
c This sets the temperature to a constant 50 Kelvin
constant temperature 50 linear
c
c commands controlling output =========
punch leiden lines "pdr_leiden_hack_f2.lin"
punch leiden "pdr_leiden_hack_f2.lei"
punch dr "pdr_leiden_hack_f2.dr"
punch grain physical conditions "pdr_leiden_hack_f2.grn"
punch overview "pdr_leiden_hack_f2.ovr"
c
c
c pdr_leiden_hack_f2.in
c class pdr
c ========================================
c
```

c This is one of the tests in Rollig et al. 2007, A&A, 467, 187 includes hacks to get standard answer.

`pdr_leiden_hack_f3` *Leiden meeting model 3 with hacks*

```
title Leiden meeting model 3 with hacks
c This is one of the tests in Rollig et al. 2007, A&A, 467, 187
c
c commands controlling continuum =========
c Use the Draine 1978 field, for a semi-infinite slab we have to use half
c the requested value, so the actual value
c they want for the model is actually twice the value below
table draine 5
extinguish 24
c
c commands for density & abundances =========
grains ism 1.16 no qheat
c hydrogen density
```

```
hden 5.5
c
c commands controlling geometry =========
c
c other commands for details =========
failures 3
c use leiden initialization file
init file="pdr_leiden_hack.ini"
c This command defines the grain temperature to be a constant 20 Kelvin
constant grain temperature 20
c This sets the temperature to a constant 50 Kelvin
constant temperature 50 linear
c
c commands controlling output =========
punch leiden lines "pdr_leiden_hack_f3.lin"
punch leiden overview "pdr_leiden_hack_f3.ovr"
punch leiden "pdr_leiden_hack_f3.lei"
punch dr "pdr_leiden_hack_f3.dr"
punch grain physical conditions "pdr_leiden_hack_f3.grn"
c
c
c pdr_leiden_hack_f3.in
c class pdr
c ========================================
c
```

c This is one of the tests in Rollig et al. 2007, A&A, 467, 187 include hacks to get standard answer.

`pdr_leiden_hack_f4` *Leiden meeting model 4 with hacks*

```
title Leiden meeting model 4 with hacks
c This is one of the tests in Rollig et al. 2007, A&A, 467, 187
c
c commands controlling continuum =========
c
c commands for density & abundances =========
grains ism 1.16 no qheat
c
c commands controlling geometry =========
c Use the Draine 1978 field, for a semi-infinite slab we have to use half
c the requested value, so the actual value
c they want for the model is actually twice the value below
table draine 50000
extinguish 24
c
c other commands for details =========
failures 3
c use leiden initialization file
init file="pdr_leiden_hack.ini"
c This command defines the grain temperature to be a constant 20 Kelvin
constant grain temperature 20
c This sets the temperature to a constant 50 Kelvin
constant temperature 50 linear
c hydrogen density
hden 5.5
c
c commands controlling output =========
punch leiden lines "pdr_leiden_hack_f4.lin"
punch leiden "pdr_leiden_hack_f4.lei"
punch dr "pdr_leiden_hack_f4.dr"
punch grain physical conditions "pdr_leiden_hack_f4.grn"
```

```
punch overview "pdr_leiden_hack_f4.ovr"
c
c
c pdr_leiden_hack_f4.in
c class pdr
c =======================================
c
```

This is one of the tests in Rollig et al. 2007, A&A, 467, 187 includes hacks needed to get standard answer.

`pdr_leiden_hack_v1` *Leiden meeting model 5 with hacks*

```
title Leiden meeting model 5 with hacks
c This is one of the tests in Rollig et al. 2007, A&A, 467, 187
c
c commands controlling continuum =========
c Use the Draine 1978 field, for a semi-infinite slab we have to use half
c the requested value, so the actual value
c is half the requested value
c they want for the model is actually twice the value below
table draine 5
extinguish 24
c
c commands for density & abundances =========
c hydrogen density
hden 3.
grains ism 1.16 no qheat
c
c commands controlling geometry =========
c
c other commands for details =========
failures 3
c use leiden initialization file
init file="pdr_leiden_hack.ini"
* element nitrogen on
* element nitrogen scale -1
* element silicon on
* element silicon scale -1
* element sulphur on
* element sulphur scale -1
* element chlorine on
* element chlorine scale -1
c Define the abundance of PAHs so we can reproduce the same photoelectric
c heating curve as BT94
c We have also scaled the grain abundance to achieve the desired AV/NH ratio
* grains PAH no qheat 8
c
c commands controlling output =========
punch leiden lines "pdr_leiden_hack_v1.lin"
punch leiden "pdr_leiden_hack_v1.lei"
punch dr "pdr_leiden_hack_v1.dr"
punch grain temperature "pdr_leiden_hack_v1.grn"
punch overview "pdr_leiden_hack_v1.ovr"
c
c
c pdr_leiden_hack_v1.in
c class pdr
c =======================================
c
```

This is one of the tests in Rollig et al. 2007, A&A, 467, 187 includes hacks to get standard answer.

---

pdr_leiden_hack_v2 *Leiden meeting model 6 with hacks*

```
title Leiden meeting model 6 with hacks
c This is one of the tests in Rollig et al. 2007, A&A, 467, 187
c
c commands controlling continuum =========
c Use the Draine 1978 field, for a semi-infinite slab we have to use half
c the requested value, so the actual value
c they want for the model is actually twice the value below
table draine 50000
extinguish 24
c
c commands for density & abundances =========
c Define the abundance of PAHs so we can reproduce the same photoelectric
c heating curve as BT94
c We have also scaled the grain abundance to achieve the desired AV/NH ratio
* grains PAH no qheat 40
grains ism 1.16 no qheat
c hydrogen density
hden 3.
c
c commands controlling geometry =========
c
c other commands for details =========
failures 3
c use leiden initialization file
init file="pdr_leiden_hack.ini"
c
c commands controlling output =========
punch leiden lines "pdr_leiden_hack_v2.lin"
punch leiden "pdr_leiden_hack_v2.lei"
punch dr "pdr_leiden_hack_v2.dr"
punch grain temperature "pdr_leiden_hack_v2.grn"
punch overview "pdr_leiden_hack_v2.ovr"
punch heating "pdr_leiden_hack_v2.het"
punch cooling "pdr_leiden_hack_v2.col"
punch h2 rates "pdr_leiden_hack_v2.h2rat"
c
c
c pdr_leiden_hack_v2.in
c class pdr
c =========================================
c
```

This is one of the tests in Rollig et al. 2007, A&A, 467, 187 includes hacks to get standard answer.

---

pdr_leiden_hack_v3 *Leiden meeting model 7 with hacks*

```
title Leiden meeting model 7 with hacks
c This is one of the tests in Rollig et al. 2007, A&A, 467, 187
c
c commands controlling continuum =========
c Use the Draine 1978 field, for a semi-infinite slab we have to use half
c the requested value, so the actual value
c they want for the model is actually twice the value below
```

```
       table draine 5
       extinguish 24
       c
       c commands for density & abundances =========
       c Define the abundance of PAHs so we can reproduce the same photoelectric
       c heating curve as BT94
       c We have also scaled the grain abundance to achieve the desired AV/NH ratio
       * grains PAH no qheat 5.7
       grains ism 1.16 no qheat
       c hydrogen density
       hden 5.5
       c
       c commands controlling geometry =========
       c
       c other commands for details =========
       c this sim has more than one thermal solution - force into lowest one,
       c the one found by the rest of the group - without this we will find a
       c soln at 1400 K
       force temperature to 60K
       failures 3
       c use leiden initialization file
       init file="pdr_leiden_hack.ini"
       c
       c commands controlling output =========
       punch leiden lines "pdr_leiden_hack_v3.lin"
       punch leiden "pdr_leiden_hack_v3.lei"
       punch dr "pdr_leiden_hack_v3.dr"
       punch grain temperature "pdr_leiden_hack_v3.grn"
       punch overview "pdr_leiden_hack_v3.ovr"
       punch heating "pdr_leiden_hack_v3.het"
       punch cooling "pdr_leiden_hack_v3.col"
       c
       c
       c pdr_leiden_hack_v3.in
       c class pdr
       c =========================================
       c
```

This is one of the tests in Rollig et al. 2007, A&A, 467, 187 includes hacks to get standard answer

---

pdr_leiden_hack_v4 *Leiden meeting model 8 with hacks*

```
       title Leiden meeting model 8 with hacks
       c This is one of the tests in Rollig et al. 2007, A&A, 467, 187
       c
       c commands controlling continuum =========
       c Use the Draine 1978 field, for a semi-infinite slab we have to use half
       c the requested value, so the actual value
       c they want for the model is actually twice the value below
       table draine 50000
       extinguish 24
       c
       c commands for density & abundances =========
       c Define the abundance of PAHs so we can reproduce the same photoelectric
       c heating curve as BT94
       c We have also scaled the grain abundance to achieve the desired AV/NH ratio
       * grains PAH no qheat 40
       * grains ism 0.936 no qheat
       grains ism 1.16 no qheat
       set PAH Bakes
       c hydrogen density
```

```
hden 5.5
c
c commands controlling geometry =========
c
c other commands for details =========
* failures 3
c use leiden initialization file
init file="pdr_leiden_hack.ini"
c
c commands controlling output =========
punch leiden lines "pdr_leiden_hack_v4.lin"
punch leiden "pdr_leiden_hack_v4.lei"
punch dr "pdr_leiden_hack_v4.dr"
punch grain temperature "pdr_leiden_hack_v4.grn"
punch overview "pdr_leiden_hack_v4.ovr"
punch heating "pdr_leiden_hack_v4.het"
punch cooling "pdr_leiden_hack_v4.col"
punch transmitted continuum "pdr_leiden_hack_v4.con"
c
c
c pdr_leiden_hack_v4.in
c class pdr
c =========================================
c
```

This is one of the tests in Rollig et al. 2007, A&A, 467, 187 includes hacks to get standard answer.

`pdr_leiden_v1` *Leiden meeting model 5*

```
title Leiden meeting model 5
c This is one of the tests in Rollig et al. 2007, A&A, 467, 187
c
c commands controlling continuum =========
c Use the Draine 1978 field, for a semi-infinite slab we have to use half
c the requested value, so the actual value
c is half the requested value
c they want for the model is actually twice the value below
table draine 5
extinguish 24
c
c commands for density & abundances =========
c hydrogen density
hden 3.
grains ism 1.16 no qheat
c add PAHs
grains PAH no qheat 3 function
c
c commands controlling geometry =========
c
c other commands for details =========
failures 3
c use leiden initialization file
init file="pdr_leiden.ini"
c
c commands controlling output =========
punch overview "pdr_leiden_v1.ovr"
punch leiden lines "pdr_leiden_v1.lin"
punch leiden "pdr_leiden_v1.lei"
punch dr "pdr_leiden_v1.dr"
punch grain temperature "pdr_leiden_v1.grn"
c
```

```
c
c
c pdr_leiden_v1.in
c class pdr
c =========================================
c
```

This is one of the tests in Rollig et al. 2007, A&A, 467, 187

`pdr_leiden_v2` *Leiden meeting model 6*

```
title Leiden meeting model 6
c This is one of the tests in Rollig et al. 2007, A&A, 467, 187
c
c commands controlling continuum =========
c Use the Draine 1978 field, for a semi-infinite slab we have to use half
c the requested value, so the actual value
c they want for the model is actually twice the value below
table draine 50000
extinguish 24
c
c commands for density & abundances =========
c add PAHs and grains
grains PAH no qheat 3 function
grains ism 1.16 no qheat
c hydrogen density
hden 3.
c
c commands controlling geometry =========
c
c other commands for details =========
failures 3
c use leiden initialization file
init file="pdr_leiden.ini"
c
c commands controlling output =========
punch overview "pdr_leiden_v2.ovr"
punch leiden lines "pdr_leiden_v2.lin"
punch leiden "pdr_leiden_v2.lei"
punch dr "pdr_leiden_v2.dr"
punch grain temperature "pdr_leiden_v2.grn"
c
c
c pdr_leiden_v2.in
c class pdr
c =========================================
c
```

c This is one of the tests in Rollig et al. 2007, A&A, 467, 187

`pdr_leiden_v3` *Leiden meeting model 7*

```
title Leiden meeting model 7
c This is one of the tests in Rollig et al. 2007, A&A, 467, 187
c
c commands controlling continuum =========
```

```
c Use the Draine 1978 field, for a semi-infinite slab we have to use half
c the requested value, so the actual value
c they want for the model is actually twice the value below
table draine 5
extinguish 24
c
c commands for density & abundances =========
c add PAHs and grains
grains PAH no qheat 3 function
grains ism 1.16 no qheat
c hydrogen density
hden 5.5
c
c commands controlling geometry =========
c
c other commands for details =========
failures 3
c use leiden initialization file
init file="pdr_leiden.ini"
c
c commands controlling output =========
punch overview "pdr_leiden_v3.ovr"
punch leiden lines "pdr_leiden_v3.lin"
punch leiden "pdr_leiden_v3.lei"
punch dr "pdr_leiden_v3.dr"
punch grain temperature "pdr_leiden_v3.grn"
punch heating "pdr_leiden_v3.het"
punch cooling "pdr_leiden_v3.col"
c
c
c pdr_leiden_v3.in
c class pdr
c =========================================
c
```

---

pdr_leiden_v4   *Leiden meeting model 8*

```
title Leiden meeting model 8
c This is one of the tests in Rollig et al. 2007, A&A, 467, 187
c
c commands controlling continuum =========
c Use the Draine 1978 field, for a semi-infinite slab we have to use half
c the requested value, so the actual value
c they want for the model is actually twice the value below
table draine 50000
extinguish 24
c
c commands for density & abundances =========
grains PAH no qheat 3 function
grains ism 1.16 no qheat
c hydrogen density
hden 5.5
c
c commands controlling geometry =========
c
c other commands for details =========
failures 3
c use leiden initialization file
init file="pdr_leiden.ini"
c
c commands controlling output =========
```

```
punch overview "pdr_leiden_v4.ovr"
punch leiden lines "pdr_leiden_v4.lin"
punch leiden "pdr_leiden_v4.lei"
punch dr "pdr_leiden_v4.dr"
punch grain temperature "pdr_leiden_v4.grn"
punch transmitted continuum "pdr_leiden_v4.con"
c
c
c pdr_leiden_v4.in
c class pdr
c =========================================
c
```

pdr_orion_veil *Orion's veil*

```
title Orion's veil
c
c commands controlling continuum =========
q(h) 50.000000
table star rubin
c AGN & CMB background
background z=0
cosmic rays background
c as a test this was tried - atomic temperature raised fm 100 to 140
c but nothing really wrong
* cosmic rays equipartition
c the observed magnetic field, roughly 100 micro Gauss
magnetic field -4
c
c commands for density & abundances =========
hden 3.1
init "ism.ini"
abundances orion no grains
grains orion no qheat
c
c commands controlling geometry =========
stop neutral hydrogen column density 21.6
radius 19.1
stop temperature linear 5
c this will not be hit, but put it in to insure that we stop if
c gas becomes totally molecular
stop av 10
c
c other commands for details =========
failures 3
* uncomment to include large H2 molecule - results will CHANGE!
* atom h2
turbulence 0.5 km/sec
sphere
c c this simulates Lyman lines in the stellar continuum
c atom h-like lyman pumping off
c
c commands controlling output =========
print line optical depths
punch PDR "pdr_orion_veil.pdr"
punch feii column densities "pdr_orion_veil.fe2col"
punch hydrogen 21 cm "pdr_orion_veil.21cm"
punch molecules "pdr_orion_veil.mol"
punch overveiw "pdr_orion_veil.ovr"
punch column densities "pdr_orion_veil.col"
punch heating "pdr_orion_veil.het"
```

```
punch cooling "pdr_orion_veil.col"
punch dr "pdr_orion_veil.dr"
punch H2 rates "pdr_orion_veil.h2rat"
punch pressure "pdr_orion_veil.pre"
punch wind "pdr_orion_veil.wnd"
print line optical depths
punch continuum units microns "pdr_orion_veil.con" last
c
c pdr_orion_veil.in
c class pdr
c ========================================
c
```

This is Abel et al 2004 model of Orion's veil, the layer of gas in front of the Orion Nebula. >>refer Orion Veil
Abel, N. P., Brogan, C. L., Ferland, G. J., >>refercon O'Dell, C. R., Shaw, G., & Troland, T. H. 2004

---

pdr_th85ism *TH85 PDR with ISM grains*

```
title TH85 PDR with ISM grains
c
c commands controlling continuum =========
c cosmic radiation background
background z=0
c cosmic ray background ionization and heating
cosmic rays, background
c first continuum is FIR hot grain continuum produced in
c unmodeled HII Region
blackbody, t = 75 K
intensity 2.7 (total)
c this is the attenuated hot stellar continuum
black 30000
intensity 2.204 range 0.4412 to 1 Ryd
c this will remove all ionizing radiation
extinguish 24 0
c
c commands controlling geometry =========
c simulate effects of gas we do not model
double
c their turbulence
turbulence 2.7 km/sec
c stopping criteria
c this should be the one actually used
stop AV 33.2 extended
c stop when gas is fully neutral
stop efrac -10
c stop when gas is cold
stop temperature 10 linear
c stop at thickness of 16.5 so that mole limit does not
c stop this calculation - that would make results very
c detail dependent
stop thickness 19
c
c commands for density & abundances =========
hden 5.362
c this turns off some elements, and processes we don't need
c so needs to come before the elements
init file="ism.ini"
abundances he -1.01 c -3.52 n-8 o-3.30 ne-8 mg-5.89
continue si -6.10 s -5.10 cl-7 ar-8 fe -6.60
c use ism abundances but turn of quantum heating
```

```
grains ism, abundance log 0.16 no qheat
c
c other commands for details =========
c must iterate since many FIR MIR lines are optically thick
iterate
c this is a closed geometry, in Orion, veil covers 2pi sr
sphere
c make Lyman lines very optically thick, which stops pumping by
c Balmer continuum
case b
c this should run cleanly - turn down number of allow conv fails
failures 3
c
c commands controlling output =========
normalize to "C 2" 157.6m
c want the spectrum to be relative to this [C II] line
c uv lines are strongly pumped by stellar continuum, break out contribution
print line pump
print line optical depths
print ages
punch overview last "pdr_th85ism.ovr"
punch pdr last "pdr_th85ism.pdr"
punch molecules last "pdr_th85ism.mol"
punch dr last "pdr_th85ism.dr"
punch hydrogen physical conditions last "pdr_th85ism.hyd"
punch results last "pdr_th85ism.rlt"
punch continuum units microns last "pdr_th85ism.con"
punch heating last "pdr_th85ism.het"
punch cooling last "pdr_th85ism.col"
punch hydrogen 21 cm last "pdr_th85ism.21cm"
punch element magnesium last "pdr_th85ism.mag"
punch grain extinction last "pdr_th85ism.grnext"
punch grain potential last "pdr_th85ism.grnpot"
punch grain temperature last "pdr_th85ism.grntem"
punch grain charge last "pdr_th85ism.grnchr"
c
table lines "LineList_PDR.dat"
c pdr_th85ism.in
c class pdr
c =========================================
c
```

This is the Tielens and Hollenbach (1985a, b) standard model of the Orion photodissociation region (PDR). Specifically, this is my attempt at their Table 2 of paper 2, ApJ 291, p749. The case b command appears because the Lyman lines are vastly optically thick in this environment. If the command is removed the Lyman lines will be optically thin, which will result in fast fluorescent excitation of the (nearly totally neutral) hydrogen atom.

// >>refer model pdr Tielens, A. G. G. M., & Hollenbach, D. 1985a, ApJ, 291, 722 // >>refer model pdr Tielens, A. G. G. M., & Hollenbach, D. 1985b, ApJ, 291, 746

`pdr_th85ism_cgto` *TH85 pdr with ISM grains and C > O*

```
title TH85 pdr with ISM grains and C > O
c
c commands controlling continuum =========
c cosmic radiation background
background z=0
c cosmic ray background ionization and heating
```

```
          cosmic rays, background
          c first continuum is FIR hot grain continuum produced in
          c unmodeled HII Region
          blackbody, t = 75 K
          intensity 2.7 (total)
          c this is the attenuated hot stellar continuum
          black 30000
          intensity 2.204 range 0.4412 to 1 Ryd
          c this will remove all ionizing radiation
          extinguish 24 0
          c
          c commands controlling geometry =========
          c simulate effects of gas we do not model
          double
          c their turbulence
          turbulence 2.7 km/sec
          c stopping criteria
          c this should be the one actually used
          stop AV 33.2 extended
          c stop when gas is fully neutral
          stop efrac -10
          c stop when gas is cold
          stop temperature 10 linear
          c stop at thickness of 16.5 so that mole limit does not
          c stop this calculation - that would make results very
          c detail dependent
          stop thickness 19
          c
          c commands for density & abundances =========
          hden 5.362
          c this turns off some elements, and processes we don't need
          c so needs to come before the elements
          init file="ism.ini"
          c this is the TH85 mix except C is raised by 0.2 dex,
          c and O lowered byt his amount, so that C/O > 1
          abundances he -1.01 c -3.4 n-8 o-3.50 ne-8 mg-5.89
          continue si -6.10 s -5.10 cl-7 ar-8 fe -6.60
          c use ism abundances but turn of quantum heating
          grains ism, abundance log 0.16 no qheat
          c
          c other commands for details =========
          c must iterate since many FIR MIR lines are optically thick
          iterate
          c this is a closed geometry, in Orion, veil covers 2pi sr
          sphere
          c make Lyman lines very optically thick, which stops pumping by
          c Balmer continuum
          case b
          c this should run cleanly - turn down number of allow conv fails
          failures 3
          c
          c commands controlling output =========
          normalize to "C 2" 157.6m
          c want the spectrum to be relative to this [C II] line
          c uv lines are strongly pumped by stellar continuum, break out contribution
          print line pump
          print line optical depths
          print ages
          punch overview last "pdr_th85ism_cgto.ovr"
          punch pdr last "pdr_th85ism_cgto.pdr"
          punch molecules last "pdr_th85ism_cgto.mol"
          punch dr last "pdr_th85ism_cgto.dr"
          punch hydrogen physical conditions last "pdr_th85ism_cgto.hyd"
          punch results last "pdr_th85ism_cgto.rlt"
          punch continuum units microns last "pdr_th85ism_cgto.con"
```

```
punch heating last "pdr_th85ism_cgto.het"
punch cooling last "pdr_th85ism_cgto.col"
punch hydrogen 21 cm last "pdr_th85ism_cgto.21cm"
punch element magnesium last "pdr_th85ism_cgto.mag"
punch grain extinction last "pdr_th85ism_cgto.grnext"
punch grain potential last "pdr_th85ism_cgto.grnpot"
punch grain temperature last "pdr_th85ism_cgto.grntem"
punch grain charge last "pdr_th85ism_cgto.grnchr"
c
c pdr_th85ism_cgto.in
c class pdr
c =========================================
c
```

This is the Tielens and Hollenbach (1985a, b) standard model of the Orion photodissociation region (PDR) but with the C abundance increased by 0.2 dex, and O lowered by this amount, so that C/O>1. Specifically, this is their Table 2 of paper 2, ApJ 291, p749. The case b command appears because the Lyman lines are vastly optically thick in this environment. If the command is removed the Lyman lines will be optically thin, which will result in fast fluorescent excitation of the (nearly totally neutral) hydrogen atom.

// >>refer model pdr Tielens, A. G. G. M., & Hollenbach, D. 1985a, ApJ, 291, 722 // >>refer model pdr Tielens, A. G. G. M., & Hollenbach, D. 1985b, ApJ, 291, 746

---

pdr_th85orion *TH85 PDR with Orion grains*

```
title TH85 PDR with Orion grains
c
c commands controlling continuum =========
c cosmic background
background
c galactic cosmic ray background
cosmic rays, background
c first continuum is FIR hot grain continuum produced in
c unmodeled HII Region
blackbody, t = 75 K
intensity 2.7 (total)
c this is the attenuated hot stellar continuum
black 30000
intensity 2.204 range 0.4412 to 1 Ryd
c this will remove all ionizing radiation
extinguish 24 0
c
c commands for density & abundances =========
hden 5.362
init file="ism.ini"
grains orion, abundance log 0.16 no qheat
abundances he -1.01 c -3.52 n-8 o-3.30 ne-8 mg-5.89
continue si -6.10 s -5.10 cl-7 ar-8 fe -6.60
c
c commands controlling geometry =========
c simulate effects of gas we do not model
double
sphere
c stopping criteria
c this should be the one actually used
c >>chn 04 mar 13, from 10 to 36.3, bug in AV had stopped at this depth,
c so change it to this to keep model the same
stop AV 36.3 extended
c stop when gas is fully neutral
```

```
stop efrac -10
c stop when gas is cold
stop temperature 10 linear
c stop at thickness so that would make results very
c detail dependent
stop thickness 19
c
c other commands for details =========
turbulence 2.7 km/sec
iterate
failures 3
c
c this is done to not allow pumping and subsequent photoionization
c of H by Balmer continuum in keeping with std pdr assumptions
case b
c
c commands controlling output =========
normalize to "C 2" 157.6m
c uv lines are strongly pumped by stellar continuum, break out contribution
print line pump
print line optical depths
print ages
c
punch overview "pdr_th85orion.ovr" no hash
punch pdr last "pdr_th85orion.pdr"
c this only outputs major molecules
punch molecules last "pdr_th85orion.mol"
punch dr "pdr_th85orion.dr"
c this is to get all of H molecules
punch hydrogen physical conditions last "pdr_th85orion.hyd"
punch results last "pdr_th85orion.rlt"
punch continuum units microns last "pdr_th85orion.con"
punch ionizing continuum last "pdr_th85orion.ion"
punch heating last "pdr_th85orion.het"
punch cooling last "pdr_th85orion.col"
punch hydrogen 21 cm last "pdr_th85orion.21cm"
c these are for the grains
punch grain extinction last "pdr_th85orion.grnext"
punch grain potential last "pdr_th85orion.grnpot"
punch grain temperature last "pdr_th85orion.grntem"
punch grain charge last "pdr_th85orion.grnchr"
c
c
c pdr_th85orion.in
c class pdr
c =========================================
```

This is the Tielens and Hollenbach (1985a, b) standard model of the Orion photodissociation region (PDR). Specifically, this is my attempt at their Table 2 of paper 2, ApJ 291, p749. The case b command appears because the Lyman lines are vastly optically thick in this environment. If the command is removed the Lyman lines will be optically thin, which will result in fast fluorescent excitation of the (nearly totally neutral) hydrogen atom.

// >>refer model pdr Tielens, A. G. G. M., & Hollenbach, D. 1985a, ApJ, 291, 722 // >>refer model pdr Tielens, A. G. G. M., & Hollenbach, D. 1985b, ApJ, 291, 746

---

pn_fluc *Paris PN with density fluctuations*

```
title Paris PN with density fluctuations
```

```
c
c commands for density & abundances =========
fluctuations 16.5 4 3
c this sets up the code like version c84, with fewer chemical elements
init file="c84.ini"
c this sets the abundances of the elements that we will use
abund -1 C-3.523 N-4. O-3.222 ne-3.824 na=-10 mg-4.523 al=-10
continue si-4.523 s-4.824 ar-10 ca=-10 fe-10 ni=-10
c
c commands controlling continuum =========
black body, T = 150000 K, radius = 10
black 5
luminosity total 38
c
c commands controlling geometry =========
sphere
radius 17
c
c other commands for details =========
c
c commands controlling output =========
punch overview last "pn_fluc.ovr"
punch element nitrogen last "pn_fluc.nit"
punch dr last "pn_fluc.dr"
punch results last "pn_fluc.rlt"
c
c pn_fluc.in
c class pn function
c ========================================
```

The boundary conditions are similar to those for the Paris meeting PN, a homogeneous grain-free PN. This model has density fluctuations, with values chosen so that the mean ne np V are the same for the nebulae with and without fluctuations.

Checks:
- Check that the fluctuations command works.
- How do results compare with homogeneous Paris pn?

pn_ots *Paris PN with ots*

```
title Paris PN with ots
c recompute "standard" PN model of the Pequignot Meudon Conferance
c
c commands controlling continuum =========
black body, T = 150000 K, radius = 10
c
c commands for density & abundances =========
hden = 3.4771213
init file="c84.ini"
abund -1 C-3.523 N-4. O-3.222 ne-3.824 na=-10 mg-4.523 al=-10
continue si-4.523 s-4.824 ar-10 ca=-10 fe-10 ni=-10
c
c commands controlling geometry =========
radius = 17
sphere
c
c other commands for details =========
diffuse ots
```

```
no level2
iterate
atom h-like element hydrogen levels resolved 10
atom h-like element helium levels resolved 15
c
c commands controlling output =========
punch overview last "pn_ots.ovr"
punch element nitrogen last "pn_ots.nit"
punch results last "pn_ots.rlt"
punch dr last "pn_ots.dr"
c
c pn_ots.in
c class pn
c ========================================
c
```

This is the on-the-spot version of the Paris Planetary Nebula.
Checks:
- Q(H) total 4861 luminosity is close to expected value
- Line spectrum similar to that predicted by default conditions.

---

pn_paris *Paris meeting PN*

```
title Paris meeting PN
c standard" PN model of the Pequignot Meudon Conferance
c
c commands controlling continuum =========
black body, T = 150000 K, radius = 10
c
c commands for density & abundances =========
hden = 3.4771213
init file="ism.ini"
abund he -1 C-3.523 N-4. O-3.222 ne-3.824 mg-4.523
continue si-4.523 s-4.824 cl-7 ar-10 fe-10
c
c commands controlling geometry =========
radius = 17
sphere
c
c other commands for details =========
age 10000 years
c
c commands controlling output =========
normalize to "Ca b" 4861
print line optical depths
print ages
print column densities
punch overview "pn_paris.ovr"
punch transmitted continuum "pn_paris.trn"
punch continuum units microns "pn_paris.con"
punch element carbon "pn_paris.car"
punch physical conditions "pn_paris.phy"
punch element nitrogen "pn_paris.nit"
punch dr "pn_paris.dr"
punch temperature "pn_paris.tem"
punch lines emissivity "pn_paris.ems"
h 1 6563
s 2 6720
end of lines
punch averages "pn_paris.avr" last
```

```
temperature hydrogen 1
end of averagers
c
c pn_paris.in
c class pn
c =========================================
c
```

This is one of the "standard" models computed at the Paris meeting on photoionization and shock calculations. A table in Hazy compares the predictions of the current version of CLOUDY with predictions of a few of the other codes.

The model is meant to represent a planetary nebula ionized by a very hot central object. As a result there is a great deal of He++ and the associated line and continuum emission is very important.

An age of 10000 years was set. This tests logic in the age command.

---

pn_paris_fast *Paris PN, fast version*

```
title Paris PN, fast version
c standard" PN model of the Pequignot Meudon Conferance
c
c commands controlling continuum =========
black body, T = 150000 K, radius = 10
c
c commands for density & abundances =========
hden = 3.4771213
init file="fast.ini"
abund he -1 C-3.523 N-4. O-3.222 ne-3.824 mg-4.523
continue si-4.523 s-4.824 ar-10
c
c commands controlling geometry =========
sphere
radius = 17
c
c other commands for details =========
c
c commands controlling output =========
normalize to "Ca b" 4861
punch overview "pn_paris_fast.ovr"
punch physical conditions "pn_paris_fast.phy"
punch element nitrogen "pn_paris_fast.nit"
punch dr "pn_paris_fast.dr"
c
c pn_paris_fast.in
c class pn
c =========================================
c
```

This is one of the "standard" models computed at the Paris meeting on photoionization and shock calculations. A table in Hazy compares the predictions of the current version of CLOUDY with predictions of a few of the other codes.

The model is meant to represent a planetary nebula ionized by a very hot central object. As a result there is a great deal of He++ and the associated line and continuum emission is very important.

`pn_sqrden` *PN with density propt R^-2 and filling factor*

```
title PN with density propt R^-2 and filling factor
c this is an optically thin model, which will have a broad range
c of density but the same ionization parameter across the model.
c the physical condtions should not vary by much across the structure
c
c commands controlling continuum =========
c continuum is hot blackbody at ~Ledd for solar mass
blackbody, t=5 luminosity=38
c
c commands for density & abundances =========
c the density will fall off as the inverse square of the radius, hence the name
hden 5 -2
init file "ism.ini"
c use default PN abundances but turn off quantum heating since
c we are not going to save the predicted near IR continuum
abundances planetary no qheat
c
c commands controlling geometry =========
c small filling factor so that region will be optically thin
filling factor -2
c set inner and outer radius
radius 16 18
sphere
c
c other commands for details =========
iterate
c
c commands controlling output =========
print line faint 1
punch overview "pn_sqrden.ovr" last
punch dr "pn_sqrden.dr" last
c
c pn_sqrden.in
c class pn
c =========================================
```

Checks:
- Zone thickness budgeting handled OK.
- physical conditions nearly constant across computed structure

---

`stars_atlas` *Atlas stellar atmosphere*

```
title Atlas stellar atmosphere
c
c commands controlling continuum =========
c Approximate model of a hot star ionizing its surroundings.
c Assume H density of 1e+6 per cc. This run tests the new Kurucz model option.
c I choose the T_eff and log(g) values arbitrarily. It implies a star
c with luminosity 32300 L_sun, radius 6.4745 R_sun, and mass 3.745 M_sun.
table star atlas 30400.0 4.2
luminosity 4.509202522 solar
c
c commands for density & abundances =========
abundances old solar 84
hden 6.0
```

```
grains ism
c
c commands controlling geometry =========
radius -3.0 parsec
sphere
stop zone 1
set dr 0
c
c other commands for details =========
constant temperature 4
c
c commands controlling output =========
print continuum
c
punch overview last "stars_atlas.ovr"
punch results last "stars_atlas.rlt"
punch transmitted continuum last "stars_atlas.trn" no header
punch continuum last "stars_atlas.con" units microns
c
c stars_atlas.in
c class stars
c ========================================
```

This is a test that the code can correctly access the large block of Atlas model atmosphere continua described by Kurucz (1991). Kevin Volk provided it as part of his original coding of these stellar atmosphere files. Constant temperature is assumed since this tests shape of stellar continuum rather than thermal physics.

Checks: table star atlas command works.

---

stars_atlas_3d *Atlas stellar atmosphere, 3D interpolation*

```
title Atlas stellar atmosphere, 3D interpolation
c
c commands controlling continuum =========
c Approximate model of a hot star ionizing its surroundings.
c Assume H density of 1e+6 per cc. This run tests the atlas 3dim model option.
c I choose the T_eff and log(g) values arbitrarily.
table star atlas 3-dim 38400.0 4.82 -1.42
luminosity 4.509202522 solar
table star atlas odfnew 3-dim 38400.0 4.82 0.42
luminosity 4.509202522 solar
c
c commands for density & abundances =========
abundances old solar 84
hden 6.0
grains ism
c
c commands controlling geometry =========
radius -3.0 parsec
sphere
stop zone 1
set dr 0
c
c other commands for details =========
constant temperature 4
c
c commands controlling output =========
c
punch overview last "stars_atlas_3d.ovr"
```

```
punch results last "stars_atlas_3d.rlt"
punch transmitted continuum last "stars_atlas_3d.trn" no header
punch continuum last "stars_atlas_3d.con" units microns
c
c stars_atlas_3d.in
c class stars
c =======================================
```

This is a test that the code can correctly access the large 3-dim block of Atlas model atmosphere continua.

Checks: table star atlas 3dim command works.

---

stars_atlas_all *interpolate on Atlas grids*


```
title interpolate on Atlas grids
c
c commands controlling continuum =========
c Include stars with a wide range of metallicities to test access to these grids.
table star atlas Z+1.0 38400.0 4.82
luminosity 4.509202522 solar
table star atlas Z+0.5 38400.0 4.82
luminosity 4.509202522 solar
table star atlas Z+0.3 38400.0 4.82
luminosity 4.509202522 solar
table star atlas Z+0.2 38400.0 4.82
luminosity 4.509202522 solar
table star atlas Z+0.1 38400.0 4.82
luminosity 4.509202522 solar
table star atlas Z+0.0 38400.0 4.82
luminosity 4.509202522 solar
table star atlas Z-0.1 38400.0 4.82
luminosity 4.509202522 solar
table star atlas Z-0.2 38400.0 4.82
luminosity 4.509202522 solar
table star atlas Z-0.3 38400.0 4.82
luminosity 4.509202522 solar
table star atlas Z-0.5 38400.0 4.82
luminosity 4.509202522 solar
table star atlas Z-1.0 38400.0 4.82
luminosity 4.509202522 solar
table star atlas Z-1.5 38400.0 4.82
luminosity 4.509202522 solar
table star atlas Z-2.0 38400.0 4.82
luminosity 4.509202522 solar
table star atlas Z-2.5 38400.0 4.82
luminosity 4.509202522 solar
table star atlas Z-3.0 38400.0 4.82
luminosity 4.509202522 solar
table star atlas Z-3.5 38400.0 4.82
luminosity 4.509202522 solar
table star atlas Z-4.0 38400.0 4.82
luminosity 4.509202522 solar
table star atlas Z-4.5 38400.0 4.82
luminosity 4.509202522 solar
table star atlas Z-5.0 38400.0 4.82
luminosity 4.509202522 solar
c
c commands for density & abundances =========
abundances old solar 84
```

```
hden 6.0
grains ism
c
c commands controlling geometry =========
radius -2.0 parsec
sphere
stop zone 1
set dr 0
c
c other commands for details =========
constant temperature 4
c
c commands controlling output =========
c
punch overview last "stars_atlas_all.ovr"
punch results last "stars_atlas_all.rlt"
punch transmitted continuum last "stars_atlas_all.trn" no header
punch continuum last "stars_atlas_all.con" units microns
c
c stars_atlas_all.in
c class stars
c ========================================
```

This is a test that the code can correctly access the 2D grids of various metallicity of Atlas model atmosphere continua.

Checks: access to the various atlas grids

`stars_atlas_odfnew_all` *access various Atlas ODFNEW grids*

```
title access various Atlas ODFNEW grids
c
c commands controlling continuum =========
c Include stars with a wide range of metallicities to test access to these grids.
table star atlas odfnew Z+0.5 38400.0 4.82
luminosity 4.509202522 solar
table star atlas odfnew Z+0.2 38400.0 4.82
luminosity 4.509202522 solar
table star atlas odfnew Z+0.0 38400.0 4.82
luminosity 4.509202522 solar
table star atlas odfnew Z-0.5 38400.0 4.82
luminosity 4.509202522 solar
table star atlas odfnew Z-1.0 38400.0 4.82
luminosity 4.509202522 solar
table star atlas odfnew Z-1.5 38400.0 4.82
luminosity 4.509202522 solar
table star atlas odfnew Z-2.0 38400.0 4.82
luminosity 4.509202522 solar
table star atlas odfnew Z-2.5 38400.0 4.82
luminosity 4.509202522 solar
c
c commands for density & abundances =========
abundances old solar 84
hden 6.0
grains ism
c
c commands controlling geometry =========
radius -2.0 parsec
sphere
```

```
      stop zone 1
      set dr 0
      c
      c other commands for details =========
      constant temperature 4
      c
      c commands controlling output =========
      c
      punch overview last "stars_atlas_odfnew_all.ovr"
      punch results last "stars_atlas_odfnew_all.rlt"
      punch transmitted continuum last "stars_atlas_odfnew_all.trn" no header
      punch continuum last "stars_atlas_odfnew_all.con" units microns
      c
      c stars_atlas_odfnew_all.in
      c class stars
      c ========================================
```

This is a test that the code can correctly access the Atlas ODFNEW model atmosphere continua.

Checks: access to the various atlas odfnew grids

_____

`stars_bstar2006` *tlusty bstar grid*


```
      title tlusty bstar grid
      c
      c commands controlling continuum =========
      c Approximate model of a B star ionizing its surroundings.
      c Assume H density of 1e+6 per cc. This run tests the new Tlusty model option.
      c I choose the T_eff and log(g) values arbitrarily.
      table star tlusty bstar 20400.0 4.2
      luminosity 4.509202522 solar
      c
      c commands for density & abundances =========
      abundances old solar 84
      hden 6.0
      grains ism
      c
      c commands controlling geometry =========
      radius -3.0 parsec
      sphere
      stop zone 1
      set dr 0
      c
      c other commands for details =========
      c
      c in this model the Lyman lines are pumped significantly by the stellar continuum;
      c disable this process to avoid critical dependence on the precise shape of the
      continuum.
      no induced processes
      constant temperature 4
      c
      c commands controlling output =========
      c
      punch overview last "stars_bstar2006.ovr"
      punch results last "stars_bstar2006.rlt"
      punch transmitted continuum last "stars_bstar2006.trn" no header
      c
      c stars_bstar2006.in
      c class stars
```

```
c ========================================
```

This is a test that the code can correctly access the large block of Tlusty B-star model atmosphere continua.

Checks: table star tlusty bstar command works.

---

stars_bstar2006_3d *tlusty bstar grid*

```
title tlusty bstar grid
c
c commands controlling continuum =========
c Approximate model of a B star ionizing its surroundings.
c Assume H density of 1e+6 per cc. This run tests the new Tlusty model option.
c I choose the T_eff, log(g) and log(Z) values arbitrarily.
table star tlusty bstar 3-dim 20400.0 4.2 -0.573
luminosity 4.509202522 solar
c
c commands for density & abundances =========
abundances old solar 84
hden 6.0
grains ism
c
c commands controlling geometry =========
radius -3.0 parsec
sphere
stop zone 1
set dr 0
c
c other commands for details =========
c
c in this model the Lyman lines are pumped significantly by the stellar continuum;
c disable this process to avoid critical dependence on the precise shape of the
continuum.
no induced processes
constant temperature 4
c
c commands controlling output =========
c
punch overview last "stars_bstar2006_3d.ovr"
punch results last "stars_bstar2006_3d.rlt"
punch transmitted continuum last "stars_bstar2006_3d.trn" no header
c
c stars_bstar2006_3d.in
c class stars
c ========================================
```

This is a test that the code can correctly access the large 3-dim block of Tlusty B-star model atmosphere continua.

Checks: table star tlusty bstar 3-dim command works.

---

stars_bstar2006_all *tlusty bstar grid*

```
title tlusty bstar grid
c
c commands controlling continuum =========
c Approximate model of 6 B stars ionizing their surroundings.
c Assume H density of 1e+6 per cc. This run tests all the various
c metallicities contained in the BSTAR 2006 grids. I choose the T_eff
c and log(g) values arbitrarily.
table star tlusty bstar Z+0.3 20000.0 4.5
luminosity 4.509202522 solar
table star tlusty bstar Z+0.0 21400.0 4.5
luminosity 4.509202522 solar
table star tlusty bstar Z-0.3 15000.0 4.2
luminosity 4.509202522 solar
table star tlusty bstar Z-0.7 20400.0 4.2
luminosity 4.509202522 solar
table star tlusty bstar Z-1.0 27500.0 3.0
luminosity 4.509202522 solar
table star tlusty bstar Z-INF 18000.0 4.75
luminosity 4.509202522 solar
c
c commands for density & abundances =========
abundances old solar 84
hden 6.0
grains ism
c
c commands controlling geometry =========
radius -2.0 parsec
sphere
stop zone 1
set dr 0
c
c other commands for details =========
c
c in this model the Lyman lines are pumped significantly by the stellar continuum;
c disable this process to avoid critical dependence on the precise shape of the
continuum.
no induced processes
constant temperature 4
c
c commands controlling output =========
c
punch overview last "stars_bstar2006_all.ovr"
punch results last "stars_bstar2006_all.rlt"
punch transmitted continuum last "stars_bstar2006_all.trn" no header
c
c stars_bstar2006_all.in
c class stars
c =========================================
```

This is a test that the code can correctly access the large block of Tlusty B-star model atmosphere continua.

Checks: table star tlusty bstar metallicity option works.

---

stars_costar1 *costar interpolation mode 1*


```
title costar interpolation mode 1
c
c commands controlling continuum =========
table star costar, 40000 K
```

```
ionization parameter -2
table star costar, 31000 K, index 4
ionization parameter -2
c
c commands for density & abundances =========
hden 2
grains orion single no qheat
abundances ism no grains
c
c commands controlling geometry =========
set dr 0
stop zone 1
c
c other commands for details =========
* >>chng 06 jul 01, add this to stars set since want to test shape
* of continuum and resulting ion dist, not thermal equilibrium of nebula
constant temperature 6000K
c
c commands controlling output =========
punch continuum "stars_costar1.con" units microns last
c
c stars_costar1.in
c class stars
c =======================================
c
```

This is a test of whether the code can read the CoStar stellar atmospheres. It checks the resulting ionization to confirm that the continuum shape was ok. Constant temperature is assumed since this tests shape of stellar continuum rather than thermal physics.

---

stars_costar2 *costar interpolation mode 2*

```
title costar interpolation mode 2
c
c commands controlling continuum =========
table star costar 35575.4 4.2367
ionization parameter -2
table star costar 34468.5 3.86765
ionization parameter -2
table star costar 41741.2 4.02
ionization parameter -2
table star costar 45000 4.1593
ionization parameter -2
table star costar 27500 3.3
ionization parameter -2
c
c commands for density & abundances =========
hden 2
grains orion single no qheat
abundances ism no grains
c
c commands controlling geometry =========
set dr 0
stop zone 1
c
c other commands for details =========
c
c commands controlling output =========
punch continuum "stars_costar2.con" units microns last
c
```

```
c stars_costar2.in
c class stars
C =======================================
c
```

This is a test of whether the code can read the CoStar stellar atmospheres. It checks the resulting ionization to confirm that the continuum shape was ok.

---

stars_costar3 *costar interpolation mode 3*

```
title costar interpolation mode 3
c
c commands controlling continuum =========
table star costar zams 40.0 Msol 2.85e6
ionization parameter -2
table star costar zams 40.0 Msol 4e6
ionization parameter -2
table star costar zams 50.9 Msol 2.85e6
ionization parameter -2
table star costar zams 42.9 Msol 1.6e6
ionization parameter -2
table star costar zams 70.5 Msol 2.1e6
ionization parameter -2
c
c commands for density & abundances =========
hden 2
grains orion single no qheat
abundances ism no grains
c
c commands controlling geometry =========
set dr 0
stop zone 1
c
c other commands for details =========
c
c commands controlling output =========
punch continuum "stars_costar3.con" units microns last
c
c stars_costar3.in
c class stars
C =======================================
c
```

This is a test of whether the code can read the CoStar stellar atmospheres. It checks the resulting ionization to confirm that the continuum shape was ok.

---

stars_costar4 *costar interpolation mode 4*

```
title costar interpolation mode 4
c
c commands controlling continuum =========
table star costar age 2.85e6 40.0
ionization parameter -2
table star costar age 4e6 40.0
ionization parameter -2
table star costar age 2.85e6 50.9
```

```
ionization parameter -2
table star costar age 1.6e6 42.9
ionization parameter -2
table star costar age 2.1e6 70.5
ionization parameter -2
c
c commands for density & abundances =========
hden 2
grains orion single no qheat
abundances ism no grains
c
c commands controlling geometry =========
set dr 0
stop zone 1
c
c other commands for details =========
c
c commands controlling output =========
punch continuum "stars_costar4.con" units microns last
c
c stars_costar4.in
c class stars
c =======================================
c
```

This is a test of whether the code can read the CoStar stellar atmospheres. It checks the resulting ionization to confirm that the continuum shape was ok.

`stars_costarhalo` *costar halo abundances*

```
title costar halo abundances
c
c commands controlling continuum =========
table star costar, halo abundances, 40000 K
ionization parameter -2
c
c commands for density & abundances =========
hden 2
abundances ism
c
c commands controlling geometry =========
set dr 0
stop zone 1
c
c other commands for details =========
* >>chng 06 jul 01, add this to stars set since want to test shape
* of continuum and resulting ion dist, not thermal equilibrium of nebula
constant temperature 6900K
c
c commands controlling output =========
punch continuum last "stars_costarhalo.con" units microns
c
c
c stars_costarhalo.in
c class stars
c =======================================
c
```

This test checks that the code can read the CoStar stellar atmospheres. It checks the resulting ionization to make sure that the continuum shape is ok. Constant temperature is assumed since this tests shape of stellar

continuum rather than thermal physics.

---

stars_kurucz79 *Kurucz 1979 SED*

```
title Kurucz 1979 SED
c
c commands controlling continuum =========
c Approximate model of a hot star ionizing its surroundings.
c Assume H density of 1e+6 per cc. This run tests the Kurucz model option.
c I choose the T_eff and log(g) values arbitrarily. It implies a star
c with luminosity 32300 L_sun, radius 6.4745 R_sun, and mass 3.745 M_sun.
table star kurucz 30400.0
luminosity 4.509202522 solar
c
c commands for density & abundances =========
abundances old solar 84
hden 6.0
grains ism
c
c commands controlling geometry =========
radius -3.0 parsec
sphere
stop zone 1
set dr 0
c
c other commands for details =========
constant temperature 4
c
c commands controlling output =========
c
punch overview last "stars_kurucz79.ovr"
punch results last "stars_kurucz79.rlt"
punch transmitted continuum last "stars_kurucz79.trn" no header
punch continuum last "stars_kurucz79.con" units microns
c
c stars_kurucz79.in
c class stars
c =========================================
```

This is a test that the code can correctly access the Kurucz model atmosphere continua described by Kurucz (1979). Kevin Volk provided it as part of his original coding of these stellar atmosphere files. Constant temperature is assumed since this tests shape of stellar continuum rather than thermal physics.

Checks: table star kurucz command works.

---

stars_mihalas *Mihalas SED*

```
title Mihalas SED
c
c commands controlling continuum =========
c Approximate model of a hot star ionizing its surroundings.
c Assume H density of 1e+6 per cc. This run tests the new Kurucz model option.
c I choose the T_eff and log(g) values arbitrarily. It implies a star
c with luminosity 32300 L_sun, radius 6.4745 R_sun, and mass 3.745 M_sun.
```

```
table star mihalas 30400.0
luminosity 4.509202522 solar
c
c commands for density & abundances =========
abundances old solar 84
hden 6.0
grains ism
c
c commands controlling geometry =========
radius -3.0 parsec
sphere
stop zone 1
set dr 0
c
c other commands for details =========
constant temperature 4
c
c commands controlling output =========
c
punch overview last "stars_mihalas.ovr"
punch results last "stars_mihalas.rlt"
punch transmitted continuum last "stars_mihalas.trn" no header
punch continuum last "stars_mihalas.con" units microns
c
c stars_mihalas.in
c class stars
c =========================================
```

This is a test that the code can correctly access the small Mihalas grid of NLTE model atmospeheres Constant temperature is assumed since this tests shape of continuum not thermal physics.

Checks: table star mihalas command works.

`stars_optimize1` *optimizer on stellar grids*


```
title optimizer on stellar grids
c
c commands controlling continuum =========
table star tlusty ostar 3-dim 31200 3.6 -1.12 vary
ionization par -3
c
c commands for density & abundances =========
hden 4 vary
init file "hheonly.ini"
element oxygen on
c
c commands controlling geometry =========
case B
stop zone 1
c
c other commands for details =========
optimize tolerance 0.03
constant temper 4.3 vary
optimize phymir sequential
optimize iteration 500
c the following spectrum was derived by running this sim at hden 5, t=4,
c and T_star=51200
c resulting eden was 5.040 and T=4, which we shall assert that we find
optimize lines
```

```
o ii 3729 0.0416
o ii 3726 0.1179
o ii 7323 0.1535
o ii 7332 0.1237
o 3 5007 9.5265
totl 4363 0.0900
end of lines
c
c commands controlling output =========
print line faint -2
c
c
c optimize_phymir.in
c class optimizer
c ========================================
c
```

This checks whether the optimizer sets corrects limits for Teff. The line spectrum was calculated at T = 1e4 K, n_H=1e5 cm^-3, T_star = 51200 K.

---

stars_optimize2 *optimizer on stellar grids*

```
title optimizer on stellar grids
c
c commands controlling continuum =========
table star costar 36200 vary
ionization par -3
c
c commands for density & abundances =========
hden 4 vary
init file "hheonly.ini"
element oxygen on
c
c commands controlling geometry =========
Case B
stop zone 1
c
c other commands for details =========
optimize tolerance 0.03
constant temper 4.3 vary
optimize phymir sequential
optimize iteration 500
c the following spectrum was derived by running this sim at hden 5, t=4,
c and T_star=51200
c resulting eden was 5.041 and T=4, which we shall assert that we find
optimize lines
o ii 3729 0.0335
o ii 3726 0.0950
o ii 7323 0.1238
o ii 7332 0.0998
o 3 5007 9.9714
totl 4363 0.0943
end of lines
c
c commands controlling output =========
print line faint -2
c
c
c optimize_phymir.in
c class optimizer
```

```
c =======================================
c
```

This checks whether the optimizer sets corrects limits for Teff. The line spectrum was calculated at T = 1e4 K, n_H=1e5 cm^-3, T_star = 51200 K.

---

stars_optimize3 *optimizer on stellar grids*

```
title optimizer on stellar grids
c
c commands controlling continuum =========
table star costar age 5.8 log 36 msol vary
ionization par -3
c
c commands for density & abundances =========
hden 4 vary
init file "hheonly.ini"
element oxygen on
c
c commands controlling geometry =========
Case B
stop zone 1
c
c other commands for details =========
optimize tolerance 0.03
constant temper 4.3 vary
optimize phymir sequential
optimize iteration 500
c the following spectrum was derived by running this sim at hden 5, t=4,
c and log(age)=6.3
c resulting eden was 5.040 and T=4, which we shall assert that we find
optimize lines
o ii 3729 0.0623
o ii 3726 0.1765
o ii 7323 0.2300
o ii 7332 0.1854
o 3 5007 8.2979
totl 4363 0.0784
end of lines
c
c commands controlling output =========
print line faint -2
c
c
c optimize_phymir.in
c class optimizer
c =======================================
c
```

This checks whether the optimizer sets corrects limits for log(age). The line spectrum was calculated at T = 1e4 K, n_H=1e5 cm^-3, log(age/yr) = 6.3.

---

stars_ostar2002 *tlusty O star grid*

```
title tlusty O star grid
c
```

```
c commands controlling continuum =========
c Approximate model of a hot star ionizing its surroundings.
c Assume H density of 1e+6 per cc. This run tests the new Tlusty model option.
c I choose the T_eff and log(g) values arbitrarily. It implies a star
c with luminosity 32300 L_sun, radius 6.4745 R_sun, and mass 3.745 M_sun.
table star tlusty ostar 30400.0 4.2
luminosity 4.509202522 solar
c
c commands for density & abundances =========
abundances old solar 84
hden 6.0
grains ism
c
c commands controlling geometry =========
radius -3.0 parsec
sphere
stop zone 1
set dr 0
c
c other commands for details =========
c
c in this model the Lyman lines are pumped significantly by the stellar continuum;
c disable this process to avoid critical dependence on the precise shape of the
continuum.
no induced processes
constant temperature 4
c
c commands controlling output =========
c
punch overview last "stars_ostar2002.ovr"
punch results last "stars_ostar2002.rlt"
punch transmitted continuum last "stars_ostar2002.trn" no header
c
c stars_ostar2002.in
c class stars
c =========================================
```

This is a test that the code can correctly access the large block of Tlusty O-star model atmosphere continua.

Checks: table star tlusty ostar command works.

`stars_ostar2002_3d` *tlusty Ostar grid*

```
title tlusty Ostar grid
c
c commands controlling continuum =========
c Approximate model of a hot star ionizing its surroundings.
c Assume H density of 1e+6 per cc. This run tests the new Tlusty model option.
c I choose the T_eff, log(g) and log(Z) values arbitrarily.
table star tlusty ostar 3-dim 30400.0 4.2 -1.573
luminosity 4.509202522 solar
c
c commands for density & abundances =========
abundances old solar 84
hden 6.0
grains ism
c
c commands controlling geometry =========
radius -3.0 parsec
```

```
sphere
stop zone 1
set dr 0
c
c other commands for details =========
c
c in this model the Lyman lines are pumped significantly by the stellar continuum;
c disable this process to avoid critical dependence on the precise shape of the
continuum.
no induced processes
constant temperature 4
c
c commands controlling output =========
c
punch overview last "stars_ostar2002_3d.ovr"
punch results last "stars_ostar2002_3d.rlt"
punch transmitted continuum last "stars_ostar2002_3d.trn" no header
c
c stars_ostar2002_3d.in
c class stars
c ========================================
```

This is a test that the code can correctly access the large block of Tlusty O-star model atmosphere continua.

Checks: table star tlusty ostar 3-dim command works.

---

stars_ostar2002_all *tlusty Ostar grid*


```
title tlusty Ostar grid
c
c commands controlling continuum =========
c Approximate model of 10 hot stars ionizing their surroundings.
c Assume H density of 1e+6 per cc. This run tests all the various
c metallicities contained in the OSTAR 2002 grids. I choose the T_eff
c and log(g) values such that the selection algorithm gets a good workout.
table star tlusty ostar Z+0.3 30000.0 4.5
luminosity 4.509202522 solar
table star tlusty ostar Z+0.0 31400.0 4.5
luminosity 4.509202522 solar
table star tlusty ostar Z-0.3 45000.0 4.2
luminosity 4.509202522 solar
table star tlusty ostar Z-0.7 30400.0 4.2
luminosity 4.509202522 solar
table star tlusty ostar Z-1.0 27500.0 3.0
luminosity 4.509202522 solar
table star tlusty ostar Z-1.5 55000.0 4.75
luminosity 4.509202522 solar
table star tlusty ostar Z-1.7 27500.0 4.75
luminosity 4.509202522 solar
table star tlusty ostar Z-2.0 55000.0 3.00
luminosity 4.509202522 solar
table star tlusty ostar Z-3.0 44400.0 3.2
luminosity 4.509202522 solar
table star tlusty ostar Z-INF 53000.0 4.75
luminosity 4.509202522 solar
table star tlusty ostar Z+0.0 48400.0 3.87
luminosity 4.509202522 solar
table star tlusty ostar Z-0.7 48400.0 3.55
luminosity 4.509202522 solar
```

```
table star tlusty ostar Z+0.3 43400.0 3.65
luminosity 4.509202522 solar
c
c commands for density & abundances =========
abundances old solar 84
hden 6.0
grains ism
c
c commands controlling geometry =========
radius -2.0 parsec
sphere
stop zone 1
set dr 0
c
c other commands for details =========
c
c in this model the Lyman lines are pumped significantly by the stellar continuum;
c disable this process to avoid critical dependence on the precise shape of the
continuum.
no induced processes
constant temperature 4
c
c commands controlling output =========
c
punch overview last "stars_ostar2002_all.ovr"
punch results last "stars_ostar2002_all.rlt"
punch transmitted continuum last "stars_ostar2002_all.trn" no header
c
c stars_ostar2002_all.in
c class stars
c =========================================
```

This is a test that the code can correctly access the large block of Tlusty O-star model atmosphere continua.

Checks: table star tlusty ostar metallicity option works, also test selection algorithm.

`stars_rauch_3d` *Rauch 3-dimensional grid*


```
title Rauch 3-dimensional grid
c
c commands controlling continuum =========
table star rauch 3-dim T = 150000 K, g = 6.5 , log(Z) = -0.6
luminosity 4.5 solar
table star rauch old 3-dim T = 233000 K, g = 7.5 , log(Z) = -0.32
luminosity 4.5 solar
c
c commands controlling geometry =========
sphere
radius 18
stop zone 1
set dr 0
c
c commands for density & abundances =========
abundances planetary
hden 3.0
c
c other commands for details =========
no level2
c
```

```
c commands controlling output =========
plot continuum range .1
punch overview last "stars_rauch_3d.ovr"
punch results last "stars_rauch_3d.rlt"
punch continuum last "stars_rauch_3d.con" units microns
c
c stars_rauch_3d.in
c class stars
c =======================================
```

This model checks that the code is able to access the 3-dim versions of the Rauch grids of stellar atmosphere models. Constant temperature is set since this test is intended to test shape of ionizing continuum, not thermal physics.

`stars_rauch_h+he` *Rauch PN central star*

```
title Rauch PN central star
c
c commands controlling continuum =========
c this tests the various interpolation modes of the 3d grid
table star rauch h+he T = 150000 K, g = 6.5 , f(He) = 0.2
luminosity 4.5 solar
table star rauch h+he T = 155000 K, g = 6.5 , f(He) = 0.2
luminosity 4.5 solar
table star rauch h+he T = 150000 K, g = 6.25 , f(He) = 0.2
luminosity 4.5 solar
table star rauch h+he T = 150000 K, g = 6.5 , f(He) = 0.25
luminosity 4.5 solar
table star rauch h+he T = 155000 K, g = 6.25 , f(He) = 0.2
luminosity 4.5 solar
table star rauch h+he T = 155000 K, g = 6.5 , f(He) = 0.25
luminosity 4.5 solar
table star rauch h+he T = 150000 K, g = 6.25 , f(He) = 0.25
luminosity 4.5 solar
table star rauch h+he T = 155000 K, g = 6.25 , f(He) = 0.25
luminosity 4.5 solar
c
c commands controlling geometry =========
sphere
radius 18
stop zone 1
set dr 0
c
c commands for density & abundances =========
abundances planetary
hden 3.0
c
c other commands for details =========
no level2
c
c commands controlling output =========
plot continuum range .1
punch overview last "stars_rauch_h+he.ovr"
punch results last "stars_rauch_h+he.rlt"
punch continuum last "stars_rauch_h+he.con" units microns
c
c stars_rauch_h+he.in
c class stars
c =======================================
```

This is a model of a very hot planetary nebula, and checks that the code is able to access the H+He version of the Rauch grid of stellar atmosphere models.

`stars_rauch_h-ca` *Rauch central star hot PN*

```
title Rauch central star hot PN
c NB
c this uses the H-Ca rauch atmospheres
c
c commands controlling continuum =========
table star old rauch T = 250000 K, g = 7.5
luminosity 4.5 solar
c
c commands controlling geometry =========
sphere
radius 18
stop zone 1
set dr 0
c
c other commands for details =========
no level2
c
c commands for density & abundances =========
hden 3.0
abundances planetary
c
c commands controlling output =========
plot continuum range .1
punch overview last "stars_rauch_h-ca.ovr"
punch results last "stars_rauch_h-ca.rlt"
punch continuum last "stars_rauch_h-ca.con" units microns
c
c stars_rauch_h-ca.in
c class stars
c =======================================
```

This is a model of a very hot planetary nebula, and checks that the code is able to access the H-Ca Rauch grid of stellar atmosphere models.

`stars_rauch_h-ni` *Rauch hot PN star*

```
title Rauch hot PN star
c
c commands controlling continuum =========
table star rauch T = 150000 K, g = 6.5
luminosity 4.5 solar
c
c commands controlling geometry =========
sphere
radius 18
stop zone 1
set dr 0
c
c commands for density & abundances =========
abundances planetary
hden 3.0
```

```
c
c other commands for details =========
no level2
c
c commands controlling output =========
plot continuum range .1
punch overview last "stars_rauch_h-ni.ovr"
punch results last "stars_rauch_h-ni.rlt"
punch continuum last "stars_rauch_h-ni.con" units microns
c
c stars_rauch_h-ni.in
c class stars
c =======================================
```

This is a model of a very hot planetary nebula, and checks that the code is able to access the H-Ni version of the Rauch grid of stellar atmosphere models. Constant temperature is set since this test is intended to test shape of ionizing continuum, not thermal physics.

stars_rauch_helium *Rauch hot PN star*

```
title Rauch hot PN star
c
c commands controlling continuum =========
table star rauch helium T = 240000 K, g = 8.5
luminosity 4.5 solar
c
c commands controlling geometry =========
sphere
radius 18
stop zone 1
set dr 0
c
c commands for density & abundances =========
abundances planetary
hden 3.0
c
c other commands for details =========
no level2
c
c commands controlling output =========
plot continuum range .1
punch overview last "stars_rauch_helium.ovr"
punch results last "stars_rauch_helium.rlt"
c
c stars_rauch_helium.in
c class stars
c =======================================
```

This is a model of a very hot planetary nebula, and checks that the code is able to access the pure helium version of the Rauch grid of stellar atmosphere models.

stars_rauch_hydr *Rauch hot PN star*

```
title Rauch hot PN star
c
c commands controlling continuum =========
```

```
table star rauch hydr T = 240000 K, g = 8.5
luminosity 4.5 solar
c
c commands controlling geometry =========
sphere
radius 18
stop zone 1
set dr 0
c
c commands for density & abundances =========
abundances planetary
hden 3.0
c
c other commands for details =========
no level2
c
c commands controlling output =========
plot continuum range .1
punch overview last "stars_rauch_hydr.ovr"
punch results last "stars_rauch_hydr.rlt"
c
c stars_rauch_hydr.in
c class stars
c =======================================
```

This is a model of a very hot planetary nebula, and checks that the code is able to access the pure hydrogen version of the Rauch grid of stellar atmosphere models.

`stars_rauch_pg1159` *Rauch hot [WR] PN*

```
title Rauch hot [WR] PN
c
c commands controlling continuum =========
table star rauch pg1159 T=150000K g=6.5
luminosity 4.5 solar
c
c commands controlling geometry =========
sphere
radius 18
stop zone 1
set dr 0
c
c commands for density & abundances =========
abundances planetary
hden 3.0
c
c other commands for details =========
no level2
c
c commands controlling output =========
plot continuum range .1
punch overview last "stars_rauch_pg1159.ovr"
punch results last "stars_rauch_pg1159.rlt"
c
c stars_rauch.in
c class stars
c =======================================
```

This is a model of a very hot planetary nebula, and checks that the code is able to access the PG1159 version of the Rauch grid of stellar atmosphere models.

stars_starburst99 *Starburst 99 SED*

```
title Starburst 99 SED
c
c commands controlling continuum =========
c read the table starburst and interploate to an age of 1e8 years
table star log age=8 "starburst99.mod"
luminosity 43
c add background continuum - starburs 99 does not extend
c into fir so code would complain about zero continuum
c adding the cosmic background stops this complaint
background
c
c commands for density & abundances =========
hden 5
c
c commands controlling geometry =========
stop zone 1
radius 20
c
c other commands for details =========
constant temper 4
c
c commands controlling output =========
punch continuum "stars_starburst99.con" last
punch continuum last "stars_starburst99.con" units microns
c
c
c stars_starburst99.in
c class stars
c ========================================
c
```

this is a demonstration of the use of a Starburst 99 spectrum. It was provided to me by anand Srianand, and we used it in Srianand et al. 2003. A constant temperature is set since this is to test shape of continuum not thermal physics. It also tests access to user-supplied grids as the mechanism is the same.

stars_starburst99_2d *Starburst 99 2D SED*

```
title Starburst 99 2D SED
c
c commands controlling continuum =========
c read the table starburst and interpolate to an age of 2.09e6 years
c and a metallicity of 12.
table star log age=6.32 Z=12. "starburst99_2d.mod"
luminosity 43
c add background continuum - starburs 99 does not extend
c into fir so code would complain about zero continuum
c adding the cosmic background stops this complaint
background
c
c commands for density & abundances =========
hden 5
c
c commands controlling geometry =========
stop zone 1
```

```
radius 20
c
c other commands for details =========
constant temper 4
c
c commands controlling output =========
punch continuum "stars_starburst99_2d.con" last
punch continuum last "stars_starburst99_2d.con" units microns
c
c
c stars_starburst99_2d.in
c class stars
c =======================================
c
```

this is a demonstration of the use of a manually crafted Starburst 99 2D grid, allowing for interpolation in both age and metallicity. It was provided by Christophe Morisset. A constant temperature is set since this is to test shape of continuum not thermal physics. It also tests access to user-supplied grids as the mechanism is the same.

`stars_werner` *Werner stars grid*

```
title Werner stars grid
c
c commands controlling continuum =========
table star werner 190000 g=7.5
luminosity total 38
c
c commands for density & abundances =========
hden 4
abundances planetary no grains
grains agm no qheat single
c
c commands controlling geometry =========
radius 17
sphere
set dr 0
stop zone 1
c
c other commands for details =========
* >>chng 06 jul 01, add this to stars set since want to test shape
* of continuum and resulting ion dist, not thermal equilibrium of nebula
constant temperature 4.373
c
c commands controlling output =========
c
punch overview last "stars_werner.ovr"
punch results last "stars_werner.rlt"
punch transmitted continuumlast "stars_werner.trn"
c
c stars_werner.in
c class stars
c =======================================
c
```

This checks that the code can access Kevin Volk39s Werner atmospheres.

Checks:

- table star Werner atmosphere works. constant temperature is set to that this tests shape of stellar continuum

---

`stars_wmbasic` *wmbasic stellar SED*

```
title wmbasic stellar SED
c
c commands controlling continuum =========
c Approximate model of a hot star ionizing its surroundings.
c Assume H density of 1e+6 per cc. This run tests the WMBASIC model option.
c I choose the T_eff, log(g), and log(Z) values arbitrarily.
table star wmbasic 30000.0 4.0 -0.3
luminosity 4.509202522 solar
table star wmbasic 32000.0 4.0 0.0
luminosity 4.509202522 solar
table star wmbasic 40000.0 3.6 -0.3
luminosity 4.509202522 solar
table star wmbasic 35000.0 4.0 -0.15
luminosity 4.509202522 solar
table star wmbasic 57000.0 4.0 0.0
luminosity 4.509202522 solar
table star wmbasic 37000.0 3.4 -0.3
luminosity 4.509202522 solar
table star wmbasic 32000.0 4.0 -0.1
luminosity 4.509202522 solar
table star wmbasic 36000.0 3.5 -0.12
luminosity 4.509202522 solar
c
c commands for density & abundances =========
abundances old solar 84
hden 6.0
grains ism
c
c commands controlling geometry =========
radius -2.0 parsec
sphere
stop zone 1
set dr 0
c
c other commands for details =========
c
c in this model the Lyman lines are pumped significantly
c by the stellar continuum; disable this process to avoid critical
c dependence on the precise shape of the continuum.
no induced processes
constant temperature 4
c
c commands controlling output =========
c
punch overview last "stars_wmbasic.ovr"
punch results last "stars_wmbasic.rlt"
punch transmitted continuum last "stars_wmbasic.trn" no header
c
c stars_wmbasic.in
c class stars
c ========================================
```

This is a test that the code can correctly access the WMBASIC O-star model atmosphere continua.

Checks: table star wmbasic command works.

D.4   The Slow Test Suite

Table D.3 lists the simulations in the `slow` test directory.

Table D.3: The simulations in the slow test suite.

Class	Function
dynamics_HII_PDR	fast H+He+metals (low ionization parameter) 009 km s
feii_blr_n09_p18	FeII model
feii_blr_n09_p18_Z20	FeII model
feii_blr_n11_p20	FeII model
feii_blr_n11_p20_Z20	FeII model
feii_blr_n12_p19	FeII model
feii_blr_n12_p19_Z20	FeII model
feii_blr_n13_p18	BLR model, density 1e13 cm-3, flux of H-ion phots 1e18 cm2 s-1
feii_blr_n13_p18_Z20	BLR model, density 1e13 cm-3, flux of H-ion phots 1e18 cm2 s-1
feii_blr_n13_p22	FeII model
feii_blr_n13_p22_Z20	FeII model
h2_HTT91	Hollenbach et al. 1991 low-density PDR
h2_orion_hii_pdr	constant gas pressure orion into pdr
h2_pdr_leiden_f2	low density high flux model 2 as defined in e-mail
h2_pdr_leiden_f3	high density low flux model 3 as defined in e-mail
h2_pdr_leiden_f4	high density high flux model 4 as defined in e-mail
h2_pdr_leiden_hack_f1	low density low flux model 1 as defined in e-mail
h2_pdr_leiden_v1	model 5 as defined in e-mail
h2_pdr_leiden_v2	model 6 as defined in e-mail
h2_pdr_leiden_v3	model 7 as defined in e-mail
h2_pdr_leiden_v4	model 8 as defined in e-mail
h2_th85orion	Tielens and Hollenbach pdr model with orion grains, Table 2, paper b
m17_P07	Pellegrini et al 2008 model for M17

This HTML file was created by the program doc_tsuite.pl. doc_tsuite.txt contains a tab delimited list of the files.

`dynamics_HII_PDR.in` *fast H+He+metals (low ionization parameter) 009 km s*

```
title fast H+He+metals (low ionization parameter) 009 km s
failures 1
*
* ADVECTION PARAMETERS
*
wind -9 advection no continuum
set dynamics pressure mode subsonic
iterate 10 to convergence 0.001
* iterate 10 to convergence 0.001
set nend 2000
*
* INPUT SPECTRA
*
* hot wd star
blackbody 120,000
phi(h) 9.5
sphere
turbulence 8 km/sec
*
* OTHER PHYS PARAMS
normalize to "Q(H)" 4861
*
hden 3.5
*init "fast.ini"
init "ism.ini"
abundances hii region no grains
grains orion no qheat single
*
* STOPPING CRITERIA
*
stop thickness 18
stop efrac -3
stop temperature 100
no molecules
*
* SPEEDUP
*
atom h-like levels small
atom he-like levels small
* set drmin 12.5
*
* Output
punch dr "dynamics_HII_PDR.dr"
punch overview "dynamics_HII_PDR.ovr"
punch wind "dynamics_HII_PDR.wnd"
*
*
* asserts
c
c dynamics_HII_PDR.in
c
```

---

feii_blr_n09_p18.in *FeII model*

```
title FeII model
table agn
atom feii
print lines column
print lines sort intensity
hden 9
phi(h) 18
stop column density 23
iterate convergence
normalize to "totl" 1216 100
init "c84.ini"
punch feii lines "feii_blr_n09_p18.lin" last
punch feii column densities "feii_blr_n09_p18.fe2col" last
punch feii relative populations all "feii_blr_n09_p18.lev" last
punch feii relative populations range 0 200 "feii_blr_n09_p18.lv1" last
punch feii relative populations range 201 371 "feii_blr_n09_p18.lv2" last
punch overview "feii_blr_n09_p18.ovr" last
punch dr "feii_blr_n09_p18.dr" last
punch convergence reason "feii_blr_n09_p18.cvr"
```

This is one of the 5 models that sample the LOC plane.

feii_blr_n09_p18_Z20.in *FeII model*

```
title FeII model
table agn
print lines column
print lines sort intensity
atom feii
abundances starburst 20
hden 9
phi(h) 18
stop column density 23
iterate convergence
normalize to "totl" 1216 100
init "c84.ini"
punch feii lines "feii_blr_n09_p18_Z20.lin" last
punch feii colum density "feii_blr_n09_p18_Z20.fe2col" last
punch feii populations all "feii_blr_n09_p18_Z20.lev" last
punch overview "feii_blr_n09_p18_Z20.ovr" last
punch dr "feii_blr_n09_p18_Z20.dr" last
punch convergence reason "feii_blr_n09_p18_Z20.cvr"
```

This is one of the 5 models that sample the LOC plane.

feii_blr_n11_p20.in *FeII model*

```
title FeII model
table agn
print lines column
print lines sort intensity
atom feii
```

```
hden 11
phi(h) 20
stop column density 23
iterate convergence
normalize to "totl" 1216 100
init "c84.ini"
punch convergence reason "feii_blr_n11_p20.cvr"
punch feii lines "feii_blr_n11_p20.lin" last
punch feii column densities "feii_blr_n11_p20.fe2col" last
punch feii populations all "feii_blr_n11_p20.lev" last
punch overview "feii_blr_n11_p20.ovr" last
punch dr "feii_blr_n11_p20.dr" last
```

This is one of the 5 models that sample the LOC plane.

`feii_blr_n11_p20_Z20.in` *FeII model*

```
title FeII model
table agn
print lines column
print lines sort intensity
atom feii
failures 2
abundances starburst 20
hden 11
phi(h) 20
stop column density 23
iterate convergence
normalize to "totl" 1216 100
init "c84.ini"
punch cooling "feii_blr_n11_p20_Z20.col"
punch convergence reason "feii_blr_n11_p20_Z20.cvr"
punch feii lines "feii_blr_n11_p20_Z20.lin" last
punch feii column densites "feii_blr_n11_p20_Z20.fe2col" last
punch feii populations all "feii_blr_n11_p20_Z20.lev"
punch heating "feii_blr_n11_p20_Z20.het"
punch element calcium "feii_blr_n11_p20_Z20.ca"
punch overview "feii_blr_n11_p20_Z20.ovr" last
punch dr "feii_blr_n11_p20_Z20.dr"
```

This is one of the 5 models that sample the LOC plane.

`feii_blr_n12_p19.in` *FeII model*

```
title FeII model
table agn
print lines column
print lines sort intensity
atom feii
hden 12
phi(h) 19
stop column density 23
iterate convergence
normalize to "totl" 1216 100
init "c84.ini"
punch convergence reason "feii_blr_n12_p19.cvr"
```

```
punch feii lines "feii_blr_n12_p19.lin" last
punch feii populations all "feii_blr_n12_p19.lev" last
punch feii column densities "feii_blr_n12_p19.fe2col" last
punch overview "feii_blr_n12_p19.ovr" last
punch dr "feii_blr_n12_p19.dr" last
```

This is one of the 5 models that sample the LOC plane.

`feii_blr_n12_p19_Z20.in` *FeII model*

```
title FeII model
table agn
print lines column
print lines sort intensity
atom feii
abundances starburst 20
hden 12
phi(h) 19
stop column density 23
iterate convergence
normalize to "totl" 1216 100
init "c84.ini"
punch feii lines "feii_blr_n12_p19.lin" last
punch feii column densiteis "feii_blr_n12_p19.fe2col" last
punch feii populations all "feii_blr_n12_p19.lev" last
punch convergence reason "feii_blr_n12_p19.cvg"
punch overview "feii_blr_n12_p19.ovr" last
punch dr "feii_blr_n12_p19.dr" last
```

This is one of the 5 models that sample the LOC plane.

`feii_blr_n13_p18.in` *BLR model, density 1e13 cm-3, flux of H-ion phots 1e18 cm2 s-1*

```
title BLR model, density 1e13 cm-3, flux of H-ion phots 1e18 cm2 s-1
print lines column
print lines sort intensity
atom feii
c
c this is a very low ionization cloud
c the conditions, and some lines, are surprisingly sensitive
c to the treatment of hydrogen molecules
c
table agn
hden 13
phi(h) 18
stop column density 22
double
iterate convergence
normalize to "totl" 1216 100
init "c84.ini"
punch feii lines "feii_blr_n13_p18.lin" last
punch feii column densites "feii_blr_n13_p18.fe2col" last
punch feii populations all "feii_blr_n13_p18.lev" last
punch overview "feii_blr_n13_p18.ovr" last
```

```
punch dr "feii_blr_n13_p18.dr" last
punch convergence reason "feii_blr_n13_p18.cvr"
```

This is one of the 5 models that sample the LOC plane.

`feii_blr_n13_p18_Z20.in` *BLR model, density 1e13 cm-3, flux of H-ion phots 1e18 cm2 s-1*

```
title BLR model, density 1e13 cm-3, flux of H-ion phots 1e18 cm2 s-1
print lines column
print lines sort intensity
atom feii
abundances starburst 20
c
c this is a very low ionization cloud
c the conditions, and some lines, are surprisingly sensitive
c to the treatment of hydrogen molecules
c
table agn
hden 13
phi(h) 18
stop column density 19
double
iterate convergence limit 15
normalize to "totl" 1216 100
init "c84.ini"
punch feii lines "feii_blr_n13_p18_Z20.lin" last
punch feii column densities "feii_blr_n13_p18_Z20.fe2col" last
punch feii populations all "feii_blr_n13_p18_Z20.lev" last
punch overview "feii_blr_n13_p18_Z20.ovr" last
punch dr "feii_blr_n13_p18_Z20.dr" last
punch convergence reason "feii_blr_n13_p18_Z20.cvr"
```

This is one of the 5 models that sample the LOC plane.

`feii_blr_n13_p22.in` *FeII model*

```
title FeII model
table agn
print lines column
print lines sort intensity
atom feii
hden 13
phi(h) 22
stop column density 23
iterate convergence
normalize to "totl" 1216 100
init "c84.ini"
punch convergence reason "feii_blr_n13_p22.cvr"
punch feii lines "feii_blr_n13_p22.lin" last
punch feii populations all "feii_blr_n13_p22.lev" last
punch feii column density "feii_blr_n13_p22.col" last
punch overview "feii_blr_n13_p22.ovr" last
punch dr "feii_blr_n13_p22.dr" last
```

This is one of the 5 models that sample the LOC plane.

`feii_blr_n13_p22_Z20.in` *FeII model*

```
title FeII model
table agn
print lines column
print lines sort intensity
atom feii
abundances starburst 20
hden 13
phi(h) 22
stop column density 23
iterate convergence
normalize to "totl" 1216 100
init "c84.ini"
punch convergence reason "feii_blr_n13_p22_Z20.cvr"
punch feii lines "feii_blr_n13_p22_Z20.lin" last
punch feii column densities "feii_blr_n13_p22_Z20.fe2col" last
punch feii populations all "feii_blr_n13_p22_Z20.lev" last
punch overview "feii_blr_n13_p22_Z20.ovr" last
punch dr "feii_blr_n13_p22_Z20.dr"
punch heating "feii_blr_n13_p22_Z20.het"
punch cooling "feii_blr_n13_p22_Z20.col"
```

This is one of the 5 models that sample the LOC plane.

`h2_HTT91.in` *Hollenbach et al. 1991 low-density PDR*

```
title Hollenbach et al. 1991 low-density PDR
iterate
c
c
hden 3
c
c this is hot star continuum
black 30000
intensity 0.2 range 0.4412 to 1 Ryd
c this will remove all ionizing radiation
extinguish 24 0
c
turbulence 1.5 km/sec
c first continuum is FIR hot grain continuum produced in
c unmodeled HII Region
grains orion, abundance log 0.16
grains PAH
atom h2
init file="ism.ini"
abundances he -1.01 c -3.52 n-8 o-3.30 ne-8 mg-5.89
continue si -6.10 s -5.10 cl-7 ar-8 fe -6.60
normalize to 157.6m "C 2"
sphere
case b
cosmic rays, background
c stop when gas is fully neutral
stop efrac -10
c this is to stop at an intended place, since results would be
```

```
c very dependent on details if we stop on temperature
stop thickness 18.954
c stop when gas is cold
stop temperature 10 linear
c add this to mimic unmodelled neutral gas
double optical depths
c
c uv lines are strongly pumped by stellar continuum, break out contribution
print line pump
print line optical depths
print ages
c
punch overview last "h2_HTT91.ovr"
punch dr "h2_HTT91.dr "
punch molecules last "h2_HTT91.mol"
punch results last "h2_HTT91.rlt"
punch continuum units microns last "h2_HTT91.con"
punch heating last "h2_HTT91.het"
punch cooling last "h2_HTT91.col"
punch H2 destruction "h2_HTT91.H2d"
punch H2 creation "h2_HTT91.H2c"
c
c h2_HTT91.in
```

This is the Hollenbach et al 1991 Low-density PDR The case b command appears because the Lyman lines are vastly optically thick in this environment. If the command is removed the Lyman lines will be optically thin, which will result in fast fluorescent excitation of the (nearly totally neutral) hydrogen atom.

there is very little CO in this model since it is not thick enough for the UV pumping lines to become optically thick

---

`h2_orion_hii_pdr.in` *constant gas pressure orion into pdr*

```
title constant gas pressure orion into pdr
stop temperature 10 linear
c
c print lots of faint CO lines
print line faint -4
stop AV 1000 point
constant gas pressure
sphere
c
c the incident continuum is two parts
c star and flux of photons striking it
table read "star_kurucz_39600.dat"
c
Q(H) 49
radius 17.4507
c plus hot brems
brems 6
phi(h) 10
c
c add cosmic rays, which are important at depth
cosmic rays, background
c
c we have a spherical geometry but want to simulate observing
c through a spectrometer's slit. use the aperture
c command for this
aperture beam
c
```

```
c the observed microturbulence
turbulence 8 km/sec no pressure
c
c density and abundances
hden 4
init file="ism.ini"
abundances hii region no grains
grains orion
grains pah
atom h2
c
set nend 2000
c
punch overview last "h2_orion_hii_pdr.ovr"
punch hydrogen 21 cm last "h2_orion_hii_pdr.21cm"
punch heating "h2_orion_hii_pdr.het"
punch cooling "h2_orion_hii_pdr.col"
punch dr last "h2_orion_hii_pdr.dr"
punch results last "h2_orion_hii_pdr.rlt"
punch continuum last "h2_orion_hii_pdr.con" units microns
punch hydrogen lya last "h2_orion_hii_pdr.lya"
punch grain charge last "h2_orion_hii_pdr.grnchr"
punch grain potential last "h2_orion_hii_pdr.grnpot"
punch H2 lines last "h2_orion_hii_pdr.h2lin"
punch H2 column density last "h2_orion_hii_pdr.h2col"
punch H2 destruction "h2_orion_hii_pdr.H2d"
punch H2 creation "h2_orion_hii_pdr.H2c"
punch molecules "h2_orion_hii_pdr.mol"
c
c h2_orion_hii_pdr.in
```

05 dec 19, had stopped at 1 pc, gas/dust got very cold and H2 stopped forming, H2 went to H0 - this below the CO network out of the water. stop at AV of 1000 instead. Is this loss of H2 at low grain temperature physical? Is it ever seen?

`h2_pdr_leiden_f2.in` *low density high flux model 2 as defined in e-mail*

```
title low density high flux model 2 as defined in e-mail
c
c commands controlling continuum =========
c Use the Draine 1978 field, for a semi-infinite slab we have to use
c half the requested value, so the actual value
c they want for the model is actually twice the value below
table draine 50000
c insure that no H-ionizing radiation strikes cloud
extinguish 24
c
c commands for density & abundances =========
grains ism 1.16 no qheat
c hydrogen density
hden 3.
c
c commands controlling geometry =========
c
c other commands for details =========
atom h2
failures 3
c use leiden initialization file
```

```
init file="pdr_leiden.ini"
c This command defines the grain temperature to be a constant 20 Kelvin
constant grain temperature 20
c This sets the temperature to a constant 50 Kelvin
constant temperature 50 linear
c
c commands controlling output =========
punch overview "h2_pdr_leiden_f2.ovr"
punch leiden lines "h2_pdr_leiden_f2.lin"
punch leiden "h2_pdr_leiden_f2.lei"
punch dr "h2_pdr_leiden_f2.dr"
punch grain physical conditions "h2_pdr_leiden_f2.grn"
punch H2 lines "h2_pdr_leiden_f2.h2lin" all
punch H2 column density "h2_pdr_leiden_f2.h2col"
punch H2 populations matrix zone "h2_pdr_leiden_f2.pop"
punch H2 destruction "h2_pdr_leiden_f2.H2d"
punch H2 creation "h2_pdr_leiden_f2.H2c"
c
c
c h2_pdr_leiden_f2.in
c class pdr
c ==========================================
c
```

`h2_pdr_leiden_f3.in` *high density low flux model 3 as defined in e-mail*

```
title high density low flux model 3 as defined in e-mail
c
c commands controlling continuum =========
c Use the Draine 1978 field, for a semi-infinite slab we have to use half
c the requested value, so the actual value
c they want for the model is actually twice the value below
table draine 5
c insure that no H-ionizing radiation strikes cloud
extinguish 24
c
c commands for density & abundances =========
grains ism 1.16 no qheat
c hydrogen density
hden 5.5
c
c commands controlling geometry =========
c
c other commands for details =========
c ice formation is exteme and will establish the thickness, which would
c make this very sensitive to details - turn off ices
no grain molecules
atom h2
failures 3
c use leiden initialization file
init file="pdr_leiden.ini"
c This command defines the grain temperature to be a constant 20 Kelvin
constant grain temperature 20
c This sets the temperature to a constant 50 Kelvin
constant temperature 50 linear
c
c commands controlling output =========
punch overview "h2_pdr_leiden_f3.ovr"
punch leiden lines "h2_pdr_leiden_f3.lin"
```

```
punch leiden "h2_pdr_leiden_f3.lei"
punch dr "h2_pdr_leiden_f3.dr"
punch grain physical conditions "h2_pdr_leiden_f3.grn"
punch H2 lines "h2_pdr_leiden_f3.h2lin" all
punch H2 column density "h2_pdr_leiden_f3.h2col"
punch H2 populations matrix zone "h2_pdr_leiden_f3.pop"
punch H2 rates "h2_pdr_leiden_f3.rat"
punch H2 destruction "h2_pdr_leiden_f3.H2d"
punch H2 creation "h2_pdr_leiden_f3.H2c"
c
c
c h2_pdr_leiden_f3.in
c class pdr
c =========================================
c
```

`h2_pdr_leiden_f4.in` *high density high flux model 4 as defined in e-mail*

```
title high density high flux model 4 as defined in e-mail
c
c commands controlling continuum =========
c
c commands for density & abundances =========
grains ism 1.16 no qheat
c
c commands controlling geometry =========
c Use the Draine 1978 field, for a semi-infinite slab we have to use half
c the requested value, so the actual value
c they want for the model is actually twice the value below
table draine 50000
c insure that no H-ionizing radiation strikes cloud
extinguish 24
c
c hydrogen density
hden 5.5
c
c other commands for details =========
atom h2
failures 3
c use leiden initialization file
init file="pdr_leiden.ini"
c This command defines the grain temperature to be a constant 20 Kelvin
constant grain temperature 20
c This sets the temperature to a constant 50 Kelvin
constant temperature 50 linear
c
c commands controlling output =========
punch overview "h2_pdr_leiden_f4.ovr"
punch leiden lines "h2_pdr_leiden_f4.lin"
punch leiden "h2_pdr_leiden_f4.lei"
punch dr "h2_pdr_leiden_f4.dr"
punch grain physical conditions "h2_pdr_leiden_f4.grn"
punch H2 lines "h2_pdr_leiden_f4.h2lin" all
punch H2 column density "h2_pdr_leiden_f4.h2col"
punch H2 populations matrix zone "h2_pdr_leiden_f4.pop"
punch H2 solomon "h2_pdr_leiden_f4.sol"
punch H2 destruction "h2_pdr_leiden_f4.H2d"
punch H2 creation "h2_pdr_leiden_f4.H2c"
c
```

```
c
c h2_pdr_leiden_f4.in
c class pdr
c =======================================
c
```

`h2_pdr_leiden_hack_f1.in` *low density low flux model 1 as defined in e-mail*

```
title low density low flux model 1 as defined in e-mail
atom h2
c
c commands controlling continuum =========
c Use the Draine 1978 field, for a semi-infinite slab we have to use half
c the requested value, so the actual value
c they want for the model is actually twice the value below
table draine 5
c insure that no H-ionizing radiation strikes cloud
extinguish 24
c
c commands for density & abundances =========
c hydrogen density
hden 3.
grains ism 1.16
c
c commands controlling geometry =========
c
c other commands for details =========
failures 3
c use leiden initialization file
init file="pdr_leiden_hack.ini"
c This command defines the grain temperature to be a constant 20 Kelvin
constant grain temperature 20
c This sets the temperature to a constant 50 Kelvin
constant temperature 50 linear
c
c commands controlling output =========
punch leiden lines "h2_pdr_leiden_hack_f1.lin"
punch leiden "h2_pdr_leiden_hack_f1.lei"
punch dr "h2_pdr_leiden_hack_f1.dr"
punch grain physical conditions "h2_pdr_leiden_hack_f1.grn"
punch overview "h2_pdr_leiden_hack_f1.ovr"
c
punch H2 lines "h2_pdr_leiden_hack_f1.h2lin"
punch H2 column density "h2_pdr_leiden_hack_f1.h2col"
punch H2 populations matrix zone "h2_pdr_leiden_hack_f1.pop"
c
c h2_pdr_leiden_hack_f1.in
c class pdr
c =======================================
c
```

`h2_pdr_leiden_v1.in` *model 5 as defined in e-mail*

```
title model 5 as defined in e-mail
c
```

```
c commands controlling continuum =========
c Use the Draine 1978 field, for a semi-infinite slab we have to use half
c the requested value, so the actual value
c is half the requested value
c they want for the model is actually twice the value below
table draine 5
c insure that no H-ionizing radiation strikes cloud
extinguish 24
c
c commands for density & abundances =========
c hydrogen density
hden 3.
grains ism 1.16 no qheat
c add PAHs
grains PAH no qheat 3 function
atom h2
c
c commands controlling geometry =========
c
c other commands for details =========
failures 3
c use leiden initialization file
init file="pdr_leiden.ini"
c
c commands controlling output =========
punch overview "h2_pdr_leiden_v1.ovr"
punch leiden lines "h2_pdr_leiden_v1.lin"
punch leiden "h2_pdr_leiden_v1.lei"
punch dr "h2_pdr_leiden_v1.dr"
punch grain dust temperature "h2_pdr_leiden_v1.grn"
punch H2 destruction "h2_pdr_leiden_v1.H2d"
punch H2 creation "h2_pdr_leiden_v1.H2c"
c
c
c
c
c h2_pdr_leiden_v1.in
c class pdr
c ==========================================
c
```

h2_pdr_leiden_v2.in *model 6 as defined in e-mail*

```
title model 6 as defined in e-mail
c
c commands controlling continuum =========
c Use the Draine 1978 field, for a semi-infinite slab we have to use half
c the requested value, so the actual value
c they want for the model is actually twice the value below
table draine 50000
c insure that no H-ionizing radiation strikes cloud
extinguish 24
c
c commands for density & abundances =========
c add PAHs and grains
grains PAH no qheat 3 function
grains ism 1.16 no qheat
c hydrogen density
hden 3.
c
c commands controlling geometry =========
```

```
c
c other commands for details =========
atom h2
failures 3
c use leiden initialization file
init file="pdr_leiden.ini"
c
c commands controlling output =========
punch overview "h2_pdr_leiden_v2.ovr"
punch leiden lines "h2_pdr_leiden_v2.lin"
punch leiden "h2_pdr_leiden_v2.lei"
punch dr "h2_pdr_leiden_v2.dr"
punch grain dust temperature "h2_pdr_leiden_v2.grn"
punch H2 destruction "h2_pdr_leiden_v2.H2d"
punch H2 creation "h2_pdr_leiden_v2.H2c"
c
c
c h2_pdr_leiden_v2.in
c class pdr
c =========================================
c
```

---

h2_pdr_leiden_v3.in *model 7 as defined in e-mail*

```
title model 7 as defined in e-mail
c
c commands controlling continuum =========
c Use the Draine 1978 field, for a semi-infinite slab we have to use half
c the requested value, so the actual value
c they want for the model is actually twice the value below
table draine 5
c insure that no H-ionizing radiation strikes cloud
extinguish 24
c
c commands for density & abundances =========
c add PAHs and grains
grains PAH no qheat 3 function
grains ism 1.16 no qheat
c hydrogen density
hden 5.5
c
c commands controlling geometry =========
c
c other commands for details =========
atom h2
failures 3
c use leiden initialization file
init file="pdr_leiden.ini"
c
c commands controlling output =========
punch overview "h2_pdr_leiden_v3.ovr"
punch leiden lines "h2_pdr_leiden_v3.lin"
punch leiden "h2_pdr_leiden_v3.lei"
punch dr "h2_pdr_leiden_v3.dr"
punch grain dust temperature "h2_pdr_leiden_v3.grn"
punch heating "h2_pdr_leiden_v3.het"
punch cooling "h2_pdr_leiden_v3.col"
punch molecules "h2_pdr_leiden_v3.mol"
punch H2 destruction "h2_pdr_leiden_v3.H2d"
punch H2 creation "h2_pdr_leiden_v3.H2c"
c
```

```
c
c h2_pdr_leiden_v3.in
c class pdr
c ========================================
c
```

`h2_pdr_leiden_v4.in` *model 8 as defined in e-mail*

```
title model 8 as defined in e-mail
c
c commands controlling continuum =========
c Use the Draine 1978 field, for a semi-infinite slab we have to use half
c the requested value, so the actual value
c they want for the model is actually twice the value below
table draine 50000
c insure that no H-ionizing radiation strikes cloud
extinguish 24
c
c commands for density & abundances =========
grains PAH no qheat 3 function
grains ism 1.16 no qheat
c hydrogen density
hden 5.5
c
c commands controlling geometry =========
c
c other commands for details =========
failures 3
c use leiden initialization file
init file="pdr_leiden.ini"
atom h2
c
c commands controlling output =========
punch overview "h2_pdr_leiden_v4.ovr"
punch leiden lines "h2_pdr_leiden_v4.lin"
punch leiden "h2_pdr_leiden_v4.lei"
punch dr "h2_pdr_leiden_v4.dr"
punch grain dust temperature "h2_pdr_leiden_v4.grn"
punch H2 destruction "h2_pdr_leiden_v4.H2d"
punch H2 creation "h2_pdr_leiden_v4.H2c"
c
c
c h2_pdr_leiden_v4.in
c class pdr
c ========================================
c
```

`h2_th85orion.in` *Tielens and Hollenbach pdr model with orion grains, Table 2, paper b*

```
title Tielens and Hollenbach pdr model with orion grains, Table 2, paper b
failures 3
c
print line sort intensity
print line column
iterate
```

```
c
c cosmic background
background
cosmic rays, background
c
set nend 2000
c simulate effects of gas we do not model
double
c
c first continuum is FIR hot grain continuum produced in
c unmodeled HII Region
blackbody, t=75K
intensity 2.7 (total)
c
c this is the attenuated hot stellar continuum
black 30000
intensity 2.204 range 0.4412 to 1 Ryd
c
c this will remove all ionizing radiation
extinguish 24 0
c
hden 5.362
grains orion
grains PAH
atom h2
init file="ism.ini"
abundances he -1.01 c -3.52 n-8 o-3.30 ne-8 mg-5.89
continue si -6.10 s -5.10 cl-7 ar-8 fe -6.60
turbulence 2.7 km/sec
normalize to "C 2" 157.6m
sphere
case b
cosmic rays, background
c this should be the one actually used
stop AV 33.2
c stop when gas is fully neutral
stop efrac -10
c stop when gas is cold
stop temperature 10 linear
c stop at thickness so that would make results very
c detail dependent
stop thickness 19
c uv lines are strongly pumped by stellar continuum, break out contribution
print line pump
print line optical depths
print ages
punch overview last "h2_th85orion.ovr" no hash
punch pdr last "h2_th85orion.pdr"
punch molecules last "h2_th85orion.mol"
punch dr "h2_th85orion.dr"
punch hydrogen physical conditions last "h2_th85orion.hyd"
punch results last "h2_th85orion.rlt"
punch continuum units microns last "h2_th85orion.con"
punch fine conti unit last micron "h2_th85orion_con.fin" range 0.7 1. skip 2
* punch fine opaci unit micron "h2_th85orion_opc.fin" range 0.7 1. skip 2
punch heating last "h2_th85orion.het"
punch cooling last "h2_th85orion.col"
punch hydrogen 21 cm last "h2_th85orion.21cm"
punch H2 destruction "h2_th85orion.H2d"
punch H2 creation "h2_th85orion.H2c"
c
c
c h2_th85orion.in
```

This is the Tielens and Hollenbach (1985a, b) standard model of the Orion photodissociation region (PDR).

Specifically, this is my attempt at their Table 2 of paper 2, ApJ 291, p749. The case b command appears because the Lyman lines are vastly optically thick in this environment. If the command is removed the Lyman lines will be optically thin, which will result in fast fluorescent excitation of the (nearly totally neutral) hydrogen atom.

// >>refer model pdr Tielens, A. G. G. M., & Hollenbach, D. 1985a, ApJ, 291, 722 // >>refer model pdr Tielens, A. G. G. M., & Hollenbach, D. 1985b, ApJ, 291, 746

`m17_P07.in` *Pellegrini et al 2008 model for M17*

```
title Pellegrini et al 2008 model for M17
c a brief description of its purpose
c
c commands controlling incident radiation field =========
c these give the shape and intensity
brems 8,500,000K
luminosity 33.5 range 7.35 to 176 Ryd
table star CoStar, 47,490K
Q(h) 49.63
table star CoStar, 45,335K
Q(h) 49.47
table star CoStar, 43,151K
Q(h) 49.30
table star CoStar, 41,209K
Q(h) 49.18
table star CoStar, 39,084K
Q(h) 49.08
table star CoStar, 36,982K
Q(h) 48.81
table star atlas, 34,914K
Q(h) 48.97
table star atlas,27,700K
absolute bolometric magnitude -8.5
table star atlas, 23,700K
absolute bolometric magnitude -7.5
table star atlas, 20,350K
absolute bolometric magnitude -6.6
table star atlas, 17,050
absolute bolometric magnitude -6.3
c
c commands for density & abundances =========
hden 2.065
abundances H II region no qheat no grains
grains PAH 10
grains orion 1
element Helium abundance -0.98
element Carbon abundance -4.14
element Nitrogen abundance -4.2
element Oxygen abundance -3.47
element Neon abundance -4.4
element Sulphur abundance -5.06
element Chlorine abundance -6.88
element Argon abundance -5.64
element Iron abundance -5.5
c these are to speed things up a bit
element limit off -6.9
c
c commands controlling geometry =========
radius 0.35355 linear parsecs
```

```
c the observed extinction along this sight line
stop Av extended 6.36
c do not want to stop at too small a depth due to low temperature
stop temperature off
c magnetostatic equilibrium
constant pressure
sphere
c
c other commands for details =========
magnetic field tangled -4.3125 2
turbulence 3km/s
iterate
cosmic rays background 2.5
c
c commands controlling output =========
punch lines emissivity last "m17_P07.ems"
H 1 6563
S 2 6720
12CO 1294m
12CO 2589m
12CO 863.0m
12CO 369.8m
C 2 157.6m
C 1 369.7m
C 1 609.2m
O 1 63.17m
O 1 145.5m
Si 2 34.81m
end of line
print line sort wavelength
punch overview last "m17_p07.ovr"
punch hydrogen 21 CM last "m17_p07.h21"
punch pressure last "m17_p07.pre"
punch molecules last "m17_p07.mol"
punch heating last "m17_p07.het"
punch cooling last "m17_p07.col"
punch continuum last "m17_p07.con" units microns
c
c
c m17_P07.in
c class hii pdr
c =======================================
c
```

This is the magnetostatic model of the M17 H+ / H0 / H2 region presented in Pellegrini et al. 2007, ApJ, 658, 1119

D.5 Sample Programs

REFERENCES

E. H. Avrett and R. Loeser. Kernel Representations in the Solution of Line-Transfer Problems. *SAO Special Report*, 201, February 1966.

E. H. Avrett and R. Loeser. Radiative transfer in the broad emission line regions of quasi-stellar objects. ApJ, 331:211–246, August 1988. doi: 10.1086/166550.

G. Baessgen, M. Baessgen, and M. Grewing. Comparison of different radiation transfer treatments in planetary nebulae. A&A, 200:51–57, July 1988.

J. Baldwin, G. Ferland, K. Korista, and D. Verner. Locally Optimally Emitting Clouds and the Origin of Quasar Emission Lines. ApJ, 455:L119+, December 1995a. doi: 10.1086/309827.

J. Baldwin, G. Ferland, K. Korista, and D. Verner. Locally Optimally Emitting Clouds and the Origin of Quasar Emission Lines. In ApJ Baldwin et al. (1995a), pages L119+. doi: 10.1086/309827.

J. Baldwin, G. Ferland, K. Korista, and D. Verner. Locally Optimally Emitting Clouds and the Origin of Quasar Emission Lines. In ApJ Baldwin et al. (1995a), pages L119+. doi: 10.1086/309827.

R. P. Bauman, R. L. Porter, G. J. Ferland, and K. B. MacAdam. J-Resolved He I Emission Predictions in the Low-Density Limit. ApJ, 628:541–554, July 2005. doi: 10.1086/430665.

F. Bertoldi and B. T. Draine. Nonequilibrium Photodissociation Regions: Ionization-Dissociation Fronts. ApJ, 458:222, February 1996. doi: 10.1086/176805.

H. Bethe. Zur Theorie des Durchgangs schneller Korpuskularstrahlen durch Materie. *Annalen der Physik*, 397:325–400, 1930. doi: 10.1002/andp.19303970303.

J. R. M. Bonilha, R. Ferch, E. E. Salpeter, G. Slater, and P. D. Noerdlinger. Monte Carlo calculations for resonance scattering with absorption or differential expansion. ApJ, 233:649–660, October 1979. doi: 10.1086/157426.

R. N. Boyd and G. J. Ferland. Photoerosion of nuclei in quasar emission-line regions. ApJ, 318:L21–L24, July 1987. doi: 10.1086/184929.

E. R. Cohen and B. N. Taylor. The 1986 adjustment of the fundamental physical constants. *Reviews of Modern Physics*, 59:1121–1148, 1987. doi: 10.1103/RevModPhys.59.1121.

A. Dalgarno and R. A. McCray. Heating and Ionization of HI Regions. ARA&A, 10:375, 1972. doi: 10.1146/annurev.aa.10.090172.002111.

K. Davidson. On photoionization analyses of emission spectra of quasars. ApJ, 218:20–32, November 1977. doi: 10.1086/155653.

M. Elitzur. On hydrogen line emission from regions exposed to strong radiation fields. ApJ, 280:653–660, May 1984. doi: 10.1086/162038.

M. Elitzur, editor. *Astronomical masers*, volume 170 of *Astrophysics and Space Science Library*, 1992.

M. Elitzur, G. J. Ferland, S. G. Mathews, and G. A. Shields. Stimulated emission and the flat Balmer decrements of cataclysmic variable stars. ApJ, 272:L55–L59, September 1983. doi: 10.1086/184116.

J. W. Ferguson and G. J. Ferland. Accurate Hydrogen Spectral Simulations with a Compact Model Atom. ApJ, 479:363, April 1997. doi: 10.1086/303880.

G. Ferland. The Lexington Benchmarks for Numerical Simulations of Nebulae. In R. Williams and M. Livio, editors, *The Analysis of Emission Lines: A Meeting in Honor of the 70th Birthdays of D. E. Osterbrock & M. J. Seaton, proceedings of the Space Telescope Science Institute Symposium, held in Baltimore, Maryland May 16–18, 1994*, page 83. Cambridge: Cambridge University Press, 1995.

G. Ferland. Reliability in the Face of Complexity - the Challenge of High-end Scientific Computing. In G. Ferland and D. W. Savin, editors, *Spectroscopic Challenges of Photoionized Plasmas*, volume 247 of *Astronomical Society of the Pacific Conference Series*, page 391, 2001.

G. Ferland. Quasars and Starbursts in Young Giant Elliptical Galaxies. In E. Perez, R. M. Gonzalez Delgado, and G. Tenorio-Tagle, editors, *Star Formation Through Time*, volume 297 of *Astronomical Society of the Pacific Conference Series*, page 69, 2003.

G. J. Ferland. On the rapid emission-line variations in IC 4997. MNRAS, 188:669–673, September 1979.

G. J. Ferland. N III line emission in planetary nebulae - Continuum fluorescence. ApJ, 389:L63–L65, April 1992. doi: 10.1086/186349.

G. J. Ferland and M. J. Rees. Radiative equilibrium of high-density clouds with application to active galactic nucleus continua. ApJ, 332:141–156, September 1988. doi: 10.1086/166639.

G. J. Ferland, A. C. Fabian, and R. M. Johnstone. The physical conditions within dense cold clouds in cooling flows - II. MNRAS, 333:876–884, July 2002. doi: 10.1046/j.1365-8711.2002.05470.x.

G. J. Ferland, A. C. Fabian, N. A. Hatch, R. M. Johnstone, R. L. Porter, P. A. M. van Hoof, and R. J. R. Williams. Collisional heating as the origin of filament emission in galaxy clusters. MNRAS, 392: 1475–1502, February 2009. doi: 10.1111/j.1365-2966.2008.14153.x.

G. J. Ferland, R. L. Porter, P. A. M. van Hoof, R. J. R. Williams, N. P. Abel, M. L. Lykins, G. Shaw, W. J. Henney, and P. C. Stancil. The 2013 Release of Cloudy. *Revista Mexicana de Astronomia y Astrofisica*, 49:137–163, April 2013.

G. B. Field. Thermal Instability. ApJ, 142:531, August 1965. doi: 10.1086/148317.

H. Friedrich. *Theoretical Atomic Physics*. Berlin: Springer, 1998.

T. J. Gaetz and E. E. Salpeter. Line radiation from a hot, optically thin plasma - Collision strengths and emissivities. ApJS, 52:155–168, June 1983. doi: 10.1086/190862.

M. Gavrila. Elastic Scattering of Photons by a Hydrogen Atom. *Physical Review*, 163:147–155, November 1967. doi: 10.1103/PhysRev.163.147.

R. C. Hilborn. Einstein coefficients, cross sections, f values, dipole moments, and all that. *American Journal of Physics*, 50:982–986, November 1982. doi: 10.1119/1.12937.

D. G. Hummer. A fast and accurate method for evaluating the nonrelativistic free-free Gaunt factor for hydrogenic ions. ApJ, 327:477–484, April 1988. doi: 10.1086/166210.

D. G. Hummer and P. B. Kunasz. Energy loss by resonance line photons in an absorbing medium. ApJ, 236:609–618, March 1980. doi: 10.1086/157779.

D. G. Hummer and G. B. Rybicki. Second-order escape probability approximations in radiative transfer. ApJ, 263:925–934, December 1982. doi: 10.1086/160560.

D. G. Hummer and P. J. Storey. Recombination-line intensities for hydrogenic ions. I - Case B calculations for H I and He II. MNRAS, 224:801–820, February 1987.

B. M. Johnson and E. Quataert. The Effects of Thermal Conduction on Radiatively Inefficient Accretion Flows. ApJ, 660:1273–1281, May 2007. doi: 10.1086/513065.

J. S. Kaastra and R. Mewe. X-ray emission from thin plasmas. I - Multiple Auger ionisation and fluorescence processes for Be to Zn. A&AS, 97:443–482, January 1993.

Kalkofen, W., editor. *Numerical radiative transfer.* . Cambridge: University Press, 1987.

W. J. Karzas and R. Latter. Electron Radiative Transitions in a Coulomb Field. ApJS, 6:167, May 1961. doi: 10.1086/190063.

J. B. Kingdon and G. J. Ferland. Temperature Fluctuations in Photoionized Nebulae. ApJ, 450:691, September 1995. doi: 10.1086/176175.

J. B. Kingdon and G. J. Ferland. Rate Coefficients for Charge Transfer between Hydrogen and the First 30 Elements. ApJS, 106:205, September 1996. doi: 10.1086/192335.

J. B. Kingdon and G. J. Ferland. The Effects of Charge Transfer on the Thermal Equilibrium of Photoionized Nebulae. ApJ, 516:L107–L109, May 1999. doi: 10.1086/312008.

J. H. Krolik, C. F. McKee, and C. B. Tarter. Two-phase models of quasar emission line regions. ApJ, 249: 422–442, October 1981. doi: 10.1086/159303.

R. L. Kurucz. Atlas: a Computer Program for Calculating Model Stellar Atmospheres. *SAO Special Report*, 309, February 1970.

A. C. Lanzafame, J. A. Tully, K. A. Berrington, P. L. Dufton, P. B. Byrne, and A. Burgess. Collision Strengths and Rate Coefficients for Electron Impact Excitation in Hei - an Extrapolation of R-Matrix Calculations to Higher Electron Impact Energies. MNRAS, 264:402, September 1993.

D. J. Lennon and V. M. Burke. New low-temperature effective collision strengths for transitions among the 1s2 2s2 2p2 terms of NE V. MNRAS, 251:628–631, August 1991.

X.-W. Liu, P. J. Storey, M. J. Barlow, and R. E. S. Clegg. The rich O II recombination spectrum of the planetary nebula NGC 7009: new observations and atomic data. MNRAS, 272:369–388, January 1995.

D. Mihalas. *Stellar atmospheres /2nd edition/.* San Francisco, W. H. Freeman and Co., 1978. 650 p.

P.J. Mohr and B.N. Taylor. CODATA recommended values of the fundamental physical constants: 1998. *Reviews of Modern Physics*, 72:351, 2000. doi: 10.1103/RevModPhys.72.351.

H. Netzer and G. J. Ferland. Some comments on models of photoionized nebulae. PASP, 96:593–597, August 1984. doi: 10.1086/131387.

E. Novotny. *Introduction to stellar atmospheres and interiors.* New York: Oxford University Press, 1973.

H. Nussbaumer and P. J. Storey. Dielectronic recombination at low temperatures. II Recombination coefficients for lines of C, N, O. A&AS, 56:293–312, June 1984.

E. Oliva, A. Pasquali, and M. Reconditi. On [NeV] atomic parameters and physical conditions in NGC 6302. A&A, 305:L21+, January 1996.

M. Peimbert. Temperature Determinations of H II Regions. ApJ, 150:825, December 1967. doi: 10.1086/149385.

D. Pequignot, P. Petitjean, and C. Boisson. Total and effective radiative recombination coefficients. A&A, 251:680–688, November 1991.

D. Péquignot, G. Ferland, H. Netzer, T. Kallman, D. R. Ballantyne, A.-M. Dumont, B. Ercolano, P. Harrington, S. Kraemer, C. Morisset, S. Nayakshin, R. H. Rubin, and R. Sutherland. Photoionization Model Nebulae. In G. Ferland and D. W. Savin, editors, *Spectroscopic Challenges of Photoionized Plasmas*, volume 247 of *Astronomical Society of the Pacific Conference Series*, page 533, 2001.

R. L. Porter and G. J. Ferland. Revisiting He-like X-Ray Emission-Line Plasma Diagnostics. ApJ, 664: 586–595, July 2007. doi: 10.1086/518882.

R. L. Porter, R. P. Bauman, G. J. Ferland, and K. B. MacAdam. Theoretical He I Emissivities in the Case B Approximation. ApJ, 622:L73–L75, March 2005. doi: 10.1086/429370.

R. C. Puetter. Theoretical quasar emission line ratios. IV - General asymptotic escape probabilities and the effects of linear Stark broadening. ApJ, 251:446–450, December 1981. doi: 10.1086/159482.

D. W. Savin. Ionization Balance, Chemical Abundances, and the Metagalactic Radiation Field at High Redshift. ApJ, 533:106–112, April 2000. doi: 10.1086/308670.

D. W. Savin, S. M. Kahn, J. Linkemann, A. A. Saghiri, M. Schmitt, M. Grieser, R. Repnow, D. Schwalm, A. Wolf, T. Bartsch, C. Brandau, A. Hoffknecht, A. Müller, S. Schippers, M. H. Chen, and N. R. Badnell. Dielectronic Recombination in Photoionized Gas. II. Laboratory Measurements for Fe XVIII and Fe XIX. ApJS, 123:687–702, August 1999. doi: 10.1086/313247.

M. Seaton. Atomic data for opacity calculations: I. General description. *Journal of Physics B Atomic Molecular Physics*, 20:6363–6378, 1987. doi: 10.1088/0022-3700/20/23/026.

L. Spitzer. *Physics of Fully Ionized Gases (2nd edition)*. New York: Interscience, 1962.

P. J. Storey and D. G. Hummer. Recombination line intensities for hydrogenic ions-IV. Total recombination coefficients and machine-readable tables for Z=1 to 8. MNRAS, 272:41–48, January 1995. on the web at http://adc.gsfc.nasa.gov/adc-cgi/cat.pl?/catalogs/6/6064/.

C. B. Tarter. *Radiative Transfer in a Gas Excited by X-Rays*. PhD thesis, AA(CORNELL UNIVERSITY.), 1967.

C. B. Tarter and C. F. McKee. The Effects of Ionizing Radiation Pressure in Quasars and X-Ray Sources. ApJ, 186:L63+, December 1973. doi: 10.1086/181358.

A. G. G. M. Tielens and D. Hollenbach. Photodissociation regions. I - Basic model. II - A model for the Orion photodissociation region. ApJ, 291:722–754, April 1985. doi: 10.1086/163111.

D. van Blerkom and D. G. Hummer. The ionization structure of planetary nebulae-VI. The Lyman continuum problem. MNRAS, 137:353, 1967.

P. A. M. van Hoof, D. A. Beintema, D. A. Verner, and G. J. Ferland. The collision strength of the [Ne v] infrared fine-structure lines. A&A, 354:L41–L44, February 2000.

H. van Regemorter. Rate of Collisional Excitation in Stellar Atmospheres. ApJ, 136:906, November 1962. doi: 10.1086/147445.

D. A. Verner and G. J. Ferland. Atomic Data for Astrophysics. I. Radiative Recombination Rates for H-like, He-like, Li-like, and Na-like Ions over a Broad Range of Temperature. ApJS, 103:467, April 1996. doi: 10.1086/192284.

D. A. Verner, G. J. Ferland, K. T. Korista, and D. G. Yakovlev. Atomic Data for Astrophysics. II. New Analytic FITS for Photoionization Cross Sections of Atoms and Ions. ApJ, 465:487, July 1996. doi: 10.1086/177435.

R. E. Williams. The Ionization and Thermal Equilibrium of a Gas Excited by Ultraviolet Synchrotron Radiation. ApJ, 147:556, February 1967. doi: 10.1086/149035.