PROGRAM MANUAL
FOR THE
NON-LTE PROGRAM SYSTEM

History:

NOV 1988	S. STUERENBURG
MAR-23-1995	I. RENTZSCH-HOLM
MAR-28-1995	I. RENTZSCH-HOLM
NOV-18-1999	I. KAMP
JUL-6-2004	I. KAMP
MAR-31-2007	I. KAMP

Contents

1 PART I

1.1 Preface

The first part of the NLTE program package comprises the program system ATMOS, which calculates a model atmosphere. The ATMOS program system consists of two programs, TTAU and ATM. The program needs the input files PTAU and ABU and creates the output datafile ATM and a line printer output. This program system has its own description. Therefore, this manual contains only a detailed description of those subroutines, that are explicitly used in the NLTE program package.

It is thus recommended to first read the manual of the ATMOS program system!

The second part of the NLTE program package consists of the programs RATES, START, EDDFAC, GAB and LINEAR with their respective subroutines (see Appendix A).

The NLTE calculations in Kiel have initially been performed on the CRAY and later on Solaris Workstations and a NEC computer at the Kiel super computer center. The current version of the code does not run with a GNU compiler. The compilation needs the -r8 option, which is not yet implemented in the f77 or f90 GNU compiler. It can be compiled with commercial compilers such as Absoft, Intel and Portland Group on Linux or Mac systems.

1.2 Basics

First, the aim of the program package will be explained. By prescribing the temperature stratification of the star in the input datafile PTAU and the chemical composition in the input datafile ABU, the program system ATMOS calculates a detailed model atmosphere. The resulting model atmosphere structure is stored in the datafile ATM.

A model atom is now "put" into this model atmosphere by reading the respective atomic data from the input datafiles ATOM and CROSS. The second part of the NLTE program package now calculates under the assumption of statistical equilibrium (non-LTE) the level population numbers of the atomic levels, which were read from the input datafile ATOM. These are compared to the respective LTE values (derived from Boltzmann and Saha equation) and the departure coefficients

$$b = n(\mathrm{NLTE}) / n(\mathrm{LTE})$$

are derived. The more these coefficients deviate from LTE, that is from one, the worse is the assumption of LTE for the considered element in the stellar atmosphere. The

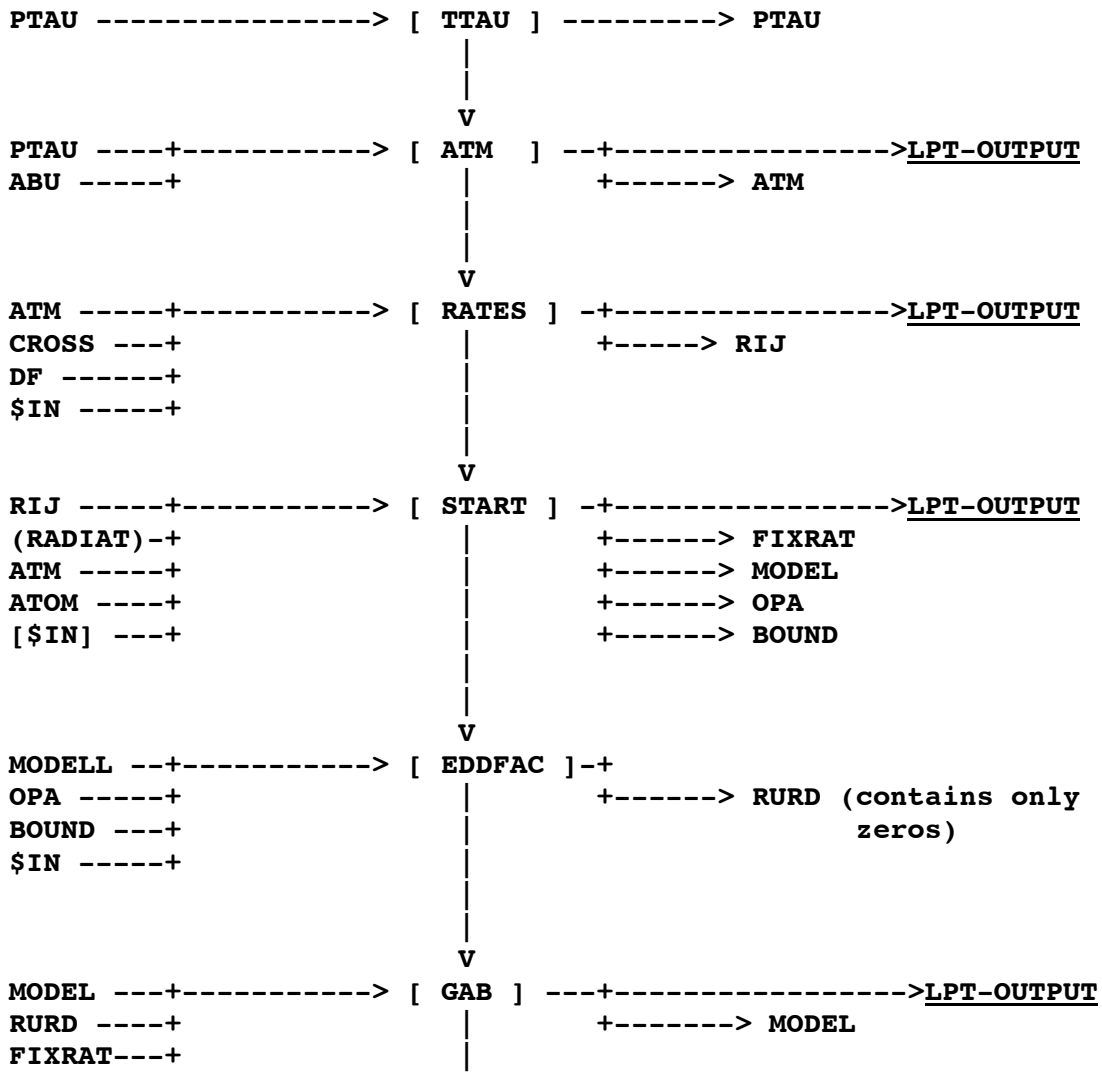
departure coefficients are stored in the output datafile DEPART. By using control indices, it is possible to obtain more information in the output datafile.

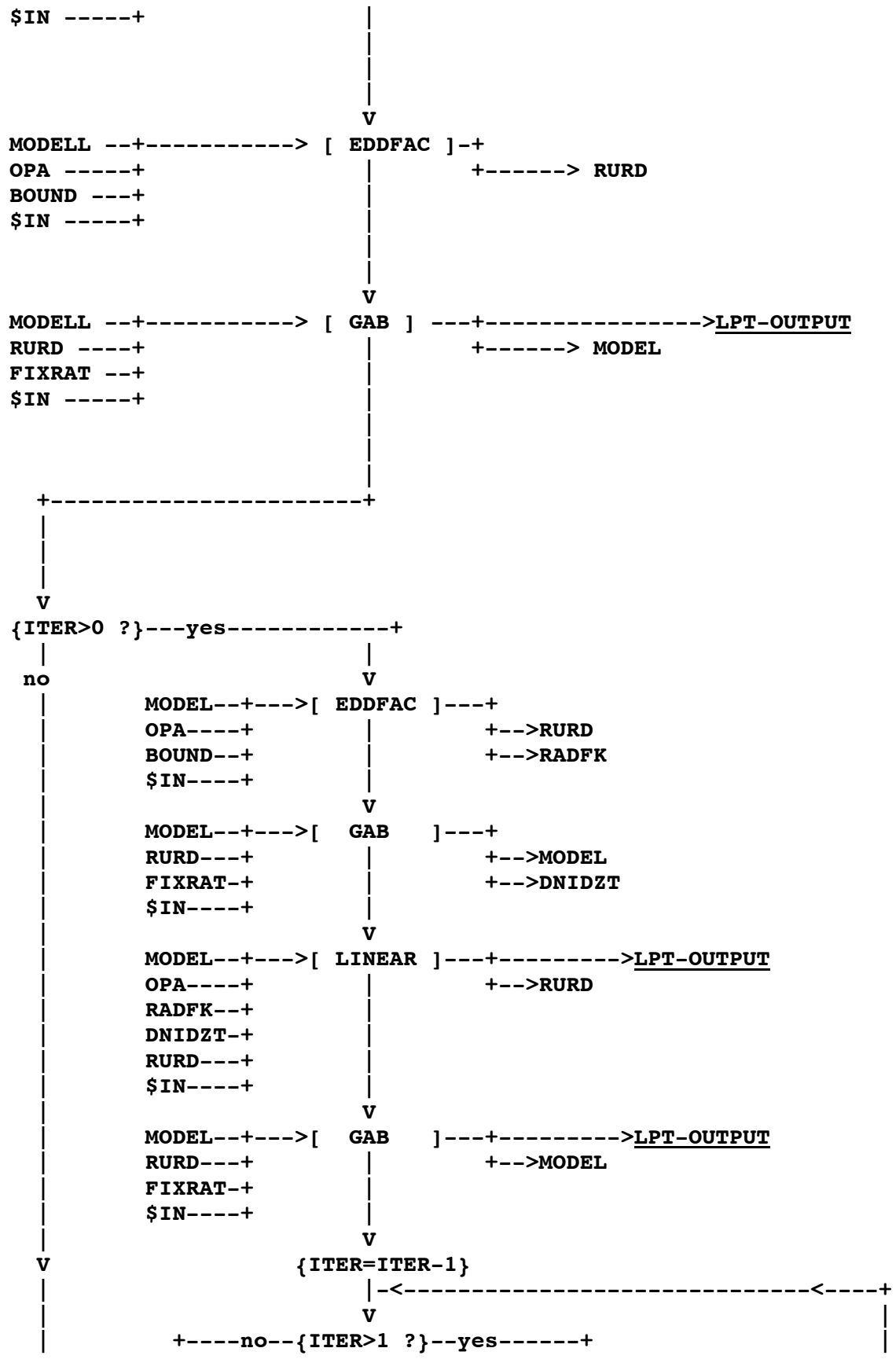
1.3 Program Flow

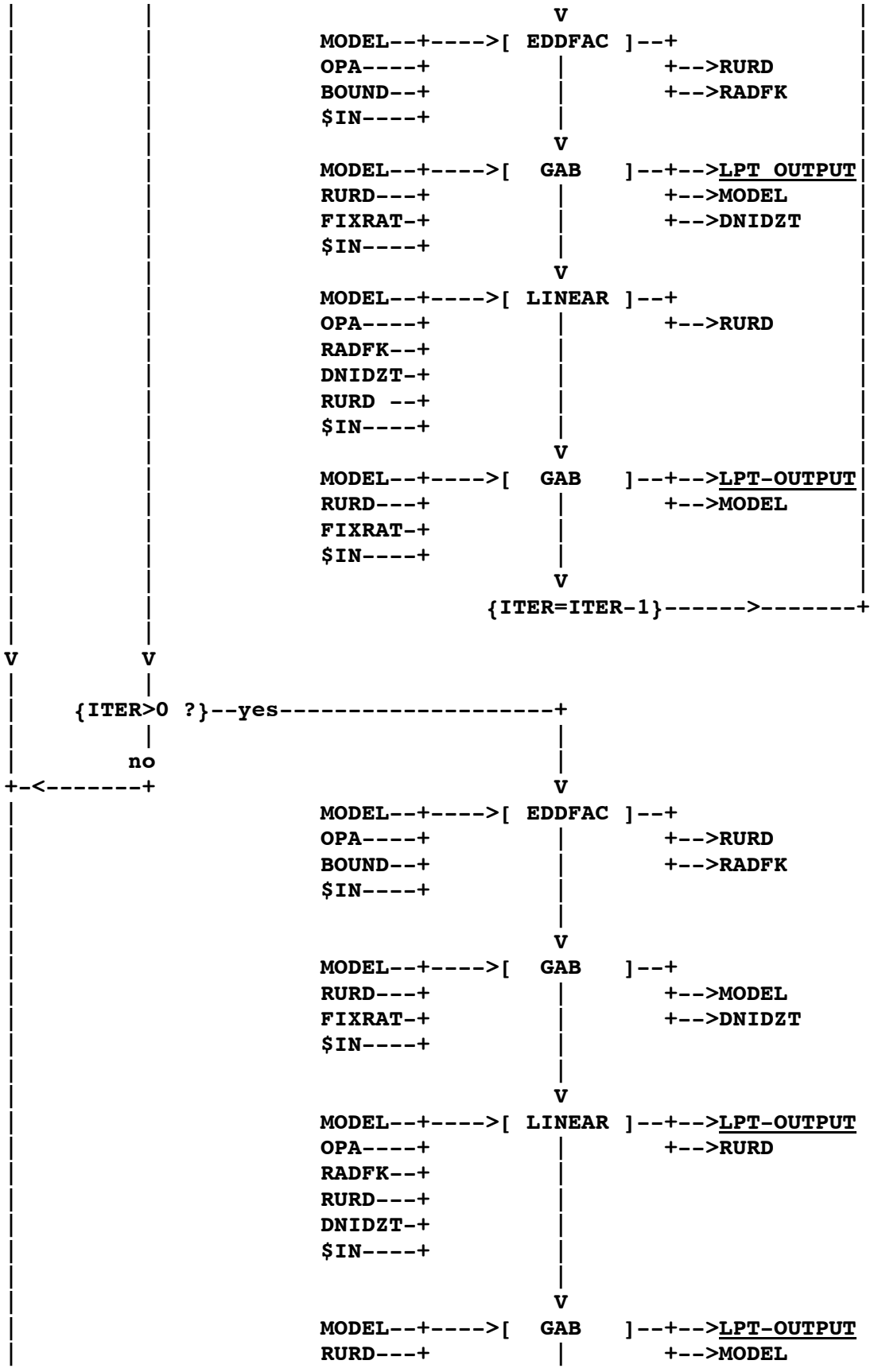
The following gives an overview of the program flow of the entire NLTE program package. It contains all input and output datafiles. The names of programs are denoted by [ ].

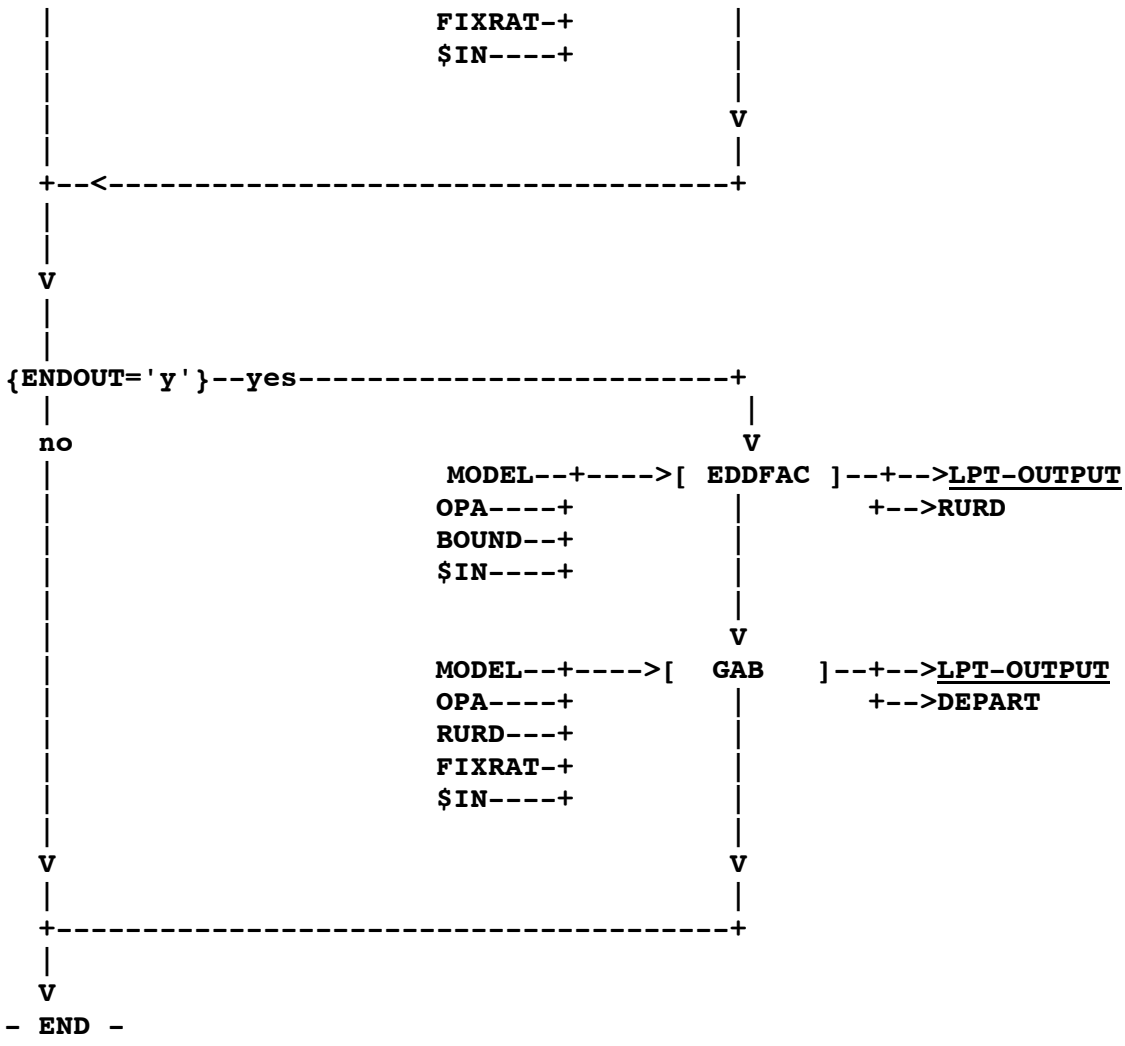
This program flow has to be obeyed in order to guarantee a correct use of the NLTE program package.

If a datafile is indicated to the left AND right of a program, this datafile is modified in the respective program.




$IN
MODELL
EDDFAC
RURD
OPA
BOUND
$IN
MODELL
GAB
LPT-OUTPUT
RURD
MODEL
FIXRAT
$IN
{ITER>0 ?}
yes
no
MODEL
OPA
BOUND
$IN
EDDFAC
RURD
RADFK
MODEL
RURD
FIXRAT
$IN
GAB
MODEL
DNIDZT
MODEL
OPA
RADFK
DNIDZT
RURD
$IN
LINEAR
LPT-OUTPUT
RURD
MODEL
RURD
FIXRAT
$IN
GAB
LPT-OUTPUT
MODEL
{ITER=ITER-1}
{ITER>1 ?}
no
yes


MODEL--+---->[EDDFAC]--+
OPA----+ +-->RURD
BOUND--+ +-->RADFK
$IN----+
MODEL--+---->[GAB]--+-->LPT OUTPUT
RURD---+ +-->MODEL
FIXRAT-+ +-->DNIDZT
$IN----+
MODEL--+---->[LINEAR]--+
OPA----+ +-->RURD
RADFK--+
DNIDZT-+
RURD --+
$IN----+
MODEL--+---->[GAB]--+-->LPT-OUTPUT
RURD---+ +-->MODEL
FIXRAT-+
$IN----+
{ITER=ITER-1}------>-------+
{ITER>0 ?}--yes--------------------+
no
+-<-------+
MODEL--+---->[EDDFAC]--+
OPA----+ +-->RURD
BOUND--+ +-->RADFK
$IN----+
MODEL--+---->[GAB]--+
RURD---+ +-->MODEL
FIXRAT-+ +-->DNIDZT
$IN----+
MODEL--+---->[LINEAR]--+-->LPT-OUTPUT
OPA----+ +-->RURD
RADFK--+
RURD---+
DNIDZT-+
$IN----+
MODEL--+---->[GAB]--+-->LPT-OUTPUT
RURD---+ +-->MODEL



1.4 Brief Explanation of Model Calculation

TTAU : The input datafile PTAU is interpolated on an optical depth scale with 50 steps.

ATM: A model atmosphere is calculated, that is for the given temperature stratification, T(tau), additional parameters are calculated: gas pressure, electron pressure, particle densities etc.

RATES: For a given wavelength grid, the continuum radiation field is calculated as well as the photoionization rates of all specified atomic levels.

START: This program calculates the electron and neutral particle collision cross sections of all transitions. It derives the respective rates connecting all specified atomic levels. The photoionization and collision rates are combined into the rate matrix Aij. In addition, the Voigt profile is calculated for the b-b transitions specified in the input datafile ATOM.

EDDFAC: The mean intensity J and the Eddington factors f are calculated for the specified frequency points of all individual lines.

GAB: This program calculates new level population numbers for all atomic levels. It takes into account the collision rates AND the photoionization rates.

EDDFAC: Using the new level population numbers, this program calculates the new J and f. It calculates now as well the radiative rates Rij and Rji of all b-b transitions.

GAB: New level population numbers are calculated taking into account collision, photoionization and radiative rates.

Iteration loop; until ITER>0:

EDDFAC:	From the last level population numbers, new values for J, f, Rij, and Rji are derived.
GAB:	Calculation of new level population numbers from Rij and Rji.
LINEAR:	To shorten the computation time, this programs solves interatively an equation, which gets corrections delta(Rij), delta(Rji) for the radiative rates. Using these corrections, new values of Rij and Rji are calculated.
GAB:	Calculation of new level population numbers from the new values of Rij and Rji.

At the end:

EDDFAC: New values for J, f, Rij and Rji.

GAB: New level population numbers. Output of the departure coefficients.

2 PART II

2.1 The Program Rates

RATES

Program to compute for a given wavelength grid the continuum radiation field and the photoionisation rates of all specified atomic levels.

- Reading the model atmosphere (File ATM)
- For each depth point:
  - Call IONDIS to compute the gas pressure, the particle densities Nkern and the particle densities ZETAJ and FRACJ:
    - ZETAJ: Fraction of a particle species that is in the ground state.
    - FRACJ: Fraction of a particle species.
- Reading parameter IFLPT from $IN (standard input) to control the standard output.
- Opening of the imput-file CROSS.
- Call subroutine INITEL to read the first part of CROSS
- Reading parameter NUMAX from CROSS to generate a radiation field with fixed rates (NUMAX > 0):
  NUMAX=0, otherwise program stops, because this mode of computation is no longer available.
- Reading parameter IFRAD to choose the radiation field
  - IFRAD=0: UV-field = BNU(T)
  - 1: JNU, cont
  - 2: JNU
- Reading parameters NUMAX, WAVEMAX, DELTAWA from CROSS to generate a radiation field using ODFs from the ATLAS6 code (NUMAX<=1200)
  NOTE: Now also ODFs from ATLAS9 possible (NUMAX<=1300)
- Reading the wavelength grid(WINT(NU), NU=1,NUMAX) to compute the radiation field
- Fill up wavelength grid if WINT(NUMAX)< WAVEMAX using the stepsize DELTAWA
- For all given wavelength grid points:
  - Call RADFLD to compute the radiation field JNUI
- Read parameter NRIJ, which specifies the number of bound-free transitions (b-f; Photoionisation)
- For all b-f transitions:
- Read parameters of transition:
  IREF, JREF, WEDGE, IMAX, (XWA(J), J=1,IMAX), (AWA(J), J=1,IMAX), ITRAD, ENDTRA

- If ITRAD>0 analytical radiation field = BNU(Trad): ENDTRA=Trad
    - ITRAD=1,2,3 denotes Trad=Tel , Tel(Photosph.), Tel(Chromosph.), respectively
    - If IMAX=1: hydrogen approximation for cross section
        - Computation of Rij and Rji using the function RINT
    - If IMAX>1: explicit cross section
        - Read wavelength ENDTRA
  - If IMAX>1
    - Compute JSTART and JEND from the wavelength grid
  - If ITRAD=0 radiation field = JNUI
    - If IMAX=1: hydrogen approximation for cross section
        - Piecewise analytic solution of the integrated cross section using stepsizes correspondiung to the wavelength grid
  - If IMAX>1: explicit cross section
    - Call INTSUM to compute partial sums between WINT(J) and WINT(J+1), if there are any wavelengthpoints of the resolved cross section between them (JNUI does not change)
    - Add up partial sums from INTSUM with varying JNUI to calculate RU and RD
  - If ITRAD>0 and IMAX>1
    - Numeric integration over the resolved cross section from the edge up to ENDTRA
  - If IFLPT<>0 standard output of the cross section integration
- Close file CROSS and if IFRAD=2 also file DF
- For all depth points
  - For all b-f transitions
    - The rates Rij and Rji are written to file RIJ
- End of program, close file RIJ

INITEL

Subroutine to read the first part of CROSS.

- Read element number (ELEM)
- Read scaling factor for opacities (SCSTART, SCEND, SCA, SCB)
- Read number of following ionization stages (NEMAX)
  - If NEMAX>0
    - Read ionization stage (NXTION(I), I=1,NEMAX)
- If DEPART.DAT exists
  - Read file DEPART
  - Stop program if element from CROSS does not coincide with elements from DEPART
- If IFLPT>0
  - Standard output of data

IONDIS

Subroutine to compute the gas pressure, the particle density NKERN and the particle densities ZETAJ and FRACJ.

- Call subroutine QATI1 to compute partition functions for the following elements: H, He, C, N, O, Ne, Na, Mg, Al, Si, P, S, Ar, K, Ca, Ti, Cr, Mn, Fe, Ni
- If MODE=2
    - Call subroutine QATI2 to compute partition functions for the following elements: Li, Sc, V, Co, Cu, Zn, Sr, Y, Zr, Ba, La, Ce, Nd, Sm, Eu, W, Os, Th
- Call subroutine DCHI to compute lowering of ionization energy
- Compute fraction SAHAJ = Nj/Sum[Nj] for all elements for four ionization stages Compute particle concentration for negative ions H-, C-, O- and renormalize positive ions Sum[Nj] = Sum[Nj] + N(c-)
- Cann subroutine QMOL to compute dissociation constant of the following molecules: H2, CH, C2, CN, CO, MgH
- For all species (atoms, ions, molecules)
    - Assign index
    - Compute atomic/molecular weight
- Compute electron concentration and abundances of atoms, positive ions, negative ions and molecules
- Iteration of abundances, because they depend on each other
- Compute the gas pressure and density NKERN
- If IPRINT=2
    - Standard output of all information

DCHI

Subroutine to compute the lowering of the ionization energy due to the electron pressure following the Debeye-theory.

- CHIJj = CHIJ - 4.98E-4*THETA*Sqrt[Pe]
- CHIJj+1 = CHIJ - 2*(4.98E-4*THETA*Sqrt[Pe])
- CHIJj+2 = CHIJ - 3*(4.98E-4*THETA*Sqrt[Pe])
- CHIJj+3 = CHIJ - 4*(4.98E-4*THETA*Sqrt[Pe])

RADFLD

Subroutine to compute the radiation field.

- For all depth points
    - Call OPALAM to compute continuum values of KAPPA and SIGMA (absorption and scattering coefficients)

    - Call ELOPA to compute NLTE opacity of the model atom
- Call RADI to solve the radiative transfer equation after Feautrier with variable Eddington factors for the continuum (without lines: JNUC)
- If IFRAD=2
    - If SCSTART<wavelength<SCEND
        - Compute scaling factors for opacity
- Call LINOP to compute line opacity with N=1 – weighting WT(N) of the result depends on N
- Call RADI to solve radiative transfer equation with line opacity (JNUI)
- For all depth points
        - Weighting of radiation field with WT(N)
        - Weighting of surface flux
- For all depth points
    - If JNUI/JNUC<0.9998, then N=N+1 and next call to LINOP
- If IFLPT>0
    - Standard output of surface flux HSURFI

ELOPA

Subroutine to compute NLTE opacity of the model atom.

- Old LTE opacity of the model atom gets substracted from the total opacity of all elements
- Call BFEXX to compute photoionisation cross sections of the relevant element (at this time only possible for aluminium and iron using BFE13 and BFE26)
- With help of the previously read departures (INITEL), the NLTE opacity of the model atom gets computed and added again to the total opacity of all elements

BFE13

Subroutine to compute photoionisation cross sections of aluminium (close to similar subroutine OPALAM for computation of all b-f opacities).

- If not Level 1 or 2
    - Level=0
    - Photoionisation cross section CROSS=0
- If wavelength WAVE>wavelength of edge ALIEDGE of level LEV
    - Photoionisation cross section CROSS=0
- If Level=2
    - CROSS = 1.2E-18 * (WAVE/ALIEDGE)^3 * 2.0 * Exp[-3.14 * 1.602192E-12/kT]
- If Level=1
    - If WAVE not in data grid for resolved cross section

        - CROSS = 65.0E-18 * (WAVE/ALIEDGE)^4 * 6.0 *Exp[-0.01 * 1.602192E-12/kT]
    - If WAVE in data grid for resolved cross section
        - Call function POLAT to interpolate in the grid
        - CROSS = POLAT(WAVE) * 1.0E-18 * 6.0 * Exp[-0.01 * 1.602192E-12/kT]

POLAT

Function to interpolate in arrays X and F which have dimension N for the value XQ.

- If X(1)<X(N)
    - Search the next larger field element i after XQ
- If X(1)>(N)
    - Search the next smaller field element i after XQ
- POLAT = F(i) - [F(i+1)-F(i)] / [X(i+1)-X(i)] * [XQ-X(i)]

RADI

Subroutine to solve the radiative transfer equation after Feautrier with variable Eddington factors (Auer, Mihalas 1970).

- For all depth points
    - Compute matrices A, B and C and the vector Q (=L), see Mihalas, Stellar Atmospheres, p.161 ff and Giagas, PhD thesis, p.75 ff
    - Compute zero, first and second moment of radiation field by summing over all angles
    - Compute Eddington factors FK, FH0 (surface) and FH1 (all depth points

LINOP

Subroutine to compute line opacity.

- For all depth points
    - Interpolation for temperature and electron pressure in tables
    - Computation of interpolation coefficients
- Fixing of the interpolation weights
- Read respective ODF for desired wavelength
- For all depth points
    - Interpolation of line opacity from tabulated values for fixed electron pressure and temperature

OPALAM

Subroutine to compute KAPPA- und SIGMA-values, absorption and scattering coefficients.

- All cross sections are given in units of the Thomson scattering coefficient
- Computation of scattering coefficients for Thomson scattering of free electrons and Rayleigh scattering of HI, HeI, H2
- Computation of free-free (f-f) absorption cross section for hydrogen, helium-nickel (four ionization stages per element), negative hydrogen ion (H-)
- Computation of bound-free (b-f) absorption cross sections for hydrogen, H-, H2, H2+, HeI, HeII, He-, C-, CI, NI, OI, NeI, NaI, MgI, MgII, AlI, SiI, SiII, CaI, CaII, FeI
- Summing up all scattering cross sections K=1,KSCAT
- Summing up all absorption cross sections K=KSCAT+1,KABS
- If MODE=2 detailed output of scattering and absorption cross sections of all atoms

RINT

Function to compute a cross section from the hydrogen approximation.

- Rij = 8*pi*c*alpha0 * E1[hc/(k*lam*T)] = AA * E1(X)
- For I=1,999
    - Call function EXPINT to solve E1[FLOAT(I)*X]
    - Summing up the solutions
    - RETURN if change in sum smaller than 1.0E-5

EXPINT

Function to support cross section computation.

- Solution of the integral of the exponential function E1(x) = Int[1/t * exp(-xt) dt]

INTSUM

Subroutine to compute partial sums (Integration of the function 1/LAM*CROSS over wavelength between two neighboring WINT(J)).

2.1.1 The Input Datafile ATM

The datafile ATM contains extensive information on the model atmosphere, which has been computed by the program ATM (ATMOS). A detailed description can be found in the manual for the program system model atmosphere.

2.1.2 The Input Datafile DF

The datafile DF contains the distribution functions of the line opacities from Kurucz (ATLAS6, ATLAS9).

2.1.3 The Input Datafile $IN

The notation datafile is somewhat sloppy in this context, because $IN denotes the standard input of each program. This input is passed onto the program by the script that controls the flow of the program system. The program RATES takes as standard input a single value, namely IFLPT. This value controls the standard output.

2.1.4 The Input Datafile CROSS

The datafile CROSS is one of the two files that have to be explicitly generated for the NLTE code. The file CROSS serves as input file for the program RATES.

It contains the desired frequency grid, controls the character of the radiation field that gets computed and then later used for the calculation of the photoionisation rates. In addition, it contains an explicit list of the bound-free transitions, for which rates will be calculated.

To make this filename unique and meaningful, it can have a specific name such as ci_cross.dat. However, once the program system is started, the file has to get copied or linked to the name CROSS. This can be done within the script that controls the flow of the program system.

The file CROSS has to be structured as follows:

*** First line:** IELEM

Format: I5

Explanation: IELEM denotes the atomic number of the element, e.g.
FeI --> IELEM = 2600
BaII--> IELEM = 5601

*** Next new line:** SCSTART SCEND SCA SCB

Format: free

Explanation: SCSTART denotes the starting wavelength, SCEND the end wavelength for scaling the line opacity. SCA and SCB are the scaling factors.

!!IMPORTANT!! This scaling is meant primarily for ATLAS6 ODFS. The missing UV opacity in those ODFs was compensated by

1675 2076 -1.7e-3 1.0

in the case of the Sun. If ATLAS9 ODFs are used, this option has to be switched off by using

1000000 0 0 0 .

*** Next new line:** NXTION

Format: I4

Explanation: Meaning unknown

*** Next new line:** NUMAX

Format: free

Explanation: NUMAX=0 : no further significance for program flow
>0: computation of a radiation field JNU(TAU) for NUMAX lines (with fixed rates). The JNU(TAU) values are stored in RADIAT

If NUMAX>0, give in the following lines the wavelength of the lines to be calculated in Å, NUMAX wavelength points in format 4F10.3.

!!IMPORTANT!! This does not work any longer. It leads to program abort.

*** Next new line:** IFRAD

Format: free

Explanation: IFFRAD controls the way JNU is calculated. Since photoionisation is mostly relevant in the ultraviolet spectral range, the radiation field is named UV

IFRAD=0 : UV-Field = BNU (Trad)
IFRAD=1 : JNUC
IFRAD=2 : JNUI (with line opacities).

*** Next new line:** NUMAX WAVEMAX DELTAWA

Format: free

Explanation: All three are control parameters that define the wavelength grid

NUMAX: denotes the total number of following wavelength points; NUMAX <= 1200
(if version with ATLAS9 ODF's is used: NUMAX <=1300)

WAVEMAX: denotes the last wavelength point
DELTAWA: denotes the stepsize of the wavelength grid; this is only important for extrapolation of the fllowing explicitly listed wavelength grid.

*** Next new line:** WAVE

Format: free

Explanation: this is the explicitly listed grid of wavelength points, for which the radiation field will be computed.

WAVE has to be tabulated in Å (if version with ATLAS9 ODF's is used, WAVE has to be in nm)

A total of NUMAX wavelength points have to be listed.

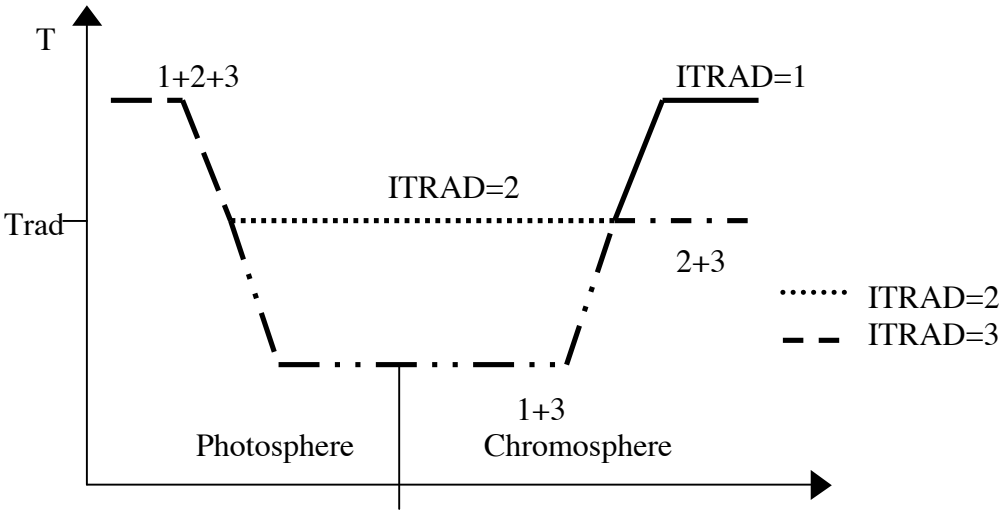
*** Next new line:** NRIJ

Format: free

Explanation: NRIJ denotes the number of following b-f transitions; NRIJ<=99.

In the next NRIJ lines, the following parameters are listed next to each other in free format:

IREF:	lower level of b-f transition
JREF:	upper level of b-f transition
WEDGE:	wavelength of edge (ionization energy) in Å
IMAX:	number of cross section points
XWA:	wavelengths, IMAX points
AQUER:	corresponding cross sections in 1E-18 cm^2, IMAX points
ITRAD:	control parameter to decide which radiation field is used to calculate the photionisation cross sections. ITRAD=0 : UV-Field = JNU ITRAD=1 : BNU(TRAD), TRAD=Tel ITRAD=2 : BNU(TRAD), TRAD=Tel(photosph.) ITRAD=3 : BNU(TRAD), TRAD=Tel(chromosph.)



Further parameters in this line: two options

a) if ITRAD=0:

ENDTRA: the cross section is calculated down to LAMBDA=ENDTRA

b) if ITRAD>0:

ENDTRA: denotes the radiative temperature Trad

if IMAX>1 then in addition in a new line (!!!):

ENDTRA: see (a) for meaning

Note: Please keep in mind that IFRAD=0 and ITRAD=0 cannot be at the same time, because IFRAD=0 precludes computation of JNU, while ITRAD=0 calculates the photoionisation rates from the JNU values. This will of course fail.

------ Example for a CROSS-datafile: CI ----------

```
 600                ielem
100000  0  0  0         from to (lambda) scalea scaleb
  0                 nxtion for depart
  0                 dummy
  2                 2: J(nu) with line opacity, 1: J(nu), 0: B(T_rad)
336  50000.0  500.0

   227.941    232.000    236.522    250.074    254.661    259.000
   261.432    271.000    280.926    286.587    299.585    302.626
   317.000    335.000    349.700    352.500    361.410    370.000
   380.160    389.664    411.549    418.799    433.000    459.000
   485.200    504.302    508.763    525.000    555.000    574.100
   600.000    625.000    655.379    688.000    722.401    750.000
   790.000    830.000    870.000    911.764    930.000    950.000
   975.000   1000.000   1025.000   1050.000   1075.000   1100.920
  1125.000   1150.000   1175.000   1200.000   1220.000   1239.618
  1270.000   1300.000   1325.000   1350.000   1375.000   1400.000
  1422.000   1444.018   1470.000   1495.000   1520.000   1545.000
  1570.000   1595.000   1621.507   1650.000   1676.721   1700.000
  1725.000   1750.000   1775.000   1800.000   1830.000   1850.000
  1875.000   1910.000   1930.000   1950.000   1978.447   2000.000
  2025.000   2050.000   2077.564   2100.000   2125.000   2150.000
  2175.000   2200.000   2225.000   2250.000   2275.000   2300.000
  2325.000   2350.000   2375.000   2400.000   2425.000   2450.000
  2475.000   2500.000   2513.815   2525.000   2550.000   2575.000
  2600.000   2625.000   2650.000   2675.000   2700.000   2725.000
  2750.000   2775.000   2800.000   2825.000   2850.000   2875.000
  2900.000   2925.000   2950.000   2975.000   3000.000   3025.000
  3050.000   3075.000   3100.000   3125.000   3150.000   3175.000
  3200.000   3225.000   3250.000   3275.000   3300.000   3325.000
  3350.000   3375.000   3400.000   3425.000   3450.000   3475.000
  3500.000   3525.000   3550.000   3575.000   3600.000   3625.000
  3647.055   3675.000   3700.000   3725.000   3750.000   3775.000
```

```
 3800.000  3825.000  3850.000  3875.000  3900.000  3925.000
 3950.000  3975.000  4000.000  4025.000  4050.000  4075.000
 4100.000  4125.000  4150.000  4175.000  4200.000  4225.000
 4250.000  4275.000  4300.000  4325.000  4350.000  4375.000
 4400.000  4425.000  4450.000  4475.000  4500.000  4525.000
 4550.000  4575.000  4600.000  4625.000  4650.000  4675.000
 4700.000  4725.000  4750.000  4775.000  4800.000  4825.000
 4850.000  4875.000  4900.000  4925.000  4950.000  4975.000
 5000.000  5050.000  5100.000  5150.000  5200.000  5250.000
 5300.000  5350.000  5400.000  5450.000  5500.000  5550.000
 5600.000  5650.000  5700.000  5750.000  5800.000  5850.000
 5900.000  5950.000  6000.000  6050.000  6100.000  6150.000
 6200.000  6250.000  6300.000  6350.000  6400.000  6450.000
 6500.000  6550.000  6600.000  6650.000  6700.000  6750.000
 6800.000  6850.000  6900.000  6950.000  7000.000  7050.000
 7100.000  7200.000  7300.000  7400.000  7500.000  7600.000
 7700.000  7800.000  7900.000  8000.000  8100.000  8205.874
 8300.000  8400.000  8500.000  8600.000  8700.000  8800.000
 8900.000  9000.000  9100.000  9200.000  9300.000  9400.000
 9500.000  9600.000  9700.000  9800.000  9900.000 10000.000
10100.000 10200.000 10300.000 10400.000 10500.000 10600.000
10700.000 10800.000 10900.000 11000.000 11100.000 11200.000
11300.000 11400.000 11500.000 11600.000 11700.000 11800.000
11900.000 12000.000 12100.000 12200.000 12300.000 12400.000
12500.000 12600.000 12700.000 12800.000 12900.000 13000.000
13100.000 13200.000 13300.000 13400.000 13500.000 13600.000
13700.000 13800.000 13900.000 14000.000 14100.000 14200.000
14300.000 14400.000 14500.000 14588.221 14700.000 14800.000
14900.000 15000.000 15200.000 15400.000 15600.000 15800.000
16000.000 16200.000 16400.000 16600.000 16800.000 17000.000

88

 1 84 1101.
19  1101. 1013. 911. 881. 869. 860. 842. 814. 804. 798. 780. 777. 753. 701.
    651. 597. 490. 466. 429.
    17.2 16.7 16.4 17.0 18.2 11.9 14.0 14.8 17.7 13.4 16.6 13.2 14.9 12.6
    10.9 9.8 7.1 6.6 5.3
    0   430.0

 2 84 1240.
18  1240. 1171. 1139. 1070. 1044. 1000. 930. 911. 895. 893. 828. 759. 701.
    651. 570. 506. 414. 351.
    10.5 9.7 10.8 25.0 45.0 35.0 22.6 51.8 69.1 33.0 15.6 10.9 10.7 9.7
    7.7 6.5 5.2 3.9
    0   355.0
```

```
 3 84 1445.
13  1445. 1302. 1139. 1040. 1013. 997. 959. 911. 828. 701. 570. 456. 351.
    14.3 12.8 8.5 3.6 30.2 139.4 39.6 21.6 14.4 10.2 7.7 5.9 3.6
    0   355.0

 4 84 998.
23  998. 988. 946. 900. 819. 751. 694. 645. 602. 565. 532. 502. 476. 431.
    394. 362. 335. 312. 292. 275. 259. 245. 232.
    13.13 13.29 13.84 14.28 14.55 14.38 13.93 13.34 12.61 11.89 11.16 10.47
    9.778 8.535 7.457 6.515 5.700 5.058 4.460 3.899 3.455 3.078 2.799
    0   235.0

     |
     | u.s.w.
     |

88 89 530.
23  530. 527. 515. 501. 475. 451. 430. 411. 393. 377. 362. 348. 335. 312.
    292. 274. 259. 245. 232. 221. 211. 201. 193.
    6.331 6.274 6.092 5.873 5.448 5.067 4.691 4.359 4.043 3.772 3.504 3.247
    3.032 2.660 2.338 2.025 1.794 1.588 1.407 1.257 1.132 1.023 0.9163
    0   230.
```

2.1.5 The Output Datafile RIJ

This datafile contains the computed photionisation rates Rij and Rji. This datafile is unformatted. The data is organized as follows:

[{ Rij(ID,I),Rji(ID,I) } I=1,...,NRIJ] ID=1,...,NDEPTH.

2.1.6 The Standard Output of RATES

The standard output contains information about the radiation field, which has been computed in RATES and used for the photoionisation rates. The following table is compiled for the wavelength grid specified in CROSS:

1. Wavelength in Å
2. The depth at which tau=1
3. lg HNUI surface flux with line opacity
4. Ratio of HNUI/HNUC
5. Loop counter for the correction of the intensity JNU due to line opacity

2.2 The Program START

START

Program to calculate the electron and neutral particle collision cross sections of all transitions. It derives the respective rates connecting all specified atomic levels. The photoionization and collision rates are combined into the rate matrix Aij. In addition, the Voigt profile is calculated for the b-b transitions specified in the input datafile ATOM.

- Opening the input datafile ATOM
- Read in title
- Standard output of title
- Reading number of levels (NL), lines (NT) and transitions with fixed rates (NRFIX)
- Read in NL atomic levels (IR, EPS, DI, RRCA, G, NXTION, LABEL)
- Initialize reference array IJREF(IR)=IL (only important if energy levels are not initially sorted in ascending order)
- Computation of the charge number IZ for each level
- Standard output of these values
- Read DLOGC6, DLOGC4, DLOGGR
- Read NT lines (IR, JR, LAMT, GFLG, C4LG, RAD, IFMAX, PHOTO)
  - If PHOTO=F
    - Standard output of these values
  - If PHOTO=T
    - NPT=NPT+1 number of b-f transitions
    - Initializing two arrays to change from NPT to IT and vice-versa (IPIT and ITIP)
    - Read the photoionisation cross section (PLAM(IJ,NPT), PSIG(NJ,NPT), PWT(IJ,NPT),IJ=1,IFMAX)
    - IT=IT+1
- If NPT>0 standard output of photoionisation transitions
- Read NRFIX transitions with fixed rates (IR, JR, PHOR, OCR)
  - If PHOR=F and JR<NXTION(IR)
    - Compute wavelength from IR and JR
  - Standard output of transitions with fixed rates
- Call of subroutine ATMOS to compute the model atmosphere
- Call of subroutine RATCAL to compute the collision rates Cij and Cji. If necessary, the photoionisation rates and/or some chosen line transitions with fixed rates get added up.
- Call subroutine CHIGEN to compute line profile for those b-b transitions that are specified in ATOM
- Call subroutine OUTMOD to generate the output datafile MODEL

ATMOS

Program that calculates a model atmosphere for the given temperature stratification, T(tau). Additional parameters are calculated: gas pressure, electron pressure, particle densities etc..

- Open input datafile ATM
- Read element composition (NAMI, WTI, AEPS)
- HE = [He/H]
- WEI = atomic weight of model atom
- Compute element fractions FRACI and mean atomic mass AVEM from the listed element abundances
- Standard output of abundances
- Read model atmosphere (NDEPTH, TEFF, GLOG, REFLAM, RSU)
  (TAUREF(ID), ID=1,NDEPTH), (TEMP(ID), ID=1,NDEPTH),
  (PELG(ID), ID=1,NDEPTH), (PGLG(ID), ID=1,NDEPTH),
  (REFKAP(ID), ID=1,NDEPTH), (REFSIG(ID), ID=1,NDEPTH),
  (R(ID), ID=1,NDEPTH), (XI(ID), ID=1,NDEPTH)
- Close data file ATM
- Call subroutine INTEGR to compute a mass scale MA(TAU) in g/cm^2
- For all depth points
  - Call subroutine IONDIS to compute particle densities
  - Compute particle density for model atom NATOMT
  - Compute total density RHO(TAU) in g/cm^3
  - If ID=1
    - Standard output of ionisation potentials for the model atom
  - Compute fraction of hydrogen in ground state ZETAH
  - Compute level population density of the continuum
  - Adding up total level population density
- Standard output of model atmosphere (TEFF, GLOG, HE, RSU, GG, REFLAM)
  (ID, TRLOG, MA, MLOG, TEMP, PELG, PGLG, RHO, XI, R, NATOMT, NATOM, ID=1,NDEPTH)
- If RSU<0
  - Standard output ' ASSUMING CONSTANT LOG G '

Note: If RSU>0.0, LOG G is computed using the following formula (see ATMOS code description):

$$\text{LOG G} = \text{LOG G(TAU1)} + 2*\text{LOG}(\text{R(TAU1)}/\text{R(TAU)})$$

INTEGR

Subroutine to integrate an equation. INTEGR is identical with the subroutine INTEG, except for the smaller dimension of the used arrays.

- Call subroutine PARCOE to compute the parabolic coefficients aj, bj and cj
- Int[x dx] = Sum[(aj +
  bj * (xj+1 + xj)/2 +
  cj * (xj+1^2 + xj+1 * xj + xj^2)/3) * (xj+1 - xj)]

PARCOE

Subroutine to compute the parabolic coefficients as weighted mean between forward and backward parabola.

RATCAL

Subroutine to compute the collision rates Cij and Cji.

- Opening the input datafile RIJ
- Opening the output datafiles FIXRAT and COLRAT
- Call subroutine COLSEC to compute collision cross sections for all levels of the model atom
- For all depth points
  - Call subroutine COLRAT to compute the collision rates Cij and Cji
  - Initialization of the matrix Aij for I,j=1…NK (all zeros);
    NK is the number of levels including the continuum
    - For all b-f and b-b transitions with fixed rates
      - If b-f transition
        - Read Rij and Rji from file RIJ
        - Multiply Rji with the ratio of level population numbers Ni/Nj
      - If b-b transition
        - If IREAD=0 (1.Iteration)
          - Open datafile RADIAT (does not exist any longer, probably one has to write this file out in RATES5)
          - IREAD=1
        - Read JNU from RADIAT
        - Compute Rij and Rji
        - Add Rij and Rji to the collision rates Cij and Cji; the names Cij and Cji stay
        - For all possible transitions between the levels of the model atom
          - Aij = -Cji
          - Aii = Sum,k[Cik]

        - For all j
        - A(NK,j)=1
        - Write the matrix Aij and the total particle density to the output datafile FIXRAT
- Write level population numbers for all depth points and all levels to FIXRAT
- If IREAD=1
    - Close the datafile RADIAT
- Close the datafiles RIJ and FIXRAT
- If IFRATIO>0
    - Standard output of the ratio of neutral particle and electron collision rates

COLSEC

Subroutine to compute collision cross sections for all elements except Helium (currently Helium needs its own subroutine, which unfortunately got lost in this program version).

- All necessary variables are set to zero for all possible transitions between all levels (also ICIND=0)
- Read the number of valence electrons XXI1, XXI2
- Read scaling factors SCALE and FTOAL, as well as the control parameter IPIA02 to compute the collision cross sections
- Electron collisional ionization cross sections are computed for all levels (Drawin #33)

  COLL(IC) = (EH/Eij)^2 * 0.665 * XXI
- Electron collisional excitation cross sections are computed for all levels
    - If IPIA02=2

      COLL(IC) = -r^2[a0^2] * SCALE
    - If IPIA02=1

      COLL(IC) = -SCALE
    - If IPIA02=0

      COLL(IC) = -(EH/Eij)/gi * SCALE

Note: For FeII, all electron excitation cross sections are set to +1.0*SCALE, because all transitions are treated as optically allowed. Only a few transitions with explicitly given cross sections from Nussbaumer are computed in COLRAT as optically forbidden transitions.

The preliminary negative sign of all cross sections has no meaning. In the following, the cross sections of optically allowed transitions are multiplied again by –1. Then, only optically forbidden cross sections have a negative sign.

- Read the number of optically allowed transitions, for which default cross sections are to be computed

- Read lower and upper level numbers of these transitions
- Scale optically allowed cross sections with factor FTOAL
- Read number of cross sections for which collision rates are to be computed using the Born approximation
- Read lower and upper level numbers of these transitions
- ICIND(IC)=-1 (set flag for Born cross sections)
- Read scaling factors for b-f and b-b transitions for which other electron cross sections are to be computed
- Read number of these transitions
- Read these transitions (ICLOW, ICUP, CCS)
  - If ICUP>0
    - Sign of ICLOW determined sign of CCS and therefore if the collision is optically allowed or forbidden
      COLL(IC) = CCS
    - If ICUP<0 and ICLOW>0
      - COLL(IC) = (EH/Eij)^2 * fij
      - If carbon
        - Minimum cross section = 0.1
    - If ICUP<0 and ICLOW<0
      - Abort the program
- Read number of transitions for which energy resolved cross sections are to be computed
- Read these transitions (ICLOW, ICUP, NPOINTS)
  - IS = IS + 1
  - Read the resolved cross section (ECOLL(IS,II), SCOLL(IS,II), II=1,IABS(NPOINTS))
  - ICIND=IS (set flag for energy resolved cross sections)
  - COLL(IC) = FLOAT(NPOINTS)
- Set default cross sections for neutral particle collsions (neutral particle cross section = electron cross section, that is neutral particle cross section = MAX(E-cross section) in case of energy resolved cross sections) COLNEU ICINDN = 0
- Compute standard deviation and variance for all electron collision cross sections
- Search for smallest and largest electron cross section
- Calculate mean electron cross section
- Standard output for electron cross section
- Read scaling factors for neutral particle collision cross sections FH1BB, FH1BF
- Read number of transitions for which other than default neutral particle collision cross sections are to be computed
- Read these transitions (ICLOW, ICUP, CCS)
  - COLNEU = CCS
- Read number of transitions for which energy resolved neutral particle cross sections are available
- Read these transitions (ICLOW, ICUP, NPOINTS)

- IS = IS + 1
- Read resolved cross sections (ECOLL(IS,II),
  SCOLL(IS,II), II=1,NPOINTS)
- COLNEU = FLOAT(NPOINTS)
- ICINDN = IS

- Standard output of cross sections of neutral particle collisions

COLRAT

Subroutine to compute collision rates for all elements except helium (the subroutine HECLRT got lost in this program version).

- Set constants and functions for computation of collision rates (for details see Sven Stürenburgs manual on computation of collision cross sections and PhD thesis of D. Gigas)
- Call subroutine EXPINT
- Initialize matrix Cij (set all values to zero)
- If necessary, read datafile BORNDAT
- For all possible transitions
  - If ICIND=0
    - Compute rate for b-f electron collisions
      Aij = AICON * Abs(COLL) * F1
  - If ICIND>0
    - Call subroutine INTEGR to integrate energy resolved collision cross section
      Aij = K * Int[ECOLL/kT*Exp(-ECOLL/kT) * SCOLL dE]
  - If ICINDN=0

b-f

    Bij = BICON * COLNEU * PSI
  - If ICINDN>0
    - Call subroutine INTEGR to integrate energy resolved collision cross section
      Bij = K * Int[ECOLL/kT * Exp(-ECOLL/kT) * SCOLL dE]
  - Compute the ratio of neutral particle to electron collision rates
  - Compute the total b-f collision rate
    Cij = Aij + Bij
    Cji = Cij * ni/nj (from detailed statistical equilibrium)
  - If ION=1, that is neutral model atom
    - Compute spin of lower level I
    - If ICIND=0
      - If COLL>0, that is optically allowed collision
        - Compute the b-b electron collision rate
          Aij = ACON * 0.8278 * Abs(COLL) * F1
      - If Coll<0, that is optically forbidden collision
        - Compute spin of upper level j
        - Compute DELTAS = Spin(i) - Spin(j)

- If DELTAS=0
  - Compute b-b electron collision rate

b-b

$$Aij = ACON * Abs(COLL) * F0$$

- If DELTAS<>0 and I=1
  - Call function MAP2 to compute function F3 by interpolation in the arrays UIJ and PSI3
  - Compute b-b electron collision rate

neutral

$$Aij = ACON*5.38*Abs(COLL)*W*F3$$

- If DELTAS<>0 and I>1
  - Compute b-b electron collision rate

$$Aij = ACON * Abs(COLL) * F00$$

  - If element not magnesium
    - Aij = 4 * Aij (consistent matching with optically allowed transitions, see PhD thesis of D. Gigas)

- If ICIND>0
  - Call subroutine INTEGR to integrate energy resolved collision cross section

$$Aij = K * Int[ECOLL/kT * Exp(-ECOLL/kT) * SCOLL\ dE] * FELBB$$

- If ICIND=-1
  - Call subroutine BORNCR to compute Born collision cross section BRAT
  - Compute b-b electron collision rate

$$Aij = NE * BRAT * FELBB$$

  - If element magnesium
    - Call function CLRTMG
    - If CLRTMG=-1
      - Compute b-b electron collision rate

$$Aij = NE * BRAT * FELBB$$

- If ION>1
  - If ICIND=0
    - If COLL>0
      - Call function MAP2 to compute F2 by interpolation in tables UIJ and PSI2
      - Compute b-b electron collision rate

$$Aij = ACON * Abs(COLL) * W * F2$$

    - If COLL<0
      - Compute b-b electron collision rate

b-b

$$Aij = ACON * Abs(COLL) * F00$$

      - If element not magnesium or barium
        - Aij = 4 * Aij (explanation see above)
      - If element is barium
        - Aij = 1.64 * Aij
  - If ICIND>0

- Call subroutine INTEGR to integrate energy resolved collision cross section

ion

$$A_{ij} = K * \text{Int}[ECOLL/kT * \text{Exp}(-ECOLL/kT) * SCOLL\ dE] * FELBB$$

- If element Barium
  - If function CLRTBA=-1
    - $A_{ij} = NE * BRAT * FELBB$
- If ICINDN=0
  - Compute b-b neutral particle collision rate
    $B_{ij} = BCON * COLNEU * PSI$
- If ICINDN>0
  - Call subroutine INTEGR to integrate energy resolved collision cross section

$$B_{ij} = K * \text{Int}[ECOLL/kT * \text{Exp}(-ECOLL/kT) * SCOLL\ dE] * FH1BB$$

- Compute total b-b collision rate
  $C_{ij} = A_{ij} + B_{ij}$
  $C_{ji} = C_{ij} * n_i/n_j$ (from detailed statistical equilibrium)

WRBIN

Subroutine for unformatted read and write from/to datafiles. WRBIN has six entry points: RE1, RE2, RE3, WR1, WR2, WR3. One, two or three arrays of dimension K, L, M are read/written from/to unit IUN.

WRIBIN

Subroutine for unformatted read and write from/to datafiles. WRIBIN has also six entry points: RE1I, RE2I, RE3I, WR1I, WR2I, WR3I. The use is the same as for WRBIN, but here integer arrays are read/written.

BORNCR

Subroutine to compute collision cross sections BRAT with the Born approximation.

- See if data is available from tables for both l-values
- Set geometric form factors
- Derive effective quantum number
- Find position in tables (if necessary extrapolation)
- Interpolation in tables
- Compute collision cross section BRAT using the Born approximation

CLRTBA

Function to compute collision rates from the Born-Coulomb calculations by Sobelman for barium.

CLRTMG

Function to compute collision rates from the Born-Coulomb calculations by Sobelman for magnesium.

MAP2

Subroutine for parabolic interpolation (same formalism as PARCOE).

OUTMOD

- Open the output datafile MODEL
- Model atom and model atmosphere data are written to the file MODEL
- Close datafile MODEL

Note: Details on the data, see 2.2.8 for structure of output datafile MODEL!

CHIGEN

Subroutine to compute line profile of the given b-b transitions.

- Open output data files OPA and BOUND
- Read desired number of frequency points NFMAX1 from ATOM
- If NFMAX1>0
  - Read NFMAX1 distances from line center in units of maximal Doppler width
- If NFMAX1=0 (standard case)
  - Standard table with 30 frequency points
- Standard output of frequency grid
- For all depth points
  - Compute the Doppler velocity VDOP
- For all depth points
  - Find depth NDMAX at which VDOP has its maximum
- VDSTD = VDOP (NDMAX)
- Standard output of title
  LAMBDA SPACING IN THE LINES ....
- For all depth points
  - Computation of pre-factors for line broadening parameters after Lindholm
- For all transitions NT

- If b-b transition
  - For all depth points
    - Call subroutine OPALAM to compute the continuum opacity CHI (absorption) and CHIE (scattering)
  - Call subroutine XBACKOP to scale the continuum opacity and compute opacity of background lines
  - If C4LG=0
    - Electron collisional broadening following Griem/Cowley
  - For all depth points
    - If C4LG<0
      - No electron collisional broadening
    - For all frequency points
      - Compute Voigt function using the function H
      - Sum up entire profile for check
    - Find depth at which this sum has a minimum/maximum
  - For all frequency points
    - For all angles MU
    - Compute the radiation field BC at the inner edge using the asymptotic form of the radiative transfer equation
      BC = BNU + MU * DBNU/DTAU (Mihalas, p.29)
- If b-f transition
  - For all frequency points
    - For all depth points
      - Call subroutine OPALAM to compute the continuum opacity
      - Remove LTE opacity of model atom
      - Call subroutine LINOP to compute line opacity
      - For all depth points
        - Scale line opacity and add continuum opacity
      - For all angles
        - Compute the radiation field BC at the inner edge using the asymptotic form of the radiative transfer equation
          BC = BNU + MU * DBNU/DTAU
          (Mihalas, p.29)
  - Standard output of b-f transition
- Close OPA and BOUND

XBACKOP

Subroutine to scale the continuum opacity and to compute opacity of background lines.

- ITT and ITYP are set to zero

- f-values, statistical weights, excitation energies of lower level, wavelengths and quantum numbers of lower/upper level for all Lyman-, Balmer-, Paschen- and Bracket-lines are set
- If IT>ITT
  - Read ITT and ITYP from ATOM
- Scaling factor OPSCA and background line opacity LIOP are set to 1 and 0 respectively
- If IT=ITT
  - If ITYP=1
    - Read OPSCA from ATOM
    - Standard output of the line and the scaling factors
  - If ITYP>=100
    - Read background line from ATOM (BLAM, BCHI, BGFLG, BDRRCA, BC4LG, BRAD)
    - Standard output of the line with respective data for the background line
    - Find background element in array NAMI
    - If not found stop program
    - If element is hydrogen
      - Compute pre-factors for line broadening of hydrogen lines; log K+17 formalism from program Linfor
    - If other element
      - Compute pre-factors for line broadening of metal lines
    - For all depth points
      - Compute the Doppler velocity
      - If element is hydrogen
        - Compute line opacity using the function XPROFHY to compute hydrogen line profiles
      - If other element
        - Compute line opacity using the H function to calculate the Voigt function for the line
  - Else stop program, because ITYP is wrong
  - Read ITT and ITYP from ATOM
- Else if ITT<0
  - If IT=1: Read VCS-tables
  - Find the appropriate hydrogen line relevant for IT
  - Take also into account the two hydrogen lines to the right and left
  - For all depth points
    - For all 5 hydrogen lines
      - If within 300 Å of IT
        - Interpolate in the VCS tables
        - Convert normalized line profile from [ESU/Å] to [s]
        - Sum up opacity of respective depth CHI(ID)
  - Jump back in UATOM, so that background opacity is taken into account once for each line
- For all depth points

- Continuum opacity/AVEMAT

XPROFHY

Function to compute hydrogen line profile.

- If DeltaLambda<5*DeltaLambdaD (DL<5*DLD)
  - Compute XPROFHY using function XASH to calculate an asymptotic profile
- If DL>=5*DLD
  - XPROFHY = XASH

XASH

Function to calculate an asymptotic profile.

- XASH = (1 + R`*Sqrt(DeltaLambda) + Ce/CH*pH/pe) / (DeltaLambda^5/2) after Griem (1962)

H

Function to compute the Voigt-function.

- H(ALPHA, V) = Sum,n[ALPHA^n * Hn(V)], where Hn(V) is tabulated for n<=4 by Harris (1948)

2.2.1 The Input Datafile ATOM

The datafile ATOM is the second large datafile that has to be explicitly generated for the NLTE program. It contains data about the model atom that is going to be analysed.

The file ATOM is an iput file for the program START and has to have this name for the computation. In libraries and directories it can (and probably should) have a more specific name.

The file has to be structured as follows:

*** First line:** HEAD

Format: free

Explanation: HEAD is a title to mark the standard output. It is possible to have the CRAY job or script that handles the program flow insert a title here, so that HEAD is a dummy.

*** Second line:** NAME

Format: free

Explanation: NAME is the element designation, that is the atomic number of the element with two additional identification numbers.
Example: 5600 is Barium BaI
2601 is Iron FeII

*** Third line:** NL NT NRFIX

Format: free free free

Explanation: NL is the number of following listed atomic levels.
NT is the number of explicitly listed b-b and b-f transitions, that are taken into account for the linearization.
NRFIX is the number of b-b and/or b-f transitions with fixed rates.

*** next NL lines:** each has the following six quantities

Designation	Format	Explanation
EPS	free	Energy of the level in eV
DI	free	Excitation energy of parent ion in eV

RRCA	free	mean square orbital radius in units of the square Bohr radius
G	free	Statistical weight
NXTION	free	Level number of the level, to which this level ionizes or number of the continuum if the ionization starts from the last ionization stage
LABEL	free	Level designation

Note: About NXTION: If two ionization stages plus continuum are to be computed, the code needs the number of the ground level of the second ionization stage. The program automatically takes NXTION(1) as ground level of the second ionization stage, that is it assumes that the ground level of the first ionization stage ionizes into the ground level of the second ionization stage.

If different: change program!!!!

About DI: The quantities DI and NXTION work together (parent ion: one electron less than before)

DI=0, if the level to which the ionization leads is either directly listed or the continuum of the model atom (total number of levels plus one)

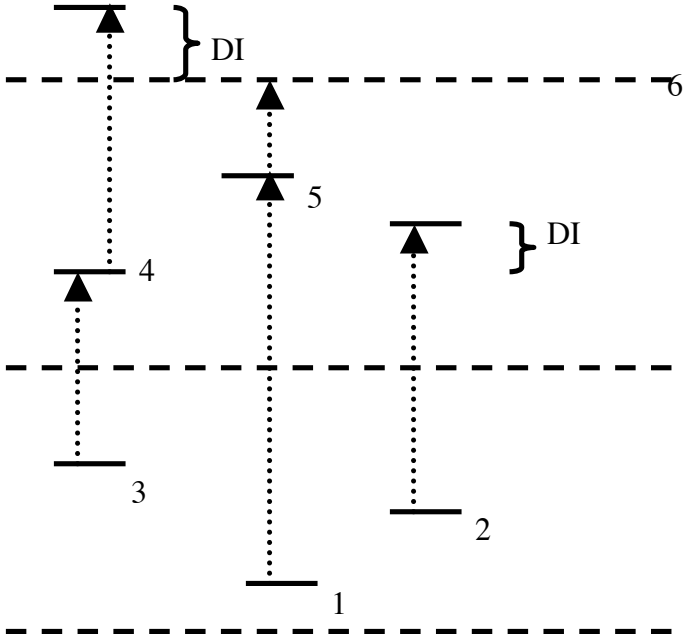
DI<>0, if the level to which the ionization leads is not explicitly listed. Then DI is the energy difference between the true level and the level given by NXTION in eV.

The continuum has the energy zero. If the level to which the ionization leads is not zero, the energy difference has to be specified here as well.

Example: Five listed energy levels plus continuum with the respective ionization transitions

The value of DI for level 2 and four is non-zero. DI is the energy distance in eV to the level listed as NXTION.

NXTION(2)=4, NXTION(4)=6.



*** next new line:** DLOGC6 DLOGC4 DLOGGR

Format: free free free

Explanation: DLOGC6 = DELTA(log [C6]) (=+1:normal)
DLOGC4 = DELTA(log [C4]) (=0 :normal)
DLOGGR = DELTA(log [GAMMA rad]) (=0 :normal)

These are scaling factors for the respective quantities. A value of +1 means that the computed or read quantity gets multiplied by a factor 10^+1 = 10. In case of –1, division by 10.

*** next NT lines:** (explicit transitions)

Designation	Format	Explanation
IR	free	from
JR	free	to level number
LAMT	free	wavelength of line in Å if PHOTO=F(alse) End wavelength for scaling the opacity (SCEND) if PHOTO=T(rue)

GFLG	free	log(gf), if PHOTO=F(alse), SCA, if PHOTO=T(rue)
C4LG	free	-log(C4), if PHOTO=F(alse), SCB, if PHOTO=T(rue);
	C4LG<0, no Stark broadening C4LG>0, C4 after Mihalas Gamma=DELTA(Omega)=11.37*C4^1.5*v^.33*N C4LG=0, approximation after Cowley/Griem (see Linfor-code description)	
RAD	free	GAMMA rad in 10^8 sec^-1
	RAD=-1 program computes the classical value GAMMA=GAMMAc =2.22E7/LAM^2 (LAM in Å), if PHOTO=F(alse) Dummy, if PHOTO=T(rue)	
IFMAX	free	Number of frequency points that are to be used to compute the Voigt profile (half profile!), if PHOTO=F(alse) Number of following triple of resolved photoionisation cross section, if PHOTO=T(rue)
PHOTO	free	T(rue) for photoionisation, b-f
		Then in next lines:
		PLAM PSIG PWT
	Format:	free free free
		There have to be IFMAX triple of quantities
	Explanation:	PLAM is the wavelength, PSIG is the photoionisation cross section, and PWT the integration weight for the resolved cross section
		F(alse) for line transitions, b-b

*** next NRFIX lines:**

Designation	Format	Explanation
IR	free	from
JR	free	to level number (if b-f: J=NXTION(I))
PHOR	free	T(rue) for b-f F(alse) for b-b
OCR	free	Dummy, if b-f log(gf), if b-b

*** next new line:** XXI1 XXI2

Format: free

Explanation: XXI1 is the number of valence electrons of the first ionization stage, XXI2 that of the second ionization stage (Drawin, Coll. and Transp. Cross-Sections, S.14)

*** next new line:** SCALE FTOAL IPIA02

Format: free

Explanation: SCALE is a scaling factor for electron- and neutral particle collision cross sections. As a starting point, SCALE should be set to 1.0.

FTOAL is the factor between forbidden and allowed cross sections (10 is a good choice in case of IPIA02=0, see Allen 1973, Astr.Quant. p.44)

IPIA02=2:	default cross section that is computed with the mean quadratic orbital radii RRCA
IPIA02=1:	default cross section = PI*a0^2
IPIA02=0:	default cross section is energy dependent after Allen (1973)

*** next new line:** N1

Format: free

Explanation: N1 is the number of optically allowed transitions of the atom, for which default cross sections are to be used

*** next new lines:** ICLOW ICUP

Format: free free

There have to be N1 pairs of values distributed over the lines. Separation of the values by blanks!

Explanation: For the optically allowed transitions, for which default cross sections are to be used, ICLOW is the number of the lower level and ICUP the one of the upper level (ICLOW<ICUP)

*** next new line:** N2

Format: free

Explanation: N2 is the number of transitions with collision rates which are to be computed following the Born approximation (Sobelman table)

*** next new line:** ICLOW ICUP

Format: free free

There have to be listed N2 pairs of values.

Explanation: ICLOW and ICUP are the level numbers of the lower and upper energy level.

*** next new line:** FELBB FELBF

Format: free

Explanation: FELBB is a scaling factor for electron collision rates for bound-bound transitions. As a starting point set FELBB=1.

FELBF is the respective scaling factor for bound-free transitions.

*** next new line:** N3

Format: free

Explanation: N3 is the number of following listed transitions, for which different cross sections are to be used.

*** next new lines:** ICLOW ICUP CCS

Format: free free free

There have to be listed N3 triples of values, arbitrarily distributed over the lines.

Explanation: ICLOW and ICUP are again the level numbers, but depending on the sign the values have different meaning. Different cases have to be distinguished:

a) ICLOW > 0, ICUP > 0 : for optically allowed transitions
=> CCS=max. electron cross section

If ICUP=NXTION(ICLOW), collisional ionization (if no default values are to be used)

b) ICLOW > 0, ICUP < 0 : for optically allowed transitions
=> CCS=log(gf) for dipole approximation (for carbon the minimal cross section is 0.1*pi*a0^2)

c) ICLOW < 0, ICUP > 0 : for optically forbidden transitions
=> CCS= max. electron cross section (only for those transitions for which no default cross section is to be used)

d) ICLOW < 0, ICUP < 0 : not allowed at this point

Note: All collisional ionization cross sections have to be given in units of pi*a0^2!! pi=3.1415..., and a0=0.52921E-8 cm^2 (Bohr radius). In total: pi*a0^2 = 8.798E-17 cm^2.

*** next new line:** N4

Format: free

Explanation: N4 is the number of following listed transitions, for which energy resolved cross sections are to be used.

*** next new lines:** ICLOW ICUP NPOINTS

Format: free free free

There have to be N4 triple of values, arbitrarily distributed over the lines.

Explanation: ICLOW and ICUP are again the level numbers, but:

ICLOW and ICUP have to have a negative sign. NPOINTS is the number of following pairs of values. Note for the sign of NPOINTS:

NPOINTS > 0: optically allowed transition
NPOINTS < 0: optically forbidden transition

This is followed by NPOINTS of pairs ECOLL and SCOLL. ECOLL is the energy in eV and SCOLL the corresponding collision cross section. All values have free format and can be distributed arbitrarily over the lines.

*** next new line:** FH1BB FH1BF

Format: free

Explanation: FH1BB is a scaling factor for bound-bound neutral particle collisions. Normally, FH1BB = 0.333, because the default values are computed from the electron collisions, and following Steenbock (1985) the ratio between neutral particle and electron collision cross sections is approximately 1/3.

FH1BF is the respective scaling factor for bound-free neutral particle collisions.

*** next new line:** N1

Format: free

Explanation: N1 is the number of transitions, for which no default values are to be used.

*** next new lines:** ICLOW ICUP CCS

Format: free free free

There have to be N1 triples of values, arbitrarily distributed over the lines.

Explanation: ICLOW, ICUP and CCS, all with positive signs, denote the numbers of the lower and upper level and the maximum neutral particle collision cross section.

*** next new line:** N2

Format: free

Explanation: N2 is the number of the following listed transitions for which energy resolved cross sections are to be used.

*** next new lines:** ICLOW ICUP NPOINTS

Format: free free free

There have to be N2 triples of these values arbitrarily listed over the lines.

Explanation: ICLOW and ICUP are again level numbers, but:

ICLOW and ICUP must have a negative sign. NPOINTS denotes the number of following pairs of values. Note for the sign of NPOINTS;

NPOINTS > 0:	optically allowed transition
NPOINTS < 0:	optically forbidden transition

There follow NPOINTS pairs of values ECOLL and SCOLL. ECOLL denotes the energy in eV and SCOLL the corresponding cross section. All values have free format and can be arbitrarily distributed over the lines.

*** next new line:** NFMAX1

Format: free

Explanation: NFMAX1 is the number of frequency points, which are to be used for the computation of the Voigt profile for the listed b-b transitions.

NFMAX1=0: a grid of 30 standard points is used
NFMAX1>0: list the frequency points FREQUA in units of the maximum Doppler width; NFMAX1 points should be listed in free format over the next lines.

*** next new line:** ITT, ITYP

Format: free

Explanation: Three cases are to be distinguished:

ITYP=1: ITT denotes the transition where the continuum background opacities are to be scaled by a factor different from 1.

a new line follows: OPSCA

Format: free

Explanation: OPCSCA is the scaling factor for the chosen b-b transitions.

ITYP>=100: ITT denotes the transition for which background lines should be taken into account; ITYP is then the atomic number of the element to which the background line belongs. List in the next line:

BLAM, BCHI, BGFLG, BDRRCA, BC4LG, BRAD

Explanation: BLAM is the wavelength of the line, BCHI the excitation energy of the lower level, BGFLG the log gf value, BDRRCA the mean quadratic orbital radius of the lower level, BC4LG log C4, BRAD Gamma rad.

ITYP<: ITT is not important and all hydrogen lines that lie within 300 Å of the respective line IT are computed as background opacity.

ITT=0: The scaling factor for all transitions if one.

---------- Example for an ATOM data file: CI ----------
Comments that do not belong into the file are listed after #

```
---------- CARBON ---------- S.Stuerenburg 1989, 1992 ----------
 600
  88   66   88
 1  0.0037  0.0000    0.629  9.00   84 '2P3P'
 2  1.2637  0.0000    1.226  5.00   84 '2P1D'
 3  2.6840  0.0000    2.323  1.00   84 '2P1S'
 4  4.1826  0.0000    4.427  5.00   85 '2P5S'
 5  7.4853  0.0000   34.254  9.00   84 '3S3P'
 6  7.6848  0.0000   38.077  3.00   84 '3S1P'
 7  7.9460  0.0000   31.844 15.00   85 '2P3D'
 8  8.5371  0.0000   49.880  3.00   84 '3P1P'
 9  8.6444  0.0000   54.591 15.00   84 '3P3D'
10  8.7711  0.0000   60.987  3.00   84 '3P3S'
11  8.8494  0.0000   65.467  9.00   84 '3P3P'
12  9.0026  0.0000   75.678  5.00   84 '3P1D'
13  9.1718  0.0000   89.763  1.00   84 '3P1S'
14  9.3304  0.0000  106.572  9.00   85 '2P3P'
15  9.6311  0.0000  103.300  5.00   84 '3D1D'
16  9.6872  0.0000  191.260  9.00   84 '4S3P'
17  9.6988  0.0000  115.676 21.00   84 '3D3F'
18  9.7092  0.0000  117.721 15.00   84 '3D3D'
19  9.7130  0.0000  197.599  3.00   84 '4S1P'
20  9.7364  0.0000  123.328  7.00   84 '3D1F'
21  9.7614  0.0000  128.765  3.00   84 '3D1P'
22  9.8338  0.0000  146.274  9.00   84 '3D3P'
23  9.9885  0.0000  259.282  3.00   84 '4P1P'
24 10.0195  0.0000  273.100 15.00   84 '4P3D'
25 10.0557  0.0000  290.613  3.00   84 '4P3S'
26 10.0842  0.0000  305.529  9.00   84 '4P3P'
27 10.1382  0.0000  337.110  5.00   84 '4P1D'
28 10.1979  0.0000  377.908  1.00   84 '4P1S'
29 10.3524  0.0000  433.962  5.00   84 '4D1D'
30 10.3865  0.0000  613.769  9.00   84 '5S3P'
```

31 10.3866 0.0000 473.871 21.00 84 '4D3F'
32 10.3948 0.0000 484.070 15.00 84 '4D3D'
33 10.3995 0.0000 632.327 3.00 84 '5S1P'
34 10.4052 0.0000 354.389 28.00 84 '4F9['
35 10.4082 0.0000 501.529 7.00 84 '4D1F'
36 10.4159 0.0000 367.021 56.00 84 '4F9"'
37 10.4187 0.0000 515.846 3.00 84 '4D1P'
38 10.4282 0.0000 529.404 9.00 84 '4D3P'
39 10.5202 0.0000 799.026 3.00 84 '5P1P'
40 10.5341 0.0000 830.782 15.00 84 '5P3D'
41 10.5628 0.0000 902.591 9.00 84 '5P3P'
42 10.5882 0.0000 974.043 5.00 84 '5P1D'
43 10.6162 0.0000 1062.715 1.00 84 '5P1S'
44 10.6856 0.0000 1200.264 5.00 84 '5D1D'
45 10.7052 0.0000 1293.771 21.00 84 '5D3F'
46 10.7062 0.0000 1520.211 9.00 84 '6S3P'
47 10.7107 0.0000 1322.271 15.00 84 '5D3D'
48 10.7139 0.0000 1114.741 28.00 84 '5F9['
49 10.7143 0.0000 1565.181 3.00 84 '6S1P'
50 10.7183 0.0000 1362.594 7.00 84 '5D1F'
51 10.7222 0.0000 1156.231 36.00 84 '5F9"'
52 10.7236 0.0000 1391.415 3.00 84 '5D1P'
53 10.7262 0.0000 1406.347 9.00 84 '5D3P'
54 10.7758 0.0000 1902.254 3.00 84 '6P1P'
55 10.7842 0.0000 1971.448 15.00 84 '6P3D'
56 10.7998 0.0000 2109.055 9.00 84 '6P3P'
57 10.8137 0.0000 2245.181 5.00 84 '6P1D'
58 10.8289 0.0000 2409.329 1.00 84 '6P1S'
59 10.8652 0.0000 2673.876 5.00 84 '6D1D'
60 10.8774 0.0000 2857.227 21.00 84 '6D3F'
61 10.8780 0.0000 3187.273 9.00 84 '7S3P'
62 10.8818 0.0000 2926.777 15.00 84 '6D3D'
63 10.8812 0.0000 2594.173 28.00 84 '6F9['
64 10.8845 0.0000 3298.076 3.00 84 '7S1P'
65 10.8867 0.0000 3008.094 7.00 84 '6D1F'
66 10.8891 0.0000 2719.122 36.00 84 '6F9"'
67 10.8896 0.0000 3057.527 3.00 84 '6D1P'
68 10.8905 0.0000 3074.254 9.00 84 '6D3P'
69 10.9182 0.0000 3858.754 3.00 84 '7P1P'
70 10.9247 0.0000 4012.020 15.00 84 '7P3D'
71 10.9344 0.0000 4257.623 9.00 84 '7P3P'
72 10.9429 0.0000 4491.197 5.00 84 '7P1D'
73 10.9520 0.0000 4764.595 1.00 84 '7P1S'
74 10.9724 0.0000 5189.684 5.00 84 '7D1D'
75 10.9808 0.0000 5518.212 21.00 84 '7D3F'
76 10.9810 0.0000 5966.385 9.00 84 '8S3P'

```
77 10.9846  0.0000  5675.446 15.00   84 '7D3D'
78 10.9819  0.0000  5123.398 28.00   84 '7F9['
79 10.9869  0.0000  6223.010  3.00   84 '8S1P'
80 10.9881  0.0000  5829.561  7.00   84 '7D1F'
81 10.9897  0.0000  5448.490 36.00   84 '7F9"'
82 10.9901  0.0000  5921.180  9.00   84 '7D3P'
83 10.9898  0.0000  5907.585  3.00   84 '7D1P'
84  0.0052  0.0000     1.718  6.00   89 '2P2P'
85  5.3358  6.4992     3.313 12.00   89 '2P4P'
86  9.2903  6.4992     5.867 10.00   89 '2P2D'
87 11.9637  6.4992     9.254  2.00   89 '2P2S'
88 13.7191  12.690    13.076  6.00   89 '2P2P'
89  0.0      0.0       0.0    0.0     0 'CONT'

 +1.000   0.000   0.000

   1    5  1657.008  -0.310  -1.000  -1.000  30  F
   1    7  1561.438  -0.500  -1.000  -1.000  30  F
   1   14  1329.578  -0.730  -1.000  -1.000  30  F
   1   18  1277.550  -0.430  -1.000  -1.000  30  F
   1   22  1261.552  -0.830  -1.000  -1.000  30  F
   2    6  1930.905  -0.250  -1.000  -1.000  30  F
   2   15  1481.764  -1.220  -1.000  -1.000  30  F
   2   20  1463.336  -0.370  -1.000  -1.000  30  F
   2   21  1459.032  -1.310  -1.000  -1.000  30  F
   3    6  2478.561  -1.070  -1.000  -1.000  30  F
   3   21  1751.828  -0.880  -1.000  -1.000  30  F
   5    9 10691.250   0.340  -1.000  -1.000  30  F
   5   11  9094.829   0.150  -1.000  -1.000  30  F
   5   25  4817.371  -2.990  -1.000  -1.000  30  F
   5   26  4771.747  -1.770  -1.000  -1.000  30  F
   5   40  4064.000  -2.830  -1.000  -1.000  30  F
   5   41  4028.732  -2.092  -1.000  -1.000  30  F
   6    8 14542.500  -0.080  -1.000  -1.000  30  F
   6   12  9405.729   0.290  -1.000  -1.000  30  F
   6   13  8335.149  -0.440  -1.000  -1.000  30  F
   6   23  5380.336  -1.890  -1.000  -1.000  30  F
   6   27  5052.167  -1.550  -1.000  -1.000  30  F
   6   28  4932.050  -1.890  -1.000  -1.000  30  F
   6   42  4269.020  -2.540  -1.000  -1.000  30  F
   6   43  4228.326  -2.790  -1.000  -1.000  30  F
   6   54  4009.930  -2.523  -1.000  -1.000  30  F
   6   58  3942.223  -2.569  -1.000  -1.000  30  F
   7   26  5793.116  -2.080  -1.000  -1.000  30  F
   7   34  5041.481  -1.220  -1.000  -1.000  30  F
   7   36  5023.849  -2.280  -1.000  -1.000  30  F
```

```
 8   15 11330.285   0.310  -1.000  -1.000  30  F
 8   29  6828.117  -1.444  -1.000  -1.000  30  F
 8   33  6655.509  -1.253  -1.000  -1.000  30  F
 8   37  6587.608  -1.600  -1.000  -1.000  30  F
 8   49  5693.110  -1.697  -1.000  -1.000  30  F
 8   52  5668.951  -1.967  -1.000  -1.000  30  F
 9   16 11895.750   0.030  -1.000  -1.000  30  F
 9   17 11753.320   0.700  -1.000  -1.000  30  F
 9   18 11659.680   0.060  -1.000  -1.000  30  F
 9   30  7116.939  -0.569  -1.000  -1.000  30  F
 9   31  7116.534  -0.502  -1.000  -1.000  30  F
 9   32  7083.193  -1.116  -1.000  -1.000  30  F
 9   45  6016.315  -1.347  -1.000  -1.000  30  F
 9   46  6013.398  -1.046  -1.000  -1.000  30  F
 9   47  6000.301  -1.870  -1.000  -1.000  30  F
 9   53  5955.626  -2.783  -1.000  -1.000  30  F
 9   61  5550.869  -1.345  -1.000  -1.000  30  F
 9   76  5306.181  -1.592  -1.000  -1.000  30  F
10   30  7675.141  -1.313  -1.000  -1.000  30  F
10   38  7482.003  -0.955  -1.000  -1.000  30  F
11   18 14420.120   0.440  -1.000  -1.000  30  F
11   22 12614.100   0.070  -1.000  -1.000  30  F
11   30  8066.110  -0.754  -1.000  -1.000  30  F
11   38  7853.068  -1.790  -1.000  -1.000  30  F
11   46  6677.307  -1.240  -1.000  -1.000  30  F
11   47  6661.163  -2.046  -1.000  -1.000  30  F
12   19 17448.600   0.010  -1.000  -1.000  30  F
12   20 16890.381   0.580  -1.000  -1.000  30  F
12   37  8753.079  -2.647  -1.000  -1.000  30  F
12   44  7364.734  -1.810  -1.000  -1.000  30  F
12   49  7241.319  -1.620  -1.000  -1.000  30  F
13   19 22906.561  -0.210  -1.000  -1.000  30  F
13   21 21023.131  -0.420  -1.000  -1.000  30  F
84   86  1335.708  -0.333  -1.000  -1.000  30  F
84   87  1037.018  -0.308  -1.000  -1.000  30  F
84   88   904.142   0.225  -1.000  -1.000  30  F

 1   84    T   0.000
 2   84    T   0.000
 3   84    T   0.000
 4   85    T   0.000
 5   84    T   0.000
 6   84    T   0.000
 7   85    T   0.000
 8   84    T   0.000
 9   84    T   0.000
```

10 84 T 0.000
11 84 T 0.000
12 84 T 0.000
13 84 T 0.000
14 85 T 0.000
15 84 T 0.000
16 84 T 0.000
17 84 T 0.000
18 84 T 0.000
19 84 T 0.000
20 84 T 0.000
21 84 T 0.000
22 84 T 0.000
23 84 T 0.000
24 84 T 0.000
25 84 T 0.000
26 84 T 0.000
27 84 T 0.000
28 84 T 0.000
29 84 T 0.000
30 84 T 0.000
31 84 T 0.000
32 84 T 0.000
33 84 T 0.000
34 84 T 0.000
35 84 T 0.000
36 84 T 0.000
37 84 T 0.000
38 84 T 0.000
39 84 T 0.000
40 84 T 0.000
41 84 T 0.000
42 84 T 0.000
43 84 T 0.000
44 84 T 0.000
45 84 T 0.000
46 84 T 0.000
47 84 T 0.000
48 84 T 0.000
49 84 T 0.000
50 84 T 0.000
51 84 T 0.000
52 84 T 0.000
53 84 T 0.000
54 84 T 0.000
55 84 T 0.000

```
56   84    T   0.000
57   84    T   0.000
58   84    T   0.000
59   84    T   0.000
60   84    T   0.000
61   84    T   0.000
62   84    T   0.000
63   84    T   0.000
64   84    T   0.000
65   84    T   0.000
66   84    T   0.000
67   84    T   0.000
68   84    T   0.000
69   84    T   0.000
70   84    T   0.000
71   84    T   0.000
72   84    T   0.000
73   84    T   0.000
74   84    T   0.000
75   84    T   0.000
76   84    T   0.000
77   84    T   0.000
78   84    T   0.000
79   84    T   0.000
80   84    T   0.000
81   84    T   0.000
82   84    T   0.000
83   84    T   0.000
84   89    T   0.000
85   89    T   0.000
86   89    T   0.000
87   89    T   0.000
88   89    T   0.000

2    1                XXI1  XXI2

1.00 1.00  1          SCALE FTOAL IPIA02

0                     Number of allowed with default cross sections

0                     Number of Born cross sections

1.00 1.00             FELBB FELBF

3863                  Number of explicit cross sections
```

```
 -1   2 1.166E-01  -1   3 2.577E-02  -1   4 1.060E-02   1   5 1.000E-01
 -1   6 1.000E-02   1   7 1.000E-01  -1   8 1.000E-02
 -1   9 1.000E-02  -1  10 1.000E-02  -1  11 1.000E-02  -1  12 1.000E-02
 -1  13 1.000E-02   1  14 1.000E-01  -1  15 1.000E-02
  1  16 1.000E-01  -1  17 1.000E-02   1  18 1.000E-01  -1  19 1.000E-02

   |
   |
   |

-82  83 2.064E+06

-84  85 1.000E-02  84  86 1.000E-01  84  87 1.000E-01  84  88 1.000E-01

-85  86 1.184E-02 -85  87 1.000E-02 -85  88 1.000E-02

-86  87 2.590E-02 -86  88 1.000E-02

-87  88 6.007E-02

   1  -5  0.095    1  -7 -0.169    1 -14 -0.351    1 -16 -0.708    1 -18 -0.096
   1 -22 -0.461    1 -30 -2.030    1 -32 -1.280    1 -38 -1.780    1 -46 -2.460
   1 -47 -1.560    1 -53 -1.660    1 -61 -2.760    1 -62 -1.860    1 -68 -2.260

    |
    |
    |

  58 -79 -1.540   58 -83 -2.260
  84 -86 -0.333   84 -87 -0.308   84 -88  0.225
  -1   2  1.303   -1   3  0.142   -1  4  0.836   -2   3  0.308   84  86  1.250

  0               Number of energy resolved cross sections

0.333 0.333         FH1BB FH1BF

  0               Number of H1 explicit cross sections

  0               Number of H1 energy resolved cross sections

  0               NFMAX (=0 default) Freq. points per half-profile

 99999 0          End of background lines
```

2.2.2 The Input Datafile RIJ

The datafile RIJ contains the photionization rates computed in the program RATES. It is read by the subroutine RATCAL. For the detailed structure see 2.2.5.

2.2.3 The Input Datafile ATM

The datafile ATM contains data from the model atmosphere that is computed by the program ATM. See also section 1.1.

2.2.4 The Input Datafile RADIAT

2.2.5 The Output Datafile FIXRAT

The datafile FIXRAT is generated by the subroutine RATCAL (which is called by START). It contains for all depth points the rate matrix, the total population density of the element under consideration, and level population densities of all listed energy levels (all data unformatted). The rate matrix Aij contains at this point of the computation the collision rates, photoionization rates and possibly the rates of a few line transitions with fixed rates.

The datafile is structured as follows:

- [{ A(L,J) } , L=1,NK] , J=1,NK _ For all depth points
- N total in cm^-3 / ID=1,NDEPTH

- [{ N(ID,I) } , ID=1,NDEPTH] ,I=1,NK

  Where: NK= number of levels including the continuum
  NDEPTH= number of depth points

2.2.6 The Output Datafile OPA

This datafile is generated by the subroutine CHIGEN. It contains some information on opacities and profiles for the b-b transitions from ATOM. The file is unformatted and structured as fiollows:

For all transitions:

- CHI ,1...NDEPTH: absorption coefficient in cm^2/g
- CHIE,1...NDEPTH: scattering coefficient
- BNU: pre-factor of BNU for the respective transition

- FREQ: frequency of the transitions
- { SIG ,1...NDEPTH} , 1...N1: Voigt profile for N1 frequency points
- { WT ,1...NDEPTH} , 1...N1: integration weights

2.2.7 The Output Datafile BOUND

This datafile is also generated by the subroutine CHIGEN. It contains the Planck function for all transitions specified in ATOM and for all listed frequency points at three angular points. The file is unformatted and structured as follows:

[{ BC ,1...NMU } ,1...N1] ,1...NT ,

where:

NMU: number of angles, usually three
N1: number of frequency points
NT: number of transitions

2.2.8 The Output Datafile MODEL

The output datafile MODEL is generated by the subroutine OUTMOD. OUTMOD is called directly by the program START. The file MODEl is unformatted and can therefore not be printed or read on the terminal. The file is structured as follows:

- HEAD(I) ,I=1,10: header from file ATOM
- ITER: Iterationszaehler.START: ITER=0.
- NL,NK,NDEPTH,NJ,NT,NPT:
  NL = number of levels in model atom (running index IL)
  NK = NL+1
  NDEPTH = number of depth points
  NJ = total numbers of frequency points for the b-b transitions
  NT = number of b-b transitions
  NPT= number of b-f transitions
- NF1(I) ,I =1,NT: number of frequency points of the individual b-b transitions
- LOW(I) ,I =1,NT: number of lower level for transition
- UP(I) ,I =1,NT: number of upper level for transition
- GR(I) ,I =1,NT: ratio of statistical weights g(low)/g(up)
- PHOTO(I) ,I =1,NT: photoionization T(rue) or F(alse)
- MA(I) ,I =1,NDEPTH: mass scale for the atmosphere in g/cm^2
- RHO(I) ,I =1,NDEPTH: density scale in g/cm^3
- [N(ID,I) ,ID=1,NDEPTH],I=1,NK: level population numbers for all levels at all depth points
- LABEL(I) ,I =1,NK+1: level designation
- NXTION(I) ,I =1,NL: continuum level number for all energy levels
- IJREF(I) ,I =1,MAXREF: translation between IR and IL
- TEMP(ID) ,ID=1,NDEPTH: temperature scale
- TAUREF(ID),ID=1,NDEPTH: optical depth scale

2.2.9 The Standard Output of START

The standard output of START contains information on the used model atom, the computed collision cross sections and the line profiles of the explicitly stated b-b transitions.

START

- LEVELS INCLUDED
    1. Level number
    2. Level designation
    3. Statistical weight
    4. Level energy in eV
    5. Excitation energy of parent ion
    6. mean quadratic orbital radius in units of the square of the Bohr radius
    7. Level number of the continuum

- RADIATIVE TRANSITIONS INCLUDED

    I. DELTA(log[C6]), DELTA(log[C4]), DELTA(log[GAMMA rad])
    II. Table:
        1. Number of the transition
        2. Wavelength in Å
        3. Number of frequency points for computation of Voigt profile (for a half profile!)
        4. Number of lower level
        5. Number of upper level
        6. Designation of lower level
        7. Designation of upper level
        8. log (gf)
        9. -log (C4)
        10. GAMMA rad in 10^8 sec^-1

- RADIATIVE TRANSITIONS WITH FIXED RATES

    1. Number of the transition
    2. PHO: T(rue) for photoionisation transition, F(alse) for b-b transitions with fixed rates
    3. Number of the lower level
    4. Number of the upper level
    5. Designation of lower level
    6. Designation of upper level
    7. if PHO=F: Wavelength in Å
       if PHO=T: 0 (without meaning)
    8. if PHO=F: log (gf)
    9. if PHO=T: Dummy (without meaning)

START,ATMOS

I. In one line

- ELEMENT

  Element number / abundance (H=12) / fraction (particle density of element divided by the total particle density) / atomic weight of element / mean particle mass in g

II. Table

- MODELL ATMOSPHERE READ IN FROM FILE

  i. In one line: Teff, log(g), HE/H, R/Rsun, R in km
  ii. Table:
    1. Number of depth point
    2. log(tau,ref) with LAMBDA,ref = 5000 Å
    3. Mass scale in g/cm^2; starting at the outer edge
    4. Logarithm of mass
    5. Temperature in K
    6. log (Pel)
    7. log (Pgas)
    8. Density in g/cm^3
    9. Microturbulence velocity in km/s
    10. Radius/stellar radius
    11. Prescribed particle density from ABU
    12. newly computed particle density from the sum of a level population densities

START,RATCAL,COLSEC

- TABLE OF ELECTRON AND NEUTRAL PARTICEL COLLISION CROSS-SECTION

This standard output contains extensive tables, that do not need further explanation, because they are well documented.

Please note the following:

Initially, all possible transitions between the levels have a cross section according to the hydrogen approximation for optically forbidden

transitions. For optically allowed transitions, this value gets multiplied by 10.

Ionisation cross sections are computed following the hydrogen approximation. Then, for all transitions where better (maximal) cross sections are known, these values are adopted and listed. There is also the possibility to specify explicit cross sections. The neutral particle cross sections get the same value as the electron collision cross section, except in the case of energy resolved electron cross sections. In that case they get the maximum value of the energy resolved cross section. This holds in all cases for which no different values are listed in ATOM.

START,CHIGEN

I.

- FREQUENCY QUADRATURE POINTS FROM LINE CENTER IN DOPPLER WIDTH

1. Number of frequency points per half profile
2. Frequencies in units of maximum Doppler width

Attention: a) This is the prescribed frequency grid, that can be changed in ATOM.
b) Note that the computation of the Voigt profiles of the desired b-b transitions does NOT run up to the last frequency point. Instead the number of points to be used from this grid can be specified in ATOM. See point II.3!

II. Table

- LAMBDA SPACING IN THE LINES

1. Number of the transition
2. Maximal width up to which the line profile was computed in units of the maximum Doppler width
3. Number of the depth point at which the integral over the Voigt profile has a maximum
4. The respective value
5. Number of depth point at which the integral over the Voigt profile has a minimum
6. The respective value
7. The used frequency grid in Å

Note:

- Concerning 4. and 6.: Since the Voigt profile is

normalized, the integral over the entire profile has to be one. If the value very different from one, the number of frequency points given in ATOM has to be extended. Hence, the listed values serve control purposes.

- Concerning 7.: If the same value appears for different lines under 2., the maximum value of the frequency grid provides direct information on the actual width of the line!

2.3 The Program EDDFAC

EDDFAC:

- Reading control indices IFLPT, IFLRAT, IFPRAT und IFRFLD
- Open files OPA, BOUND and RURD
- If IFRFLD=1
  - Open file RADFK
- Call INPMOD to read data file MODEL
- If IFLPT=1
  - Standard output of heading for this computation and of the iteration counter ITER
- For all transitions
  - Computation of OUTRAT (see 2.3.3)
  - If b-b- transition
    - Read absorption and scattering coefficients, pre-factor of Planck function and frequency of transition from file OPA
  - For all depth points
    - Initialize Rij and Rji (radiative rates=0)
  - For all frequency points of a transition
    - If b-f transition
      - Read absorption and scattering coefficients, pre-factor of Planck function and frequency of transition, ratio of statistical weights, cross section and integration weight from file OPA
    - If b-b transition
      - Read sigma and integration weight from file OPA
    - Read BC for all angles from file BOUND (radiation field at the inner boundary)
    - For all depth points
      - Compute total opacity
    - Solve radiative transfer equation using Feautrier with variable Eddington factors (Auer & Mihalas 1970) (see subroutine RADI)
    - If IFRFLD=1
      - Store radiation field JNU and Eddington factors FK in data file RADFK
      - Compute emergent flux HPLUS
      - Store Eddington factors FH0 and FH1 as well as HPLUS in data file RADFK
    - If OUTRAT=1
      - For all depth points
        - Computer radiative rates Rij and Rji
  - The line radiative rates are stored for all depth points in data file RURD
  - If IFLPT=1
    - Standard output about line fluxes
- Close data files RURD, OPA, BOUND

- If IFRFLD=1
  - Close data file RADFK

INPMOD:

Subroutine to read data file MODEL.

2.3.1 The Input Datafiles OPA, BOUND and MODEL

The three input files OPA, BOUND and MODEL are the output files from START (see 2.2.6, 2.2.7, 2.2.8).

2.3.2 The Input Datafile $IN

$IN is the standard input of the program EDDFAC. The program reads four control indices: IFLPT, IFLRAT, IFPRAT und IFRFLD. The indices can take either the value 0 (no action) or 1. They have the following functions:

- If IFLPT=1: generate a standard output of the computation
- If IFLRAT=1: radiative rates Rij and Rji are computed again in EDDFAC
- If IFPRAT=1: photoionization rates Ric and Rci are computed again in EDDFAC
- If IFRFLD=1: a file with name RADFK is generated, in which the radiation field JNU, and the Eddington factors FK are stored for all depth; also stored are FH0, FH1 and HPLUS to denote the boundary condition. All this information is stored for all frequency points of a line and foe all b-b transitions.

2.3.3 The Output Datafile RURD

This output file is generated by EDDFAC and is unformatted. If OUTRAT=1 (OUTRAT = (IFLRAT.EQ.1 .AND. .NOT. PHOTO) .OR. (IFPRAT.EQ.1 .AND. PHOTO)), radiative rates Rij and Rji (in EDDFAC those are called RU (R up) and RD (R down)) are computed and then stored in RURD (=RU+RD). If OUTRAT=0, only zeros are stored in RURD. The quantities are stored as follows:

[(Rij),1...NDEPTH ; (Rji),1...NDEPTH], 1...NT

where NT denotes the number of transitions.

2.3.4 The Output Datafile RADFK

This file is only generated by EDDFAC if IFRFLD=1. The unformatted file is structured as follows:

{[(RAD),1...NDEPTH ; (FK),1...NDEPTH ; FH0 ; FH1 ; HPLUS],1..NF},1..NT

Explanation: RAD = J : radiation field, mean intensity
FK = K/J : Eddington factor
FH0 =(H-I-)/J : for boundary condition at the first depth point

FH1 = H/J : for boundary condition at the last depth point
HPLUS=H+ : emergent Eddington flux

Please note the name of the file: RADFK = RAD + FK !

2.3.5 The Standard Output of EDDFAC

The standard output of EDDFAC is only generated if IFLPT=1. The output looks then as follows:

- Heading as specified in ATOM
- ITERATION, followed by a number that serves as iteration counter
- For each b-b transition
    - -IT= … F0= … F1= …
      Here IT denotes the transition number, F0 the flux at the first frequency point and F1 the flux at the last frequency point of the line.
    - (FLUX), 1…NF
      Here FLUX denotes the ratio of the flux at a specific frequency point to the flux at the last frequency point. NF is the number of frequency points.

2.4 The Program GAB

GAB:

- Reading 10 control indices IFLPT, IFFILE, IFITER, IFPLOT, IFDEPE, IFLOGB, IFRU, IFION, IFTAU, IFSB
- Call subroutine INPMOD to read file MODEL
- Cal subroutine POPS to compute new level population numbers including the line radiative rates
- If IFITER=1
- ITER=ITER+1 (iteration counter)
- Call subroutine OUTMOD to generate the output file MODEL
- If IFDEPE=1
  - Open file DEPART
  - Store number of levels, level designation, element number, departure coefficients for all depth and all levels, the parent term level number, the reference field for level numbers, the optical depth scale, the temperature stratification, the header and the iteration counter
  - Close data file DEPART
- If IFLPT=0
  - If IFPLOT=1
    - Standard output of iteration counter and the maximum relative change in level population numbers
- If IFPLOT=1
  - Line printer output of the relative change in level population numbers
  - Line printer output of departures
- If IFLOGB>0
  - Standard output of log(n) and log(b) for each IFLOGBs depth point
- If IFRU>0
  - For all transitions, Z = 1 - (Nj*RD) / (Ni*RU) is computed, a measure of the ratio between upwards rates and downwards rates
  - Standard output of Z for each IFRUs depth point
- If IFION>0
  - For all depth point
    - For all level
      - Read old (LTE) rate matrix from file FIXRAT
    - Read total level population number from FIXRAT
    - For all levels
      - Compute upward rates RCIJ to continuum
      - Compute downward rates from continuum RCJI
    - For all transitions
      - If b-f transition
        - Add photoionisation rates Ric to RCIJ
        - Add recombination rate Rci ro RCJI

- Compute contribution to ionization equilibrium [Ni*(Ric+Cic) - Nc*(Rci+Cci)] / Sum,i[Ni*(Ric+Cic)] separate for all ionization stages
- Close file FIXRAT
- Standard output for each IFIONs depth point of contribution of each level to photoionization

- If IFTAU>1 or IFSB>0
  - Open file OPA
  - For all transition
    - Read opacities for all depth points
    - Read pre-factor of Planck function and frequency of transition
    - If b-f transition
      - Read ratio of statistical weights for all depth points
      - Read photoionisation cross section and integration weight
    - If b-b transition
      - Read sigma and integration weight
    - For all depth points
      - Compute new opacity
    - For all other frequency points
      - If b-f transition
        - Read ratio of statistical weights for all depth points
        - Read photoionisation cross section and integration weight
      - If b-b transition
        - Read sigma and integration weight
    - For all depth points
      - Compute new opacity
  - Close file OPA
- If IFTAU=1
  - IFTAU = NDEPTH – 1
- If IFTAU>0
  - For all lines
    - Compute optical depth at first frequency point (line center) and last frequency point (continuum) of the line profile
    - Compute depth for which tau becomes smaller than 1
    - Standard output of these results
- If IFSB>0
  - For all transition
    - For all depth points
      - Compute the ratio of line source function to Planck function
        S(Linie)/BNU = [(Ni(LTE)/Nj(LTE))*(gj/gi)-1]/[(Ni/Nj)*(gj/gi)-1]
      - Standard output of logarithm of this value

first frequency point (line center)

POPS:

Subroutine to compute the new level population numbers.

- If IFILE=1
  - Open file DNIDZT
    - For all level
      - For all transitions
        - Find those levels that are involved in the given b-b transitions
        - Numerize these levels
        - Store this enumeration scheme in file DNIDZT
- Open file RURD
- For all transitions
  - Read radiative rates from file DNIDZT for all depth points
- Close file RURD
- For all transitions
  - For all depth points
    - Initialize up- and downwards rates to zero
- Open file FIXRAT
- For all depth points
  - For all levels
    - Read matrix A with rates from FIXRAT
    - Read total level population density (in cm^{-3}) of the atom
  - For all transitions
    - The radiative rates are added to the rate matrix
  - Invert rate matrix A using subroutine MATINV
  - For all levels
    - Compute new level population numbers from inverted rate matrix
      n = A^-1 * b
    - Compute relative change in level population numbers P = n/n(old) – 1
    - Compute maximum change in level population numbers
  - Compute level population number of continuum
  - Compute relative change in level population number of continuum
  - For all level
    - For all transitions
      - Compute quantity DNKDZT, d(n)i/d(Z)t = -A(i,l), where l is the lower level of the transition
      - If i is not the continuum
        - d(n)i/d(Z)t = d(n)i/d(Z)t + A(i,u), where u is the upper level of the transition
    - store DNKDZT for all transitions in file DNIDZT

Note: For the levels of the individual transitions, the rates are computed leading to the levels of the other transitions!

- if IFILE=1

    - Close the file DNIDZT
- For all levels
    - The old level population numbers are read from the data file FIXRAT

MATINV:

Subroutine to invert a matrix (max 100 X 100).

2.4.1 The Input Datafiles MODEL, FIXRAT and RURD

These files are always opened and read by the program GAB. The files FIXRAT and MODEL are output files of the program START (see 2.2.5 and 2.2.8). The file RURD is an output file of the program EDDFAC (see 2.3.3).

2.4.2 The Input Datafile OPA

If the control indices from $IN are IFTAU=1 or IFSB<>0, the file OPA is opened and read. This file is an output file of the program START (see 2.2.6).

2.4.3 Der Input Datafile $IN

$IN is here the standard input for the program GAB. It consists of 10 control indices:

IFLPT, IFFILE, IFITER, IFPLOT, IFDEPE, IFLOGB, IFRU, IFION, IFTAU, IFSB

These values have to be in the same line. Most indices can take either the value 1 (which mean something happens in the program) or 0 (nothing happens). Exceptions are the indices IFLOGB, IFRU, IFION and IFSB. The value 0 also means that nothing is happening; alternatively, these indices can take the values 1, 2, …, NDEPTH. The standard output is then restricted to depth points with the specified step width IFLOGB, IFRU, IFION and IFSB.

The control indices IFLPT and IFPLOT are closely linked:

IFLPT	IFPLOT	Action
0	0	nothing
0	1	standard output of an information line that contains the value of IFITER, the number of iterations done so far and the maximum relative change in level population numbers
1	0	standard output of header and iterations done so far
1	1	standard output of header and iterations done so far as well as line printer output about the relative change in levels population numbers and about departure coefficients
IFFILE	=0	none
	=1	the output file DNIDZT is generated
IFITER	=0	none
	=1	as program GAB is executed, the iteration counter

		ITER is increased by one.
IFDEPE	=0	none
	=1	the output file DEPART is generated
IFLOGB	=0	none
	=1… NDEPTH	standard output of log(N) and log(b) for each IFLOGBs depth point

Note: IFLPT=1 is a requirement for the standard output!

IFRU	=0	none
	=1… NDEPTH	standard output of 1-(Nj*RD)/(Ni*RU) for each IFRUs depth point
IFION	=0	none
	=1… NDEPTH	standard output of contribution of each level to the ionization equilibrium at each IFIONs depth point
IFTAU	=0	none
	=1	standard output of logarithm of TAU in the line center and neighbouring continuum, as well as that depth point, where TAU becomes smaller than 1 for each line
IFSB	=0	none
	=1… NDEPTH	standard output of log(S(line) / BNU) for each IFSBs depth point

2.4.4 The Output Datafile DNIDZT

If the control index IFFILE=1, the subroutine POPS generates the output file DNIDZT. The file is unformatted and the data is structured as follows:

- IJT, 1… NK — enumeration of all levels, which occur in the transitions
- [(DNIDZT, 1…NT), 1…NK1], 1…NDEPTH
  = d(Ni)/d(Z) for all transitions, for all levels that occur in the transitions (NK1) and for all depth points

2.4.5 The Output Datafile DEPART

If IFDEP=1, the file DEPART is generated in the main program GAB. It contains the departure coefficients b=N/N(LTE). The file is unformatted and structured as follows:

- NKP1 =NK+1, number of levels + 1 (LTE)
- LABEL, 1…NKP1 level designation
- LABEL(NKP1+1) element number
- (BS, 1…NKP1), 1…NDEPTH departure coefficients

2.4.6 The Output Datafile MODEL

The output file MODEL differs from the input file MODEL only by an improved computation of the level population numbers in GAB and by a possible increase in the iteration counter ITER. The file is structured in the same way as in START (see 2.2.8).

2.4.7 The Standard Output of GAB

The standard output of GAB provides information on a variety of quantities that are somehow connected to the newly computed level population numbers. The amount of information depends mainly on the control indices in the standard input $IN. Hence the following standard output is sorted according to control indices:

1. IFLPT=0 and IFPLOT=1:
   - IFITER=…, ITERATION=…, MAX(DELTA N/N)=…

     IFITER : here, the value of the control index is printed
     ITERATION : number of calls to GAB, where the control index IFITER=1
     MAX(…) : maximum relative change in the newly computed population numbers Max[| (N-N(old)) / N(old) |]

2. IFLPT=1:
   - Header of the computation
   - ITERATION …

   and IFPLOT=1:

   - PLOT OF RELATIVE CHANGE IN LEVEL POPULATION

   A line printer output is generated of the following quantities for all level and all depth points: P[i,tau] = (N - N(old)) / N(old).

Axis notation: 10 → 1.0 * scaling factor
5 → 0.5 * scaling factor

Level designation: Level(1)=0, Level(2)=1, … 9 A B … Z " # $ % & ' ( ) * + , - . / : ; < = > ? @ [\ ] ^ _ 0 1 2 ... 9 A B … etc.

and IFLOGB<>0:

- ID LN(1) LB(1) LN(2) LB(2) …

The values log(N) and log(N/N(LTE))=log(b) are printed for all lines and for each IFLOGBs depth point.

3. IFRU<>0:

   - ID NET (1) NET(2) NET(3) …

   The following quantity is printed for each line and each IFRUs depth point:

   $$Z = 1. - (Nj*RD)/(Ni*RU).$$

   This denotes the ratio between downwards to upwards radiative rates for each single transition.

   Z>0: upwards rate dominates
   Z<0: downwards rate dominates

4. IFION<>0:

   - CONTRIBUTION PJ OF LEVELS TO ION. EQUILIBRIUM

   The contribution of each level to the ionization balance is computed and printed, Pj=(Nj(Rjc+Cjc)-Nc(Rcj+Ccj))/(Sum[Nj(Ric+Cic)]).

   c : number of continuum
   Pj=0 : like LTE, ionization rate = recombination rate
   Pj>0 : ionization rate larger than recombination rate
   Pj<0 : recombination rate larger than ionization rate

5. IFTAU=1:

   - ID LT0(1) LOG TC LT0(2) LOG TC …

   The logarithm of the optical depth in the line center (LT0) and in the neighbouringg continnum (LOG TC) is printed for the first and last depth

point. Note: LT0=LOG T0! For each line, the depth point, where the optical depth of the line becomes smaller than 1 is printed.

6. IFSB<>0:

   - ID L(S/B) L(S/B) …

   The value log[S(line) / BNU] is printed for each line and each IFSBs depth point. S(line) is here the line source function that has been computed with the new level population numbers. And BNU is the Planck function (in LTE S(line) = BNU).

7. IFLPT=1 and IFPLOT=1:

   - PLOT OF LOG DEPARTURES ---SCALING FACTOR= 0.5---

   A line printer output of the departure coefficients (log b) is generated.

   Level designation: Level(1)=0, Level(2)=1…(see item 2.)

2.5 The Program LINEAR

LINEAR:

- Open files OPA and RADFK
- Read control indices IIMAX, W, EPS, IFPLO from $IN
- Call subroutine INPMOD to read file MODEL
- Call subroutine DN to read file DNIDZT
- For all transitions
  - If b-f transition
    - OUTPCOM = 0.0
  - If b-b transition
    - Read opacities from file OPA for all depth points
    - Read pre-factor for the Planck function and the frequency of the transition
  - For all depth points
    - OUTCOM = 0.0
  - For all frequency points in the transition
    - Call subroutine MATGEN to compute the matrices necessary for the following equation F*delta(J)m + R`*delta(n)l + S`*delta(n)u = L`
    - Call subroutine TRIINV to invert the main matrix F
    - Multiply matrices R' and S' with F^-1 for b-b and b-f transitions respectively; the result are the matrices Rt and St, which form the equation delta(Z)t + Rt*Sum,t`[d(n)l/d(Z)t`*delta(Z)t`] + St*Sum,t`[d(n)u/d(Z)t`*delta(Z)t`] = Lt
    - Call subroutine DN with MODE=1
    - Compute delta(Z)t matrix for all depth points (net rates)
    - Call subroutine LUFAC and SOLVE to solve the equation for Delta(Z)t in zero-th approximation
  - Close files OPA and RADFK
  - Iteration loop (Gauss-Seidel with over-relaxation factor)
    - For all transitions
      - Call subroutine DN with MODE=3 and SOLVE to solve the above outlined equation for Delta(Z)

        delta(Z)i+1,t,l = 1/4 [delta(Z)i,t+1,l + delta(Z)i+1,t-1,l + delta(Z)i,t,l+1 + delta(Z)i+1,t,l-1] - DELTA^2/4*(Rho)t,l
      - Introduction of the over-relaxation factor W, i.e. W* Delta(Z)i,t + (1-W)*Delta(Z)i-1,t is used in place of Delta(Z)i,t
      - Stop if number of iterations is larger than IMAX or if the relative change between the iteration steps becomes smaller than EPS
  - Open file RURD
  - For all transitions
    - Read rates Rij and Tji from RURD

- Call subroutine COMP to generate the standard output
- For all transitions
  - If b-b transition
    - For all depth points
      - Compute new radiative rates Rij and RD using a correction to the old radiative rates
        delta(RU) = Delta(Z)t / (Ni - Nj*gi/gj)
        RU = RU + delta(RU)
        RD = RD + delta(RU)*gi/gj
  - If b-f transition
    - Call subroutine DN with MODE=4 to compute delta(n)i and delta(n)j
    - Solve matrix equation for Delta(Z)t
      delta(Z')t = Lt - (Rt*delta(n)i + St*delta(n)j)
    - For all depth points
    - Compute new radiative rates RU and RD using a correction to the old radiative rates Delta(Z')t
      RU = RU + Delta(Z')t
      RD = RD + (Ni*Delta(Z')t - Delta(Z)t) / Nj
- Store the new radiative rates for all depth points in file RURD\
- Close file RURD

DN:

Subroutine DN has different tasks depending on the value of the control index MODE.

- MODE=1: o Open the file DNIDZT
  o Read data from file DNIDZT
  o close file DNIDZT

- MODE=2: o compute the derivative of the level population number of the lower and upper level with respect to the net rate d(Z)t of the transition J-I,
  DNI = DNDZ(I)
  DNJ = DNDZ(J)
- MODE=3: o sum of the derivaties multiplied with delta(Z)t, each level with respect to each transition except I-J
  result are the delta(n)i and delta(n)j
- MODE=4: o same as MODE=3, but with transition I-J

MATGEN:

This subroutine computes for a given frequency point the matrices for all depth points. The matrix equation results then in a way very similar to the Feautrier schema.

TRIINV:

Subroutine to invert a triagonal matrix.

LUFAC:

Subroutine for LU decomposition of a matrix (here delta(Z)t).

SOLVE:

Subroutine to back substitute with the columns of Lt (in place of first getting delta(Z)t^-1 and then multiplying with Lt; saves one matrix multiplication and is more exact, see Numerical Recipe in Fortran).

COMP:

Subroutine to generate the standard output on the relative changes in line rates.

- For all depth points
  - For all transitions
    - Compute relative change in line rate
      delta(R)/R = Delta(Z)t / [RU * Ni]
- If IFPLO=1
  - Line printer output on relative change in line rates
  - Standard output on number of Gauss-Seidel iterations, maximum change in line rates and the remaining arrays

2.5.1 The Input Datafiles OPA, RADFK, MODEL, RURD AND DNIDZT

These input files are output files of previous programs; OPA is an output file of START (see 2.2.6), RURD and RADFK are output files of EDDFAC (see 2.3.3 and 2.3.4) and DNIDZT and MODEL are output files of GAB (see 2.4.4 and 2.4.6).

2.5.2 The Input Datafile $IN

The standard input of the program LINEAR, $IN, is composed of four control indices, IIMAX, W, EPS, IFPLO. All four values have to be in the same line.

- IIMAX : this control index denotes the maximum number of Gauss-Seidel iterations used to solve the matrix equation in linear. If the solution is not fully converged, the iteration stops after IIMAX steps. 30 is a typical value for IIMAX.

- W : W is the over-relaxation factor to speed up convergence of the iteration. W=1 results in the normal Gauss-Seidel iteration, values larger or equal than 2 lead to divergence. Hence, 1<=W<2 is the possible parameter space for W. The best value can be found for each step using considerable computation resources (see Numerical Recipes in Fortran), but it turns out that in most cases it is good enough to use the value of 1.6, the “best” value found by Auer & Heasley (1976). This value will give a sufficiently fast convergence.
- EPS : this parameter specifies the usual stopping criterium for the iteration. It will stop when
  delta(Z)t,i - delta(Z)t,i-1| / |delta(Z)t,i| <= EPS.
  A typical value for EPS is 1.E-2.
- IFPLO =0 : standard output of an informational line containing the number of iterations and the maximum change in the radiative rates.
  =1 : standard output of the same informational line as for IFPLO=0 and a line printer output of the relative change in radiative rates.

2.5.3 The Output Datafile RURD

The program LINEAR corrects the old radiative rates read in from RURD and writes the newly computed values back into that file. The structure of the file remains the same (see 2.3.3).

2.5.4 The Standard Output of LINEAR

The standard output of LINEAR contains the relative change in radiative rates. The extent of the output depends on the control index IFPLO from $IN.

- IFPLO=0

  LINEAR: SOR-IT=… , ITERATION=… , MAX(DELTA R/R)=…

  The output contains only:
  - The number of successive over-relaxation iterations (SOR), that is the number of internal iterations within LINEAR that are necessary to obtain the correction to the rates
  - The number of main iterations
  - The maximum relative change in level population numbers

- IFPLO=1

  PLOT OF RELATIVE CHANGE IN EACH TRANSITION FOR ....

  It follows a line printer output of the quantity delta(R)/R for all transitions and all depth. Designation of transitions as follows: Line(1)=0, Line(2)=1, …, 8 9 A B … (see 2.4.7 point 2.).

  ---SCALING FACTOR= ...

  Here, the maximum absolute value of delta(R)/R is printed.

2.6 The Program GETITER

GETITER:

- Call subroutine INPMOD to read data from file MODEL
- Standard output of iteration counter ITER

2.7 The Shell Script

The shell script copies the specified input datafiles PTAU, ABUY, CROSS and ATOM into the working directory and renames them to their default names. Then the program package gets started. The program flow is described under Part I, 1.3 of this document. The standard input $IN for all programs is specified in this shell script. The shell script can be edited for changes and it is suggested to keep a copy of the specific version of the shell script together with the results.

Example for a shell script:

```
#############################################################################
#                                                                           #
#          Start the NLTE-Program on a Linux PC                             #
#                                                                           #
#############################################################################
#
#      8 Iterations: N=8
#
#
# path names and file names have to be adapted
#
#
cp ../../atmos/sun/ptauhm.dat PTAU
cp ../../atmos/sun/abusun_2001.dat ABU
cp ../../../science/atom/ci_atom.dat ATOM
cp ../../../science/atom/ci_cross.dat CROSS
ln -s ../odf/bdfP00big2 DF
#
# run ATMOS
#
../bin/ttau
../bin/atm > out
#
# compute radiation field, collision rates, photionisation rates etc.
#
echo 1 | ../bin/rates >> out
../bin/start >> out
#
# pre-iteration
#
echo "0 0 1 0" | ../bin/eddfac >> out
echo "1 0 0 1 0 0 0 0 0 0" | ../bin/gab >> out
echo "0 1 1 0" | ../bin/eddfac >> out
echo "1 0 0 1 1 0 0 0 0 0" | ../bin/gab >> out
```

```
#
# Iteration 1
#
echo "0 1 1 1" | ../bin/eddfac >> out
echo "0 1 0 1 0 0 0 0 0 0" | ../bin/gab >> out
echo "40 1.5 1.e-2 1" | ../bin/linear >> out
echo "1 0 1 1 1 0 0 0 0 0" | ../bin/gab >> out
#
# Iteration 2-(N-1)
#
echo "0 1 1 1" | ../bin/eddfac >> out
echo "0 1 0 1 0 0 0 0 0 0" | ../bin/gab >> out
echo "40 1.5 1.e-2 0" | ../bin/linear >> out
echo "0 0 1 1 1 0 0 0 0 0" | ../bin/gab >> out
#
echo "0 1 1 1" | ../bin/eddfac >> out
echo "0 1 0 1 0 0 0 0 0 0" | ../bin/gab >> out
echo "40 1.5 1.e-2 0" | ../bin/linear >> out
echo "0 0 1 1 1 0 0 0 0 0" | ../bin/gab >> out
#
echo "0 1 1 1" | ../bin/eddfac >> out
echo "0 1 0 1 0 0 0 0 0 0" | ../bin/gab >> out
echo "40 1.5 1.e-2 0" | ../bin/linear >> out
echo "0 0 1 1 1 0 0 0 0 0" | ../bin/gab >> out
#
echo "0 1 1 1" | ../bin/eddfac >> out
echo "0 1 0 1 0 0 0 0 0 0" | ../bin/gab >> out
echo "40 1.5 1.e-2 0" | ../bin/linear >> out
echo "0 0 1 1 1 0 0 0 0 0" | ../bin/gab >> out
#
echo "0 1 1 1" | ../bin/eddfac >> out
echo "0 1 0 1 0 0 0 0 0 0" | ../bin/gab >> out
echo "40 1.5 1.e-2 0" | ../bin/linear >> out
echo "0 0 1 1 1 0 0 0 0 0" | ../bin/gab >> out
#
echo "0 1 1 1" | ../bin/eddfac >> out
echo "0 1 0 1 0 0 0 0 0 0" | ../bin/gab >> out
echo "40 1.5 1.e-2 0" | ../bin/linear >> out
echo "0 0 1 1 1 0 0 0 0 0" | ../bin/gab >> out
#
# Iteration N
#
echo "0 1 1 1" | ../bin/eddfac >> out
echo "0 1 0 1 0 0 0 0 0 0" | ../bin/gab >> out
echo "40 1.5 1.e-2 1" | ../bin/linear >> out
echo "1 0 1 1 1 0 0 0 0 0" | ../bin/gab >> out
```

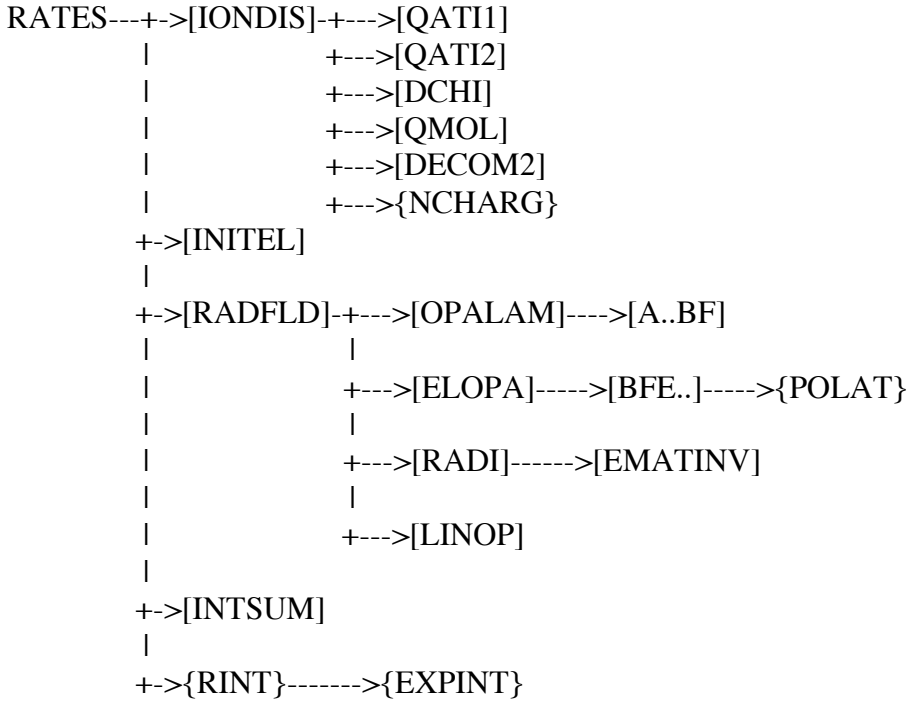
```
#
# Endausgabe
#
echo "1 1 1 0" | ../bin/eddfac >> out
echo "1 0 0 1 1 2 2 2 2 2" | ../bin/gab >> out
#
# rename output files
#
mv DEPART depart.dat
mv out nlte.lpt
```

Appendix A:

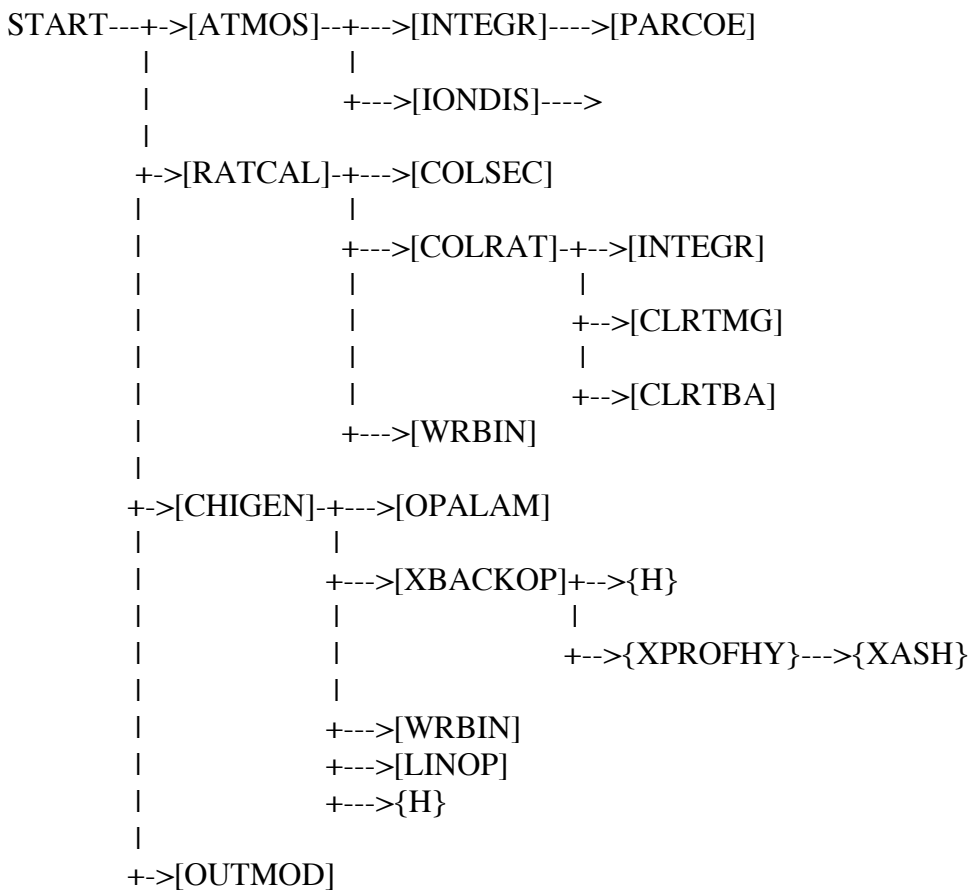
An overview of the programs and their subroutines

The following schema presents an overview of successive call of subroutines within each program in the NLTE program package. The name of subroutines are given in [], functions in {}. All programs have the extension .f. If the filename differs from that of the subroutine, the name is appended in ().

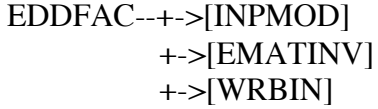
a) The Program RATES:



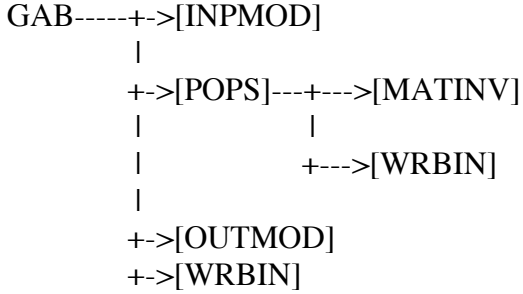
b) The Program START:



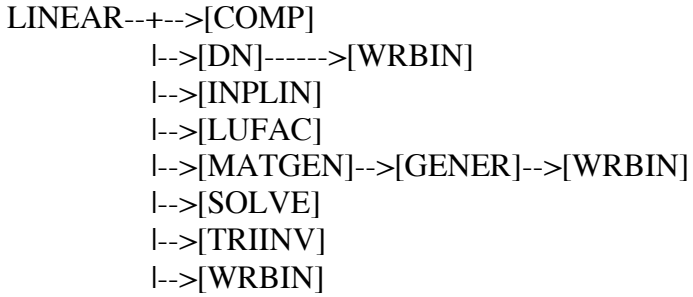
c) The Program EDDFAC:



d) The Program GAB:



e) The Program LINEAR:



Appendix B

Who produced the output?

This appendix lists in alphabetical order the first line of each standard output and which program generated it. The first mentioned program is the main program, all others are subroutines, which were called by the main program. The last item is the subroutine that actually generated the output.

CONTRIBUTION PJ OF LEVELS TO ION. EQ...	--> GAB
ELECTRON COLLISION CROSS SECTION	--> START,RATCAL,COLSEC
ELEMENT	--> START,ATMOS
F L U X E S	--> RATES
FREQUENCY QUADRATURE POINTS	--> START,CHIGEN
ID LN(1) LB(1) ...	--> GAB
ID L(S/B) L(S/B) ...	--> GAB
ID LT0(1) LOG TC ...	--> GAB
ID NET (1) NET (2) ...	--> GAB
IFITER=..,ITERATION=..,MAX(DELTA N/N)=..	--> GAB
Header/ITERATION...	--> GAB
Header/ITERATION.../-IT=..,F0=..	--> EDDFAC
LAMBDA SPACING IN THE LINES	--> START,CHIGEN
LEVELS INCLUDED	--> START
LINEAR:SOR-IT=...,ITERATION=.....	--> LINEAR
MODEL ATMOSPHERE READ IN FROM FILE	--> START,ATMOS
PLOT OF LOG DEPARTURES ...	--> GAB
PLOT OF RELATIVE CHANGE IN EACH TRA..	--> LINEAR
PLOT OF RELATIVE CHANGE IN LEVEL PO..	--> GAB
RADIATIVE TRANSITION INCLUDED	--> START
RADIATIVE TRANSITION WITH FIXED RA..	--> START
TABLE OF ELECTRON AND NEUTRAL PART..	--> START,RATCAL,COLSEC

Appendix C

Error messages

As soon as the program stops at a particular point in the flow, an short informational line will appear specifying the abort reason.

Only in few cases of improper use (e.g. wrong order of parameters in input datafiles ATOM or CROSS) uncontrolled abort can happen. Looking at the standard output will however help in most cases to determine where the abort happened in the code.